RIESBECK

Proceedings of the

Fourth Annual Conference of the Cognitive Science Society

Sponsored by the Program in Cognitive Science
of The University of Chicago and The University
of Michigan,
supported in part by a grant from the
Alfred P. Sloan Foundation.

Ann Arbor, Michigan

August 4-6, 1982

Proceedings of the

Fourth Annual Conference of the Cognitive Science Society

Sponsored by the Program in Cognitive Science
of The University of Chicago and The University
of Michigan,
supported in part by a grant from the
Alfred P. Sloan Foundation.

Ann Arbor, Michigan

August 4-6, 1982

TABLE OF CONTENTS

Symposium—Consciousness

Conscious, Subconscious, Unconscious: A Neodissociation Perspective

John F. Kihlstrom
University of Wisconsin

What gives us the impression that we are conscious? What kind of evidence would convince us that a machine such as a computer, or a lower animal such as a dolphin or a chimpanzee, or -- for that matter -- another human being, was conscious? Cognitive scientists of all stripes, especially those who specialize in psychology, philosophy, and artificial intelligence, disagree violently on the answers, and even on whether these are sensible questions. But nobody doubts that we humans, at least, possess consciousness. The facts that erase any doubt about ourselves are the facts of experience. As James put it in the Principles, "the first fact for us, then. . .is that thinking of some sort goes on" (p. 224). Introspectively, the experience of consciousness seems to have to do with two things: monitoring ourselves and our environment, such that certain perceptual events and memories come to be accurately represented in phenomenal awareness; and controlling ourselves and our environment, such that we are able to voluntarily initiate and terminate behavioral and cognitive activities.

Cognitive science has been vexed by the problem of consciousness since its prehistory. It has had a checkered past, for example, in psychology: almost the whole of the field for James, but a virtual nonentity with the onslaught of the behaviorist movement. Interest in the topic persisted in the hands of the psychoanalysts, and was revived within mainstream psychology with the cognitive revolution and its emphasis on attention and the span of apprehension. Neurologists commonly encounter disorders of consciousness of various types, and those associated with the "split-brain" syndrome have recently received much notice. Ethologists and behavioral biologists have considered whether lower animals possess the capacity for awareness and voluntary control over their actions -- though this concern within comparative psychology has been supplanted to some degree by a series of similar questions having to do with the capacity for language. Parallel concerns have sometimes caught the fancy of those in the artificial intelligence movement, who must deal with the question of whether computers will ever possess consciousness in the sense of awareness and voluntary control over what they are doing. The problems posed by the experience of consciousness for contemporary cognitive science boil down to questions like these: What is the nature of consciousness? What is it good for?

Are there unconscious mental processes, and if so what are they like and what are they good for? Finally, who cares? That is, would cognitive science proceed any differently if its practitioners did not ask questions like these? Let us get some perspective on these questions by turning to some early authorities, before examining some more recent theoretical and empirical developments.

William James devoted the better part of four chapters of the Principles to the topic of consciousness. At the same time, he argued vigorously against the notion of unconscious thought, although he did agree that there were brain processes associated with mental activity of which we might not be aware. As if in warning to Freud and the other psychoanalysts who were to follow, James asserted that the concept of unconscious states of mind "is the sovereign means of believing what one likes in psychology, and of turning what might become a science into a tumbling-ground for whimsies" (p. 163). But the Freudian psychology which was yet to come shared the force of James' critique with other trends in the psychology of his time, such as those which implicated unconscious inference in perception and judgment. To the contrary, he argued that either the allegedly unconscious thought was rapidly forgotten; or that it represented a revision of an earlier (and conscious) thought; or that it was not a thought at all, but merely an innate or habitual brain process. For James, thought and consciousness were identical. It was as difficult for him to contemplate unconscious thought as it was for Hume to contemplate a round square cupola on Berkeley College.

Nevertheless, James did admit that under some circumstances "the total possible consciousness may be split into parts which coexist but mutually ignore each other, and share the objects of knowledge between them" (p. 206). Following Janet and Prince, from whom he drew most of his examples, he referred to this phenomenon as representing "secondary" consciousness, rather than "unconsciousness." In order to understand what James had in mind, it is necessary to consider an important but almost-forgotten school of thought within psychiatry and psychology at the turn of the century.

It is commonly thought that the concept of unconscious mental processes traces its origin to Freud and the theory of psychoanalysis. To the contrary, as Ellenberger has shown, the idea has a long history before Freud. In 1775, with the appearance of Mesmer on the European medical scene, speculation about the unconscious combined with rationalized, materialistic versions of primitive psychotherapeutic procedures to form what is known as the First Dynamic Psychiatry, whose leader was the French neurologist and psychiatrist J.-M. Charcot. This psychiatry was concerned with demonstrable "functional" as opposed to "organic" mental illnesses -- that is, those pathological syndromes which appeared not to be associated with brain insult, injury, or disease. It attempted to account for a wide range of phenomena, including hysteria, fugue (then called ambulatory automatism), and multiple personality; the "magnetic diseases" of catalepsy, lethargy, and somnambulism (so named because of their resemblance to certain phenomena of animal magnetism, a precursor of hypnosis); spiritistic practices such as automatic writing and crystal-gazing; hypnosis; and suggestibility in the normal waking state. Each of these phenomena, the school held, represented the power of ideas to turn into action (one of the meanings of "dynamic" in the psychological sense); and each seemed to reflect a change in consciousness, as thought and actions occurred outside phenomenal awareness and voluntary control.

The First Dynamic Psychiatry, with its emphasis on unconscious mental contents and processes, invoked one or another of two explicit models of the mind. The point of view known as dipsychism (e.g., Dessoir) held that the mind consisted of two layers, each of which in turn consisted of chains

of associations. The "upper consciousness" was active in the normal waking state, while the "lower consciousness" was active in such phenomena as dreams, hysteria, and hypnosis. According to the "closed" version of dipsychism, the lower consciousness contained mental contents which passed into it through the upper consciousness: unattended stimuli, forgotten memories, and various daydreams and fantasies. This point of view contrasts with the less materialistic "open" version, in which the lower consciousness was held to be in direct communication with other minds. According to polypsychism (e.g., Durand de Gros), each segment of the anatomy was served by its own mental structures, called egos, each of which was capable of perception, memory, and thought. These structures, in turn, were subject to the control of a superordinate structure which was identified with normal consciousness. When the link between subordinate and superordinate egos was broken, certain aspects of cognition and action were carried out subconsciously. Clearly, the concepts of dipsychism and polypsychism are at the root of Freud's first (conscious-preconscious-unconscious) and second (id-ego-superego) models of the mind.

The issues confronted by the First Dynamic Psychiatry were subsequently taken up by another French psychiatrist, Pierre Janet. Following the principle of analysis-then-synthesis familiar in physiology, Janet began by considering the elementary parts of the mental system. Instead of following the lead of the earlier faculty psychology, or the chemical analogies of the structuralists, he argued that the elementary structures of the mind were psychological automatisms: complex acts, tuned to environmental and personal circumstances, preceeded by an idea and accompanied by an emotion. Each of these psychological automatisms, by combining cognition, conation, and emotion with action, represented a rudimentary consciousness. According to Janet, all of these elementary automatisms ordinarily were bound together into a single, united stream of consciousness, and operated in awareness and under voluntary control. Under certain circumstances, however, one or more of these automatisms could be split off — Janet's term was disaggregation -- from the rest, functioning either outside awareness, or voluntary control, or both.

This dissociation view of the unconscious, as distinct from the repression view elaborated by Freud and his followers, was further developed by the American psychologist and psychiatrist Morton Prince. Prince, following the practice of his day as exemplified by James' ten arguments against the existence of unconscious thoughts, reserved the term "unconscious" for the dormant traces of forgotten memories and unattended perceptual inputs, as well as the strictly neurophysiological processes associated with mental activity. Instead, he offered the term coconscious, referring to mental activity which takes place outside phenomenal awareness. Prince preferred this term because it connoted mental activity rather than the lack of mentation (as in the ordinary-language conception of unconsciousness associated with concussion or coma); and because it permitted the division of consciousness into parallel streams without one or more of these being outside awareness. Coconscious mental activities performed outside awareness, together with unconscious mental contents and brain processes, formed the subconscious.

This conceptualization of consciousness was very popular on both sides of the Atlantic, featured prominently in the pages of the then-new Journal of Abnormal and Social Psychology (founded and edited by Prince), and was the chief alternative within dynamic psychiatry to Freudian psychoanalysis. However, it was a conceptualization which was short-lived. The eventual dominance of psychoanalysis in clinical psychology and scientific personology led investigators to be interested in different syndromes and phenomena, a different model of the mind, and the eventual replacement of dissociation by repression as the hypothetical mechanism for blocking mental contents from consciousness. At the same time, the behaviorist revolution in academic psychology removed consciousness (not to mention the unconscious) from the vocabulary of the science. At fault as well were the dissociation theorists themselves, who often made extravagant claims for the centrality of their phenomenon and whose investigations were often methodologically flawed. The final blow to the concept stemmed from the interpretation that dissociated streams of consciousness, because they were ignorant (Janet's term) of each other, should not influence each other. Numerous demonstrations of mutual interference between ostensibly dissociated tasks showed the contrary, and reference to dissociation gradually disappeared.

In part, the insistence of both early and late dissociation theorists of non-interference between dissociated mental activities seems to stem from a misunderstanding of James' metaphor of the stream of consciousness. Following the metaphor, it is sometimes held that two streams of water, running parallel but separated by tall banks, should not affect each other. However, if the two streams originate from the same source, each will certainly draw some of the flow from the other. Given a model of attention such as Kahneman's, in which a single source of attentional capacity may be deployed in multiple directions, James' metaphor would certainly lead one to predict some degree of mutual interference between simultaneous, thought dissociated, tasks. In fact, the available evidence indicates that simultaneous tasks performed outside of awareness (for example, in hypnosis) do interfere with each other, with the extent of interference a function of the attentional demands of the tasks in question. Where the tasks are easy, there is little or no interference; where one or both are difficult, interference increases proportionately. Awareness and control are the defining feature of dissociation, while noninterference is an open, empirical question.

Viewed in these terms, a number of phenomena — observed in the laboratory, the clinic, and in the ordinary course of everyday living -- seem to invite a notion such as dissociation. Some of the observations are dramatic, some mundane; the quality of some of the research is impeccable; some demonstrations are marred by poor methodology or contaminated by extraneous social-psychological variables. Some of the results are open to alternative interpretations, and the possibility of performing a definitive experiment seems slim. Some of the claims, in fact, may turn out on close investigation to be false. But not all of them are false. To deny some of them is to deny the facts of our everyday experience. In each of these instances, some aspect of past or present experience cannot be brought into phenomenal awareness, or voluntary control has been lost over thought and action.

Consider, first, the observations of cerebral commisuotomy patients (and intact subjects run under special laboratory conditions), whose right hand literally does not know what the left one is

doing: Here is a division in consciousness associated with a literal division in brain structures. Or consider Korsakoff's syndrome, whose dominant feature is an extremely dense anterograde amnesia: recent experiments have revealed, somewhat surprisingly, that these patients can acquire new information, and that this new learning can have an impact on subsequent cognition and action -- even though the patients have no recollection of the learning experience, and cannot voluntarily retrieve the critical memories. Turning from neurology to psychiatry, there are the very syndromes that caught the attention of the practitioners of the First Dynamic Psychiatry: hysterical anesthesias, paralyses, and amnesias, in which a person complains that he or she cannot remember certain events from the past, perceive stimuli in certain modalities, or voluntarily move certain portions of the body — all in the absence of any demonstrable organic brain syndrome; fugue states, in which a person loses his or her identity as well as the whole of the autobiographical record, relocates, and takes up a new life under a new name; and multiple personality, where separate personalities, each with its own identity, characteristic features, and personal history, seem to inhabit the same body, separated by amnesic barriers and alternating control over overt action and phenomenal awareness.

In the laboratory, phenomena phenotypically similar to the symptoms of hysteria — analgesia and other negative hallucinations, spanning all the perceptual modalities; paralyses; compulsive automatisms in the form of posthypnotic suggestions; and posthypnotic amnesia for events and experiences transpiring during the state -- can be induced in normal subjects simply by the hypnotist's spoken word -- provided that the subjects are hypnotizable to begin with. Under more familiar conditions, we have numerous experiments on divided attention in which information in the unattended channel influences performance outside awareness; and experiments on multiple simultaneous tasks in which complex activities, executed at an acceptable level of performance, are unrecalled afterwards. Then there are all the experiments on perceptual defense and subliminal perception. In the domain of memory, there are of course the phenomena of state-dependent retention, context-dependent retention, and other manifestations of the encoding specificity principle. There are also compelling demonstrations that unremembered experiences can influence perceptual recognition, and of significant savings in relearning material which appears, even after sensitive testing, to have been completely forgotten.

Examples of dissociation can also be found in abundance outside the clinic and the laboratory. One such experience is familiar to all of us: the dream of REM sleep, in which vivid images are constructed without our intending to do so, and in which complex plots are played out five or more times a night (on average), only to be completely forgotten in the morning. Similarly, there is the pavor nocturnus (night terror) common in children, which scares the daylights out of their parents even though the episodes are never remembered by the children themselves. The sleepwalker carries out complex motor activities while deeply in NREM sleep, and remembers nothing of it in the morning. (Sleeptalking, by the way, which also occurs in NREM sleep, is a doubtful case of dissociation, because the speech does not seem to be intelligent or goal-directed in most cases.) Harkening back to the literature on state-dependent retention, there have been demonstrations that some individuals can respond to hypnotic-like suggestions during (REM) sleep, and continue responding on subsequent nights even though they are amnesic for their actions, and the suggestions, during intervening periods of wakefulness.

Given observations such as these, Hilgard has recently revived the concerns of the First Dynamic Psychiatry by proposing a "neodissociation" theory of divided consciousness. He begins with the assumption that the cognitive apparatus is organized hierarchically, with various subsystems monitoring and controlling thought and action in various domains. Under ordinary circumstances, each subsystem is in communication with each of the others, and with a superordinate central executive structure. It is this central executive which is the source of our subjective feelings of awareness and intentionality. Under certain circumstances, Hilgard holds, a subsystem (or more than one) can lose contact with the central executive. In this case, percepts, memories, and actions represented in one of the subsystems fail to be represented in phenomenal awareness; or perceptual exploration, memorial reconstruction, and overt action occur outside the control of the central executive. Despite this loss of communication with the central executive, the dissociated subsystems can, in principle, continue to interact with each other. This continued interaction is the source of the facilitation and interference effects which formed the basis of the empirical critique of the initial versions of dissociation theory.

It should be clear that the subconscious of neodissociation theory is rather different from the unconscious as it is conceptualized by other schools within psychology. Neodissociation theory differs from psychoanalysis, for example, because the subconscious is not restricted to primitive sexual and aggressive impulses, and those memories and ideas associated with them. Nor do subconscious mental processes operate according to the irrational "primary process" principles associated with the Freudian unconscious (as opposed to the rational, "secondary process" of the ego). Dissociated percepts and memories can be closely tied to objective reality; and dissociated ideas can be rational and even creative. Equally important, rendering something subconscious is not necessarily motivated by defense against anxiety, as is the case with Freudian repression. It can simply happen, as in the case of hysteria, fugue, or multiple personality; or it can be done for entirely adaptive purposes, as in the case of the subjects who voluntarily enter hypnosis or go to a movie precisely so they will become totally absorbed in the action on the screen, forgetting for awhile their everyday concerns (and even who they are).

The subconscious of neodissociation theory also differs in important ways from the manner in which unconscious mental contents and processes are construed, at least implicitly, in classical theories of human information processing. Here four major trends can be discerned: an identification of consciousness with attention, short-term memory, or working memory -- in other words, what we are aware of apprehending at any particular moment; with complex as opposed to simple, or difficult as opposed to routine, information-processing procedures; with the availability of linguistic representations for ideas and experiences; and with declarative, as opposed to procedural, knowledge. But the subconscious of neodissociation theory is not restricted to the procedural knowledge by which we detect features in perceptual stimuli, decode and encode language, retrieve memories, make judgments, perform routine motor tasks, and the like. It can also involve complex

factual knowledge, both semantic and episodic in nature, concerning the presence of certain stimuli or the occurrence of certain past events. Nor is it restricted to the simple, automatic, and routine: complex cognitive and behavioral activities apparently can be performed outside awareness. Linguistic contents can be rendered subconscious, and percepts and memories can be subconscious even though the person's linguistic abilities remain intact. Nor, within the realm of declarative knowledge, is the subconscious simply the repository of unattended perceptual inputs, weak memory traces, and the products of early, simple, and automatic cognitive operations.

Neodissociation theory links a diverse set of real-world and laboratory phenomena under a unified descriptive rubric, and challenges cognitive science to account for them. It comes as no surprise that attention can be divided, though that fact in itself poses problems for those information-processing theories which are predicated on the existence of limited-capacity channels or storage structures. But if attention can be divided with one stream of complex, deliberate, cognitive activity proceeding outside awareness, this seems to cause some problems for the way we usually think about things. The empirical base for the theory is sometimes problematic, but the phenomena of dissociation are trying to tell us something about the nature of conscious, subconscious, and unconscious mental processing. If we do not take these phenomena seriously, and consider their implications for our understanding of the cognitive system, our models of the mind may be led seriously astray. This seems reason enough to continue to pursue neodissociation theory, and to incorporate its insights into larger theories, to produce a comprehensive view of the mind in order and disorder.

Acknowledgments

Paper presented at the 4th annual conference of the Cognitive Science Society, Ann Arbor, August 1982. The point of view presented in this essay developed in part from research supported by Grant #MH-35856 from the National Institute of Mental Health, United States Public Health Service. I thank Patricia A. Register and Leanne Wilson for their comments during the preparation of this paper, and Ernest R. Hilgard for promoting the concept of dissociation. An expanded version of this paper, with references, is forthcoming in K. S. Bowers & D. Meichenbaum (Eds.), The unconscious: Several perspectives (Wiley).

Symposium—Representation of Processes and Time

Modeling Events, Actions, and Time

James F. Allen
Department of Computer Science
University of Rochester
Rochester, NY 14627

This brief note concerns what types of knowledge one must possess in order to be able to reason about events and actions. In particular, in comprehending stories or dialogues, many inferences are made based on what events and actions are described. These range from inferences about the temporal ordering of events to inferences concerning the beliefs and motivations of the actors. Here I will concentrate on the nature of events and actions and discuss their relation to temporal reasoning. The references below provide more detail on all these issues.

The formalism for actions and events used in most natural language understanding systems is based on case grammar. Each action is represented by a set of assertions about the semantic roles the noun phrases play with respect to the verb. Such a formalism is a start, but does not explain how to represent what an action actually signifies. If one is told that a certain action occurred, what can one conclude about how the world changed (or didn't change!). One possibility for such a mechanism is found in the work on problem-solving systems (e.g., [Fikes and Nilsson, 1971]), which suggests one common formulation of action. An action is a function from one world state to a succeeding world state and is described by a set of prerequisites and effects, or by decomposition into more primitive actions. While this model is extremely useful for modeling physical actions by a single actor, it does not cover a large class of actions describable in English. For instance, many actions seemingly describe non-activity (e.g., standing still), or acting in some non-specified manner to preserve a state (e.g., preventing your television set from being stolen).

Difficult problems also arise in this model concerning the simultaneous occurrence of actions in domains with more than one agent. For example, consider a simple blocks world with one block and two robots. Let there be two actions, PUSHR, push the block to the right, and PUSHL, push the block to the left. We would like to define the effect of these actions in terms of the block moving. But if the two robots perform a PUSHL and PUSHR simultaneously, the block does not move. Yet, we still want to say that each robot pushed the block. If we cannot express simultaneity of actions, the best we could do to model this situation would be to have the block oscillate as the robots pushed alternately.

The approach suggested here does not attempt to answer what an event or action actually is. Whatever an event is, the only way we can reason about one is by considering how the world changes (or remains constant) during some time interval in which the event occurred. Thus it is crucial that the temporal model in the logic be general enough to capture the scope of possible events. Actions are then defined as a subclass of events that involve agents and are described in a similar manner. The notions of prerequisite, result, and methods of performing actions do not play a central role in this study. While they are important for reasoning about how to attain goals, they don't play an explicit role in defining when an action can be said to have occurred. To make this point clear, consider the simple action of turning on a light.

There are few physical activities that are a necessary part of performing the action of turning on a light. Depending on the context, vastly different patterns of behavior can be classified as the same action. For example, turning on a light usually involves flipping a light switch, but in some circumstances it may involve tightening the light bulb (in the basement), or hitting the wall (in an old house). Although we have knowledge about how the action can be performed, this does *not* define what the action is. The key defining characteristic of turning on the light seems to be that the agent is performing some activity which will cause the light, which is off when the action starts, to become on when the action ends. The importance of this observation is that we could recognize an observed pattern of activity as "turning on the light" even if we had never seen or thought about that pattern previously.

With this model, it is theoretically simple to describe two actions occurring simultaneously. The temporal conditions for each will be asserted to hold over the same time interval. It is then up to the reasoning component to infer any interactions that may arise. While this has not solved anything by itself, at least the complex problem can be expressed in the temporal logic, and reasoning techniques can then be investigated.

With respect to modeling time, I want to make just two basic claims. The first is that representations based on assigning dates for each time are unworkable. The second is that the underlying logic of time should be based on the notion of time intervals rather than time points.

There are many difficulties that arise in systems based on date lines. In such an approach, each time is represented by a value (e.g., a number) and relationships between times can be computed by some operation on the values (e.g., numeric ordering). One problem is that dates are not often supplied. Much temporal information in English is supplied only on a relative basis (e.g., E occurred before E'), both by the explicit mention of such relationships and by tense. For example, in the sentence

"We found the letter while John was away,"

the temporal connective "while" indicates that the time of the find event occurred during the time that John was away, and the past tense indicates that both events occurred in the past (i.e., before now).

The other major difficulty with date-based systems is that there can be considerable uncertainty in our temporal knowledge. For instance, we might know that either event E occurred before event E', or vice versa. But in any case, the times of E and E' did not overlap. One can only capture such information with a partial ordering relationship; no dates can be assigned that capture these constraints. This is not to say that dating is not a useful technique when it is possible, it just cannot be the foundation of the representation.

Turning to the time interval/time point controversy, we can easily observe that both appear to be referred to in English. Thus, we can say,

"We found the letter at 12 o'clock."
"We found the letter yesterday."

The most straightforward approach to dealing with time then seems to be to introduce points in time and then define intervals from those points (e.g., [McDermott, 1981; Bruce, 1972]). I do not use this scheme for two reasons. The first is

that such a representation is too uniform and does not facilitate structuring knowledge in a way convenient for typical temporal reasoning tasks. The second is that it encourages one to think of time as being isomorphic to the real line, which is a serious mistake.

The central issue concerning the first point is the importance of the *during* relation for reasoning. A major part of our temporal knowledge appears to be of the form

"event E' occurred during event E."

Our knowledge of the *during* relation allows a highly structured representation of time. In particular, a common way of inferring that some condition P holds during an interval T is to show that P holds in an interval that contains T. For instance, I might know that my office is locked today because it has been locked all week.

Furthermore, such a *during* hierarchy allows reasoning processes to be localized so that irrelevant facts are never considered. For instance, if one is concerned with what is true "today," one need consider only those intervals that are *during* "today," or above "today" in the *during* hierarchy. If a fact is indexed by an interval wholly contained by an interval representing "yesterday," then it cannot affect what is true now.

On the second issue, some annoying characteristics arise from allowing zero width of time points. For instance, two intervals that meet must either have a point in common or have a point between them. Thus to describe an event consisting of a light being transformed from being off to being on, either the interval where it is off meets the interval where it is on, and thus there is a point where the light is both on and off, or the interval where it is off is strictly before the interval where it is on, and thus there is a point between the two intervals where the light is neither on or off. This can be avoided by a technical trick such as treating all intervals as open on their beginning and closed on their end, but such tricks simply emphasize the unnaturalness of the approach. In an interval-based system, such issues need not arise: two intervals may meet without having any point in common.

Given this interval-based representation of time, what is the equivalent of time points? For instance, we often talk of the beginning or ending times of events. There is no reason to assume, however, that the beginning and ending times are instantaneous points. One might suggest that there is a minimum size e of intervals, such that all intervals of size less than or equal to e are considered to be points. The consequence of this would be that two such point intervals could then only be related by the relations $<$ and $=$. This approach is useful, but only if there is not one fixed value for e, for the size at which an interval is considered to be a point depends on the reasoning task being done. For instance, the smallest time intervals we care about in everyday life are probably of the order of seconds, as physicists or computer scientists, we may consider times on the order of nanoseconds. Thus the interval size that we want to consider as points varies depending on the task as well as the proximity to the current time.

References

Allen, J.F., "An interval-based representation of temporal knowledge," *Proc.*, 7th IJCAI, Vancouver, B.C., 1981.

Allen, J.F., "What's necessary to hide?: Reasoning about action verbs," *Proc.*, 19th Annual Meeting, Assoc. Computational Linguistics, 77-81, Stanford U., 1981.

Bruce, B.C., "A model for temporal references and its application in a question answering program," *Artificial Intelligence 3*, 1972.

Fikes, R.E. and N.J. Nilsson, "STRIPS: A new approach to the application of theorem proving to problem solving," *Artificial Intelligence 2*, 189-205, 1971.

McDermott, D., "A temporal logic for reasoning about processes and plans," Research Report 196, Dept. Computer Science, Yale U., March 1981.

SOME ISSUES ON MECHANISTIC MENTAL MODELS [1]

Johan de Kleer and John Seely Brown
XEROX PARC
Cognitive and Instructional Sciences
3333 Coyote Hill Road
Palo Alto, California 94304

INTRODUCTION

Our long-range goal is to develop a model of how a person acquires an understanding of mechanistic devices such as physical machines, electronic and hydraulic devices, or reactors. We lay out a framework for investigating the structure of what we call *mechanistic mental models*: people's mental models of physical devices. Doing so involves developing a precise notion of a qualitative simulation. The concept of qualitative simulation derives from the common intuition of "picturing in one's mind's eye, how the machine operates."

Although one would intuitively expect qualitative simulations to be simpler than quantitative simulations of a given device, they turn out to be equally complex, but in a different way. These complexities arise, in part, from the fact that devices may appear nondeterministic and underconstrained when the quantities and forces involved in their makeup are viewed solely from a qualitative perspective. Therefore, if the qualitative simulation of the device is to behave deterministically, additional knowledge and reasoning must be used to disambiguate these "apparent" ambiguities.

It is surprisingly difficult to construct mental models of a device that are capable of predicting the consequences of events not considered during the creation of the model. Thus, the process for constructing a good mental model involves a different kind of problem-solving than the process for "running" the resultant mental model, a distinction that we find crucial for understanding how people use mental models. In fact, simply clarifying the differences between the work involved in constructing a qualitative simulation — a process we call *envisioning* — and the work involved in simulating the result of this construction — a process we call *running* — turn out to have both theoretical and practical ramifications.

QUALITATIVE SIMULATIONS

A Basis for Mechanistic Mental Models

Complex devices, such as machines, are built from combinations of simpler devices (components). Let us assume we know the behaviors of the components, as well as the way in which they are connected to form the composite device. The behaviors of the components are described qualitatively, such as "going up" or "going down," "high" or "low." The qualitative simulation always presents the events in the functioning of the machine in their causal order. Figure 1 illustrates a conventional door-buzzer (for the moment ignoring the button that activates the buzzer). The buzzer is a simple device, but complex enough to use for illustrating ideas of qualitative simulation.

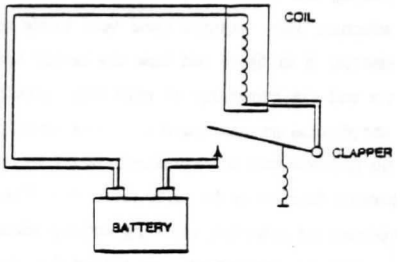


Figure 1 : Buzzer

The buzzer's qualitative simulation might be described as: *The clapper-switch of the buzzer closes, which causes the coil to conduct a current, thereby generating an electromagnetic field which in turn pulls the clapper arm away from the switch contact, opening the switch, shutting off the magnetic field, allowing the clapper arm to return to its closed position, and thereby start the whole process over again*[2]

The simplicity of the qualitative simulation as expressed in the preceding example is deceptive. Qualitative simulation encompasses a variety of ideas which need to be carefully differentiated. For example, we must distinguish simulation as a process from the results of that process. A simulation process operates on a representation describing the device, producing another representation that describes how the device functions. One source of confusion is that this latter representation can likewise be "interpreted" or simulated, but doing so will produce very little more than what is already explicitly represented in the functional representation produced by the first kind of simulation[3]

We need to distinguish four related notions which form the basic distinctions for a theory of qualitative reasoning. The most basic, *device topology*, is a representation of the structure of the device (i.e., of its physical organization). For example, the steam plant's structure consists of a steam generator, turbine, condenser, their connecting pipes, etc. The second, *envisioning*, is an inference process which, given the device's structure, determines its function. The third, *causal model*, describes the functioning of the device (i.e., a description of how the device's behavior results from its constituent components which is stated in terms of how the components causally interact). The last is the *running* of the causal model to produce a specific behavior for the device, by giving a chain of events each causally related to the previous one. Thus, both the structure and functioning of a device are represented by some knowledge-representation scheme

[1] This paper is an abridged and revised version of de Kleer & Brown [82].

[2] The repetitive opening and closing of the switch (i.e., its vibration) produces an audible sound.

[3] Note that this latter kind of simulation is just one of the kinds of inference mechanisms that can use or "interpret" the functional representation. Others can inspect it in order to answer such questions as "Could x cause y to happen?"

(device topology and causal model, respectively), with the former being the input to the envisioning process and the latter being its output; this output causal model is, in turn, then used in the running. The example of qualitative simulation presented earlier is ambiguous as to whether it refers to the envisioning, the causal model, or the running.

Envisioning, i.e., determining the functioning of a device solely from its structure often requires some very subtle reasoning. The task, in essence, is to figure out how the device works given only its structure and the knowledge of some basic principles. Structure describes the physical organization of the device, namely the constituent components and how they are connected, but it does not describe how the components function in the particular device. The "behaviors" of each component are described assuming nothing about the particular context in which the component is embedded (i.e., the description is context-free). These behaviors form a component model (or schema) which characterizes all the potential behaviors of the component; the envisioning process instantiates a specific behavior for each component from these models. These component models are the basic principles which the envisioning process draws upon to derive the functioning from the structure.

To determine the functioning of the overall device, each component's model must be examined and an individual, specific behavior instantiated for it. Thus, the functioning of the entire device is determined, in part, by "gluing together" the specific behaviors of all of its components. The problem for envisioning is determining for each component which behavior, given all the possible behaviors its model characterizes, is actually being manifested.

What makes the problem-solving effort involved in the structure-to-function inference process difficult is that the behavior of the overall device is constrained, not only by local interactions of its component behaviors, but also by global interactions. Therefore, in principle, the behavior models of the components which are specified qualitatively may not provide enough information to identify the correct functioning of the device. For example, if values are described qualitatively, often fine-grained distinctions cannot be made between them. Thus, in the case of the buzzer, the envisioning may not be able to determine which is greater, the force of the magnetic field or the restoring force of the spring. Knowing which is greater may, in fact, be crucial to deducing the correct functioning of the device.

In order to describe how the resultant behavior derives from the behaviors of the constituents, first, each important event in the overall behavior must be causally related to preceding events. Then, each causal relationship must be explained by some fragment of the component model of one of its components. The example describing Figure 1, is, at best, an abridged description of the buzzer's function. It causally relates each event to the preceding one, but fails to state any rationale for these causal connections. Because it is impossible to tell, a priori, whether the component models lead to unique behavior, the problem-solver must entertain the possibility that the structural evidence is underconstraining. Therefore the envisioning must take into account the possibility that one structure may have multiple possible functionings among which the envisioning cannot, in principle, distinguish.

"Running" the resulting causal model is closest to the original psychological intuition of "picturing, in one's mind's eye, how the machine operates." By running the model, one, in essence, does a straightforward simulation of the machine; the running itself does not have to determine or "prove" the causal or temporal ordering of events, as the envisioning process already has done so, and encoded the information in the causal model which serves as the input data for the running process.

The simplicity and elegance of the running process is the result of the complex problem-solving (i.e., envisioning) that constructed it. That our intuition that "picturing, in one's mind's eye, how the machine operates" is simple, is manifested by this running process. However, that sense of simplicity is deceptive, for the running is not possible without the more complex problem-solving which preceded it, removing all the ambiguities about how the machine *might* be functioning.

Understandably, the problems that arise in constructing causal models and the mechanisms that suffice in solving these problems are important for cognitive psychology and artificial intelligence. For psychology, they are important because they provide a framework for analyzing the "competency" involved in determining how a novel machine functions. Inasmuch as envisioning is restricted to being based solely on structural evidence, it becomes an interesting inference strategy in its own right for artificial intelligence applications, especially given the desire for artificial intelligence systems to be robust, and to be capable to deal with novel situations. The resulting models are more likely to be void of any implicit assumptions or built-in presuppositions based on how the device was intended to behave.

AMBIGUITIES AND ASSUMPTIONS

Origin of Ambiguities

In general, ambiguities originate from the fact that the information available to the qualitative analysis underdetermines or only partially characterizes the actual behavior of the overall device. There are three reasons for this underdetermination. The first and most obvious is that the quantities referenced by the component models are qualitative and thus fine-grained distinctions cannot be made between the attribute values or component states. Second, because the implicit time progression in the simulation is qualitative, it is not always possible to determine the actual ordering of events. And the third reason, not directly related to the qualitative nature of the models, comes from the limitations on the kinds of information captured by the models. Because envisioning tries to identify a global flow of action by piecing together local cause-effect rules of the component models, a component model encodes only those aspects of the component's behavior that can be used in such a fashion. However, our understanding of a given component often involves more knowledge than is (or, perhaps, could be) encoded in such mechanistic rules. For example, in modeling the internal operation of a pump we know from the laws of physics that fluid is conserved in passing through the pump. But, because this piece of knowledge is a *constraint*, it cannot be represented by

any cause-effect rule: the inability to encode it can lead to a given component model being underdetermined.

Origin of Assumptions

In the buzzer example, because of the qualitative nature of the attribute values, the envisioning process cannot determine whether the spring is stronger than the magnetic field. In this "impasse," it is forced to consider two hypothetical situations: one in which it *assumes* the spring is stronger than the magnetic field and one in which it *assumes* the spring is weaker than the field.

Impasses occur when envisioning cannot evaluate a transition condition (e.g., the condition of the switch being open) or invoke an attribute equation (e.g., that of field strength being proportional to coil current) to determine the value of an unknown attribute. In order to proceed around impasses, the envisioning must introduce assumptions about the truth or falsity of conditions or about the values of unknown attributes.

The buzzer example can be used to illustrate an impasse which arises from the envisioning being unable to determine whether a transition condition holds. In this impasse, the envisioner introduces an assumption that the condition "force from the coil > restoring force of the spring" is true, and then proceeds to analyze the new resulting state. Of course, the resulting causal model will then contain two accounts of the device's functioning: one in which the clapper rises and one in which it does not. Additional knowledge and reasoning strategies must then be used to verify or reject the various assumptions that were created to enable the envisioner to proceed around such impasses. These strategies combined with a much more extensive analysis of the kinds of assumptions needed in order to construct a causal model have been detailed in the expanded version of this paper.

REFERENCES

de Kleer, J. and J.S. Brown, "Assumptions and Ambiguities in Mechanistic Mental Models," to appear in *Mental Models*, edited by D. Gentner and A. S. Stevens, Erlbaum, 1982.

A Note Concerning Qualitative Process Theory

Ken Forbus
MIT AI Lab
545 Technology Square
Cambridge, Mass. 02139 USA

1. Introduction

Many kinds of changes occur in physical situations. Things move, collide, flow, bend, heat up, cool down, stretch, break, and boil. These and the other things that happen to cause changes in objects over time are intuitively characterized as processes. Much of formal physics consists of characterizations of processes by differential equations which describe how the parameters of objects change over time. But the notion of process is richer and more structured than this. We often reach conclusions about physical processes based on very little information. For example, we know that if we heat water in a sealed container the water can eventually boil, and if we continue to do so the container can explode. To understand common sense physical reasoning we must understand how to reason qualitatively about processes, their effects, and their limits. I have been developing a theory, called Qualitative Process theory, for this purpose[Forbus, 1981, 1982]. I expect this theory, when fully developed, to provide a representational framework for understanding human common sense physical reasoning. It should also be useful for constructing computer programs that reason about complex physical systems as well as common sense reasoning. Programs that explain, repair and operate complex systems such as nuclear power plants and steam machinery will need to draw the kinds of conclusions this theory sanctions.

Qualitative reasoning about quantities is a problem that has long plagued Artificial Intelligence and Cognitive Science. Many schemes have been tried, including simple symbolic vocabularies (TALL, VERY TALL, etc.), real numbers, intervals, fuzzy logic, and so forth. None are very satisfying. The reason is that none of the above schemes makes distinctions that are relevant to physical reasoning. Reasoning about processes provides a strong constraint on the choice of representation for quantities. Processes usually start and stop when orderings between quantities change. For example, when two objects with unequal temperatures are brought into contact there will be a heat flow from one to the other which will stop when the temperatures are equal. In Qualitative Process theory the value of quantities are represented by a partial ordering of other quantities determined by the domain physics. The representation appears both useful and natural.

QP theory is mainly concerned with the form of physical theories and only indirectly about their specific content. For example, heat flow processes which don't conserve energy and transfer "caloric fluid" can be written as well as the classical physical description. Newtonian, Aristotelian, and Impetus theories of motion can all be encoded. Thus QP theory provides a language for writing physical theories. In particular, the primitives are simple processes (such as flows, state changes, and motion), the means of combination are sequentiality and shared parameters, and the means of abstraction are naming these combinations, including encapsulating a piece of the process history (a kind of behavioral description, see [Hayes, 1979]) for the situation as a new process.

The basic Qualitative Process theory is not intended to capture the full range of qualitative reasoning about the physical world. Instead it is concerned with describing the weakest kind of information that still allows useful conclusions to be drawn. There are two reasons why this weak level of description is interesting. First, conclusions from weak information are often required to drive the search for conclusions from more detailed information (an illustration is [deKleer, 1975]). More importantly, I believe that the basic theory can be used to write what corresponds to people's common sense physical knowledge. To capture more sophisticated kinds of physical reasoning (for example, how an engineer makes estimates of circuit parameters or stresses on a bridge) extension theories containing more detailed representations of quantity, functions, and processes will be needed. Examples of extension theories could include order of magnitude estimates and numbers. By providing a shared basic theory, future studies of more sophisticated domains may yield a way to classify kinds of physical reasoning according to the extension theories they require.

2. An Example

There are several kinds of reasoning that can be performed using Qualitative Process theory, including reasoning about the limits of processes ("What might happen if this valve is left open?") and consequences of alternate situations ("How would the turning up the stove affect the heating of the kettle?") as well as explaining some problems involved in causal reasoning. Several examples of common sense phenomena have been examined in this context, including modelling a boiler, motion, materials (saying that you can push with a string but not pull with it), and an oscillator. An informal example will illustrate its flavor. Here is a simple problem involving physical systems that we solve easily:

Imagine looking at a large tank, partially filled with water. You can see two pipes leading into it, and you note that the level in the tank is dropping. Your goal is to figure out why this is happening.

In QP theory terms, "why this is happening" means finding a set of processes which are causing the changes in the situation. (In the complicated physical systems which comprise much of our technology, this is much harder than the simple example depicted here, because the relationship between what we can observe (through instruments) and the processes which serve as an explanation is much less direct). The reasoning goes as follows:

[1] No process affects level directly, but level is qualitatively proportional to Amount-of fluid.

[2] The only processes which affect Amount-of a contained fluid are boiling, evaporation, and fluid flow.

[3] No heat source is visible, so boiling can be ruled out.

[4] The time scale is short, so evaporation can be ruled out.

[5] By exclusion, fluid flow must be the source of the influence.

[6] Fluid flow requires a fluid path.

[7] Only two pipes are visible, so assume those are the only fluid connections to the tank.

[8] Only two fluid flows are possible, one through each pipe. Fluid flow can be measured; in this case both flows are into the tank.

[9] Therefore the influence of the fluid flows is positive.

[10] Therefore the level of the tank should be increasing, not decreasing.

[11] Either (1) Other processes affecting amount-of exist
(2) Evaporation or Boiling are occuring
(3) Measurements are wrong
(4) Other fluid paths exist

[12] Pragmatically, (4) is the most likely - e.g., a large leak in the tank.

Knowing what can be measured and the pragmatic information used in ruling out evaporation and in accepting the leak as the best prospect are not part of QP theory, but instead illustrate the interaction of the theory with other kinds of world knowledge. Note that the key to the deduction is the assumption of a finite vocabulary of processes that could cause the observed change. Hayes [Hayes, Liquids] suggests reasoning by elimination is a powerful technique in common sense reasoning; organizing physical knowledge around a vocabulary of processes provides further opportunity to do so.

3. Current State of the Theory

The current state of the theory is described in [Forbus, 1982]. Further theoretical developments are being carried out in the context of reasoning about simple fluid and mechanical systems. An

implementation is underway.

4. References

Clement, John "A Conceptual Model Discussed by Galileo and Used Intuitively by Physics Students" to appear in Mental Models, D. Gentner and A. Stevens, editors.

deKleer, Johan "Qualitative and Quantitative Knowledge in Classical Mechanics" TR-352, MIT AI Lab, Cambridge, Massachusetts, 1975

Forbus, K. "Qualitative Reasoning about Physical Processes" Proceedings of IJCAI-7, 1981

Forbus, K. "Qualitative Process Theory" MIT AI Lab Memo No. 664, February, 1982

Hayes, Patrick J. "Naive Physics 1 - Ontology for Liquids" Memo, Centre pour les etudes Semantiques et Cognitives, Geneva, 1979

McCloskey, M. "Naive Theories of Motion" to appear in Mental Models, D. Gentner and A. Stevens, editors.

Symposium—Metaphor

The Preconceptual Basis of Experiential Metaphor

Mark Johnson
Department of Philosophy
Southern Illinois University at Carbondale

Standard models of metaphoric comprehension share at least the following set of basic assumptions: (1) Meaning is conceptual structure. (2) Comprehending a metaphor of the form "A is B" requires a grasp of the appropriate conceptual structure for the "A" and "B" (topic-vehicle) components, and it also requires the ability to map the B domain onto the A domain in a contextually appropriate fashion. (3) The mapping or projection procedure depends principally on underlying similarities between the two domains. Versions of this position differ as to the nature of the mapping mechanism. Some treat the metaphoric projection as a simple transfer of discrete properties or relations from the B domain over to the A domain, with appropriate changes being made to apply the transferred predicates to the new domain. Others argue that a more complex model is needed, one in which the entire system of predicates for the B domain, with all of its complex internal relations, must somehow be projected as a whole in such a way as to restructure the conceptual system for the A domain.

It is commonly believed by those who operate with some version of this standard model that the chief problem posed by metaphor for artificial intelligence is to discover the way in which contextual clues determine the precise nature of the projective process of metaphoric understanding. While I agree that this is the main difficulty, I want to suggest that it is less amenable to solution than most cognitive scientists believe. The reason for my pessimism is that, contrary to the accepted view, understanding a metaphor is not just a process of grasping certain conceptual structurings. In the metaphors of ordinary and technical discourse alike, there is also a preconceptual basis in experience that gives the metaphor the meaning it has and that cannot be reduced to concepts or conceptual structure (as mental representations).

My argument is based upon an analysis of some of the preconceptual factors involved in the comprehension of what I call "experiential" metaphors. An experiential metaphor is a process of experiencing, conceptualizing, and talking about one domain of experience as it is structured in terms of another domain of a different kind. Such metaphors are basic processes of everyday experience, and they are not mere linguistic ornaments or rhetorical modes of expression. The experiential metaphor MARRIAGE IS A BUSINESS PARTNERSHIP, for example, is one of several metaphors in American culture that structures the way some people understand, act out, and reason about their marriages. It is not a matter of mere words that we use to talk about marriage; rather, it is one possible structuring of marital relations that provides coherence, order, and significance in the lives of those who live by the metaphor.

But the MARRIAGE IS A BUSINESS PARTNERSHIP metaphor is more than a conceptual structuring of some aspects of one's marriage. It involves non-structural, preconceptual elements without which the metaphor would have no significance for us. These preconceptual elements in experience consist of various capacities, skills, values, and purposes in which the conceptual structures are rooted and from which they take their nourishment. With reference to the BUSINESS PARTNERSHIP metaphor I identify four such elements: (1) General human purposes, (2) Cultural institutions and practices, (3) Theoretical paradigms, (4) Individual characteristics and patterns (including (i) individual purposes, (ii) individual tastes and values, and (iii) personality traits).

I am claiming that understanding a metaphor involves more than grasping conceptual structure—it also involves preconceptual elements that are neither discrete predicates nor structured relations. Such elements are a basic part of our ordinary experience without which no metaphor could have the power it does to shape our understanding, action, and language. If this analysis is correct, it calls for a rethinking of certain fundamental assumptions guiding work on metaphor in cognitive science.

Towards a Computational Model of Metaphor in Common Sense Reasoning

Jaime G. Carbonell
Carnegie-Mellon University
Pittsburgh, PA 15213

1. Introduction

The theory that metaphor dominates large aspects of human thinking, as well playing a significant role in linguistic communication, has been argued with considerable force [10, 8, 3, 1]. However, the validity of such a theory is a matter of continuing debate that appears neither to dissuade its proponents nor convince its detractors. Being among the proponents, I propose to develop a computationally effective, common sense reasoning system based on underlying metaphors. I claim that if such a system exhibits cognitively plausible common sense reasoning capabilities, it will demonstrate the utility of metaphorical reasoning. Moreover, if the model can account for observed instances of naive human reasoning better than existing inference systems, it will provide convincing evidence in favor of the metaphorical reasoning theory. This brief paper investigates aspects of the metaphorical reasoning phenomenon and describes the initial steps towards developing a computational model.

2. Experiential Reasoning vs Formal Systems

Humans learn from experience to a degree that no formal system, AI model, or philosophical theory can match. The statement that the human mind is (or contains) the sum total of its experiences is in itself rather vacuous. A more precise formulation of experience-based reasoning may be structured in terms of coordinated answers to the following questions: *How* are experiences brought to bear in understanding new situations? *How* is long term memory modified and indexed? *How* are inference patterns acquired in a particular domain and adapted to apply in novel situations? *How* does a person "see the light" when a previously incomprehensible problem is viewed from a new perspective? *How* are the vast majority of irrelevant or inappropriate experiences and inference patterns filtered out in the understanding process? Answering all these "how" questions requires a ***process model*** capable of organizing large amount of knowledge and mapping relevant aspects of past experience to new situations. Some meaningful starts have been made towards large-scale episodic-based memory organization [14, 15, 12, 9] and towards episodic-based analogical reasoning [5, 4, 2]. Bearing these questions in mind, I turn towards the issue of common sense reasoning in knowledge-rich mundane domains.

My central claim is that reasoning in mundane, recurrent situations is qualitatively different from reasoning in more abstract and experientially unique situations (such as some mathematical or puzzle-solving domains). The former consists of recalling appropriate past experiences and inference patterns, whereas the latter requires knowledge-poor search processes more typical of past and present AI problem solving systems. Since computer programs perform much better in simple, elegant, abstract domains than in "scruffy" experience-rich human domains, it is evident that a fundamental reasoning mechanism is lacking from the AI repertoire. The issue is not merely that AI systems lack experience in mundane human scenarios -- they would be unable to benefit from such experience if it were encoded in their knowledge base. I postulate that the missing reasoning method is one of metaphor-based transfer of proven inference patterns and experiential knowledge across domains. This is not to say that humans are largely incapable of more formal reasoning, but rather that such reasoning is seldom necessary and when applied requires a more concerted cognitive effort than mundane metaphorical inference.

3. Towards Metaphorical Reasoning: The Balance Metaphor

Consider a prevalent metaphor: reasoning about imponderable or abstract entities as though they were objects with a measurable weight. One of several reasoning patterns based on this simple metaphor is the *balance principle*. The physical analog of this reasoning pattern is a prototypical scale with two balanced plates. Large numbers of metaphors appeal to this simple device coupled with the processes of bringing the system into (and out of) equilibrium. First, consider some examples of the basic metaphor, in which *the relevant aspect of an abstract concept maps onto the weight*[1] *of an unspecified physical object.*

> Arms control is a ***weighty*** issue.
>
> The worries of a nation ***weigh heavily*** upon his shoulders.
>
> The Argentine air force launched a ***massive*** attack on the British fleet. One frigate was ***heavily*** damaged, but only ***light*** casualties were suffered by British sailors. The Argentines payed a ***heavy*** toll in downed aircraft.
>
> Not being in the mood for ***heavy*** drama, John went to a ***light*** comedy, which turned out to be a piece of meaningless ***fluff***.
>
> Pendergast was a real ***heavyweight*** in the 1920s Saint Louis political scene.
>
> The crime ***weighed heavily*** upon his conscience.
>
> The ***weight*** of the evidence was overwhelming.

Weight clearly represents different things in the various metaphors: the severity of a nation's problems, the number of attacking aircraft, the extent of physical damage, the emotional affect on audiences of theatrical productions, the amount of political muscle (to use another metaphor), the reaction to violated moral principles, and the degree to which evidence is found to be convincing. In general, more is heavier; less is lighter. One may argue that since language is heavily endowed with words that describe weight, mass and other physical attributes (such as hight and orientation [10]), one borrows such words when discussing more abstract entities [13] -- for lack of alternate vocabulary. Whereas this argument is widely accepted, it falls far short of the conjecture I wish to make.

> **Conjecture:** *Physical metaphors directly mirror the underlying inference processes. Patterns of inference valid for physical attributes are mapped invariant and reinstantiated in the target domain of the metaphor.*

In order to illustrate the validity of this conjecture consider a common inference pattern based on the weight of physical

[1] Mass is virtually synonymous with weight in naive reasoning.

objects: The inference pattern is the *balance principle* mentioned earlier as applied to a scale with two plates. The scale can be in balance or tipped towards either side, as a function of the relative weights of objects placed in the respective plates. Inference consists of placing objects in the scale and predicting the resultant situation -- no claim is made as to whether this process occurs in a propositional framework or as visual imagery, although I favor the former. How could such a simple inference pattern be useful? How could it apply to complex, non-physical domains? Consider the following examples of metaphorical communication based on this inference pattern:

> The jury found the ***weight*** of the evidence favoring the defendant. His impeccable record ***weighed heavily*** in his favor, whereas the prosecution witness, being a confessed con-man, carried ***little weight*** with the jury. ***On balance*** the state failed to ***amass*** sufficient evidence for a ***solid*** case.
>
> The SS-20 missile ***tips the balance*** of power in favor of the Soviets.
>
> Both conservative and liberal arguments appeared to ***carry equal weight*** with the president, and his decision ***hung on the balance***. However, his long-standing opposition to abortion ***tipped the scale*** in favor of the conservatives.
>
> The Steelers were the ***heavy*** pre-game favorites, but the Browns started ***piling up*** points and accumulated a ***massive*** half-time lead. In spite of a late rally, the steelers did not score ***heavily*** enough to pull the game out.
>
> The job applicant's shyness ***weighed*** against her, but her excellent recommendations ***tipped the scales*** in her favor.

In each example above the same basic underlying inference pattern recurs, whether representing the outcome of a trial, statements of relative military power, decision-making processes, or the outcome of a sporting event. The inference pattern itself is quite simple: it takes as input signed quantities -- whose magnitudes are analogous to their stated "weight" and whose signs depend on which side of a binary issue those weights correspond -- and selects the side with the maximal weight, computing some qualitative estimate of how far out of balance the system is. Moreover, the inference pattern also serves to infer the rough weight of one side if the weight of the other side and the resultant balance state are known. (E.g., If Georgia won the football game scoring only 17 points, Alabama's scoring must have been ***really light***)

The central issue in my discussion is that this very simple inference pattern based on a physical metaphor accounts for very large numbers of inferences in mundane human situations. Given the existence of such a simple and widely applicable pattern, why should one suppose that more complicated inference methods explain human reasoning more accurately? It is my belief that there exist a moderate number of general inference patterns such as the present one, which together span most mundane human situations. Moreover, the few other patterns I have found thus far are also rooted on simple physical principles or other directly experienced phenomena. However, since the current study is only in its initial stages, the hypothesis that metaphorical inference predominates human cognition retains the status of a conjecture, pending additional investigation. I would say that the weight of the evidence is as yet insufficient to tip the academic scales.

4. Requirements on a computational model

Metaphorically-based general patterns of inference do not appear confined to naive reasoning in mundane situations. Gentner [7] and Johnson [8] have argued the significant role that metaphor plays in formulating scientific theories. In our preliminary investigations, Larkin and I [11] have isolated general inference patterns in scientific reasoning that transcend the traditional boundaries of a science. For instance, the notion of equilibrium (of forces on a rigid object, or of ion transfer in aqueous solutions, etc.) is, in essence, a more precise and general formulation of the balance metaphor. Reasoning based on recurring general inference patterns seems to pervade every aspect of human cognition. These patterns encapsulate sets of rules to be used in unison, and thereby bypass the combinatorial problems in traditional rule-based deductive inference. The inference patterns are frozen from experience and generalized to apply in many relevant domains.

I have started working on a computational model that acquires and generalizes recurring inference patterns from prior experience [6], but let us focus on the equally basic issue of how such patterns may be used in the reasoning process. Conceptually, the process may be divided into three stages:

1. Index the relevant inference patterns appropriate to the situation at hand. The establishment of the appropriate metaphor is the really difficult part. This is why it is much easier to understand someone's description of observed or experienced events (the metaphor is explicitly referenced by the choice of words), than to generate appropriate action -- the typical distinction between planning and plan comprehension.
2. Instantiate the inference patterns in the specific situation. Computationally, the process of instantiation and the process of searching for appropriate inference patterns are two aspects of the same mechanism.
3. Carry out the inferences stipulated in the retrieved patterns, and check whether additional inference patterns are invoked as a result of the expanded knowledge state.

At the present stage in the investigation, I am searching for general inference patterns and the metaphors that give rise to them, both in mundane and in scientific scenarios. As these patterns are discovered, they are cataloged according to the situational features that indicate their presence. The basic metaphor underlying each inference pattern is recorded along with exemplary linguistic manifestations. The internal structure of the inference patterns themselves are simple to encode in an AI system. The difficulty arises in connecting them to the external world (i.e., establishing appropriate mappings) and in determining the conditions of applicability for each inference pattern (which are more accurately represented by continuous functions than simple binary tests). For instance, it is difficult to formulate a general process capable of drawing the mapping between the "weight" of a hypothetical object and the corresponding aspect of the non-physical entity under consideration, so that the balance inference pattern my apply. It is equally difficult to determine the degree to which this or any other inference pattern can make a useful contribution to novel situations that bear sufficient similarity to past experience [4].

5. Future Directions

If one lends credence to the metaphorical reasoning hypothesis, several avenues of continued research suggest themselves.

- Continue the development of a computational model to test the theory of metaphorical inference and thereby force a finer-grain analysis of the phenomenon.
- Examine the *extent* to which linguistic metaphors reflect underlying inference patterns. The existence of a number generally useful inference patterns based on underlying metaphors is not incompatible with the possibility that the vast majority of metaphors remain mere linguistic devices, as previously thought. In essence, the existence of a phenomenon does not necessarily imply its universal

presence. This is a matter to be resolved by more comprehensive future investigation.

- Investigate the close connection between models of experiential learning and metaphorical inference. In fact, my earlier investigation of patterns of analogical reasoning in learning problem solving strategies first suggested that the inference patterns that could be acquired from experience coincide with those underlying many common metaphors [4, 3].

- Exploit the human ability for experientially-based metaphorical reasoning in order to enhance the educational process. In fact, Sleeman and others have independently used the *balance metaphor* to help teach algebra to young or learning disabled children. Briefly, a scale is viewed as an equation, where the quantities on the right and left hand sides must balance. Algebraic manipulations correspond to adding or deleting equal amounts of weight from both sides of the scale, hence preserving balance. First, the child is taught to use the scale with color-coded boxes or different (integral) weights. Then, the transfer to numbers in simple algebraic equations is performed. Preliminary results indicate that children learn faster and better when they are able to use explicitly this general inference pattern. I foresee other applications of this and other metaphorical inference patterns in facilitating instruction of more abstract concepts. The teacher must make the mapping explicit to the student in domains alien to his or her past experience. As discussed earlier, establishing and instantiating the appropriate mapping is also the most problematical phase from a computational standpoint, and therefore should correspond to the most difficult step in the learning process.

6. References

1. Burstein, M. H., "Concept Formation Through the Interaction of Multiple Models." *Proceedings of the Third Annual Conference of the Cognitive Science Society*, 1981

2. Carbonell, J. G., "A Computational Model of Problem Solving by Analogy." *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, August 1981 , pp. 147-152.

3. Carbonell, J. G., "Metaphor: An Inescapable Phenomenon in Natural Language Comprehension," in *Knowledge Representation for Language Processing Systems*, W. Lehnert and M. Ringle, eds., New Jersey: Erlbaum, 1982.

4. Carbonell, J. G., "Learning by Analogy: Formulating and Generalizing Plans from Past Experience," in *Machine Learning*, R. S. Michalski, J. G. Carbonell and T. M. Mitchell, eds., Palo Alto, CA: Tioga Pub. Co., 1982.

5. Carbonell, J. G., "Invariance Hierarchies in Metaphor Interpretation." *Proceedings of the Third Meeting of the Cognitive Science Society*, August 1981 , pp. 292-295.

6. Carbonell, J. G., "Acquiring Problem Solving Skills by Analogy," *Proceedings of the Second Meeting of the American Association for Artificial Intelligence*, 1982 , Pittsburgh.

7. Gentner, D., "The Structure of Analogical Models in Science," Tech. report 4451, Bolt Beranek and Newman, 1980.

8. Johnson, M., "Metaphorical Reasoning," 1982. Unpublished manuscript.

9. Kolodner, J. L., *Retrieval and Organizational Strategies in Conceptual Memory: A Computer Model*, PhD dissertation, Yale University, Nov. 1980.

10. Lakoff, G. and Johnson, M., *Metaphors We Live By*, Chicago University Press, 1980.

11. Larkin, J. H. and Carbonell, J. G., "General Patterns of Scientific Inference: A Basis for Robust and Extensible Instructional Systems," 1982. Proposal to the Office of Naval Research.

12. Lebowitz, M., *Generalization and Memory in an Integrated Understanding System*, PhD dissertation, Yale University, Oct. 1980.

13. Ortony, A. (Ed.), *Metaphor and Thought*, Cambridge University Press, 1979.

14. Schank, R. C., "Reminding and Memory Organization: An Introduction to MOPS," Tech. report 170, Yale University Comp. Sci. Dept., 1979.

15. Schank, R. C., "Language and Memory," *Cognitive Science*, Vol. 4, No. 3, 1980 , pp. 243-284.

METAPHORS FOR MARRIAGE IN OUR CULTURE

Naomi Quinn

Duke University

Lakoff and Johnson, in Metaphors We Live By (1980), frequently allude to metaphors "in our culture." This paper explores the way in which culture can be said to constrain metaphorical thinking in one domain, that of American marriage. It undertakes systematic analysis of a sample of metaphors used by 22 American interviewees, spouses in 11 marriages, over an average of 15-16 hour-long interviews per individual. Superficially, the particular metaphorical expressions used by a given individual would seem to vary widely. But these expressions can be shown to cluster around one or another of a small number of underlying metaphorical models to which that individual consistently returns. Thus a husband who conceptualizes his marriage as BUILDING A DURABLE PRODUCT is able to express this underlying metaphorical model in terms of a number of different concrete products or general types of products, sometimes switching from product to product in a single utterance: a metal in "we forged a lifetime proposition," an unspecified construction of the sort one might build in one's home workshop in "marriage is a do-it-yourself project," something capable of structural improvement in "our marriage was strengthened," an edifice in "we made that the cornerstone," once again an edifice and then something made out of a malleable material, perhaps clay, in "they had a basic solid foundation in their marriages that could be shaped into something good," and something like a car built out of cannibalized parts, which then takes on the properties of a chemical such as epoxy glue in "we have both looked into the other person and found their best parts and used these parts to make the relationship gel." Another husband who conceptualizes marriage as TRAVEL sometimes speaks of the marriage as a train or trolley capable of "getting off the track," other times as foot travel in "he's running the same path I was before I got married." and "If I weren't married I'd be running down the same line," and still elsewhere as some kind of maneuverable vehicle in "I just observe others' marriages and try to run mine down the middle."

Moreover, analysis reveals that the vast majority of these stable metaphorical models themselves fall into two broad classes: metaphors of marriage as some kind of effortful activity--e.g., WORK, BUILDING A DURABLE PRODUCT, A QUEST, AN INVESTMENT, GROWTH, A STRUGGLE, A JOURNEY, TRAVEL --or marriage as some kind of dual relationship--e.g., A PARTNERSHIP, TWO PATHS CROSSING, MUTUAL PARENTING, BEING A UNITED FRONT, BEING A PAIR, BEING ONE PERSON, BEING A COUPLE, A SPATIAL RELATIONSHIP. Thus the superficially variable metaphors which interviewees employ can be seen to be highly constrained. What constrains them is apparently some kind of (still deeper) underlying folk theory about the nature of the marital relationship, which says that such an enduring attachment between two people takes effort to achieve or insure.

Individuals are free to conceptualize their marriages in terms of any kind of experience drawn from either or both of the two classes, dual relationship and effortful activity. They may also choose to foreground metaphor from one of these to the neglect of the other--understanding their marriage primarily or entirely in terms of the nature of the relationship they have, or alternatively, in terms of the effort involved in sustaining that relationship. However, given individuals are very likely to employ metaphorical models of both classes. When this is done, the metaphorical model selected from the class of effortful activities matches an entailment of the metaphorical model which characterizes the nature of the relationship. The mapping is one of goal implementation: that is, given some entailment of the relationship conveyed in one metaphorical model, how can such an entailment plausibly be implemented? Thus, for example, the husband with the model of marriage as BUILDING A DURABLE PRODUCT (an effortful activity) is implementing, in this enterprise, the goal of making a permanent marriage. Permanence is entailed by BEING A COUPLE (a dual relationship), a metaphorical model central in his thinking about the nature of his marriage and others he knows (Quinn 1981). For this husband, being a couple entails being permanently coupled together, hence "durably built."

The husband who regards marriage as TRAVEL (an effortful activity) means, by keeping his marriage on the track and running it down the middle, that he regulates the proportion of time he and his wife spend together and apart, or as he puts it, the proportion of time their "paths run together" and "run apart." His metaphorical model for their marital relationship is one of TWO PATHS CROSSING (a dual relationship). Achieving a balance between "crossed" time and separate time, which he views as the central entailment of marriage as TWO PATHS CROSSING, he then conceptualizes in terms of the necessity to stay on the path, keep on the track, and steer a correct course.

Still another husband views his marriage as a SPATIAL RELATIONSHIP (a dual relationship) in which spouses must be "pretty clear where each of us are" in some kind of uncharted territory, and "try to get a good sense of where we are" or else "we might satellite far enough away so we're not sure what's in between us" in what appears to be outer space. If two people are constantly shifting position vis-a-vis one another, as in this SPATIAL RELATIONSHIP model of marriage, the overriding concern is to keep in contact. This is met, by this husband, with an INQUIRY model (effortful activity), which involves "space to kind of work out where each of us were," "a lot of searching," "miles of talking," and "communicating." These husbands all seem to agree that marriage requires some effortful activity. The particular effort required depends upon a prior conceptualization of the nature of the relationship itself. In each case, the metaphorical model of the relationship is problematic in some entailment, the problem solution becomes a goal of the marriage, and a solution for this marital goal is couched in a further metaphorical model. Whether this view of marriage as problem and solution is universal cross-culturally, or whether it is a distinctively American way of viewing marriage, perhaps certain other relationships, and even other aspects of life, I can only speculate.

What do these observations suggest for a theory of metaphorical understanding? First, ongoing metaphorical understandings are relatively stable

and these stable understandings are based in underlying metaphorical models. Metaphorical expressions which instantiate a given model can be varied at will to take advantage of different properties of concrete objects or events. Thus MARRIAGE IS BUILDING A DURABLE PRODUCT allows its user to understand his marital experience in terms of conernstones, reassembled parts, and the gelling process with equal facility.

Second, these underlying metaphorical models themselves cannot be anything at all. They are constrained to members of those classes which are culturally appropriate source domains for the target experience (to use Carbonell's [1981] terms). Members of a culture share knowledge of these appropriate source domains. A considerable economy of learning and memory is achieved in this organization of cultural knowledge by metaphorical class that would be lost if target experiences were assigned directly to culturally permissible metaphors or metaphorical models. Within classes, an individual has latitude in selecting whatever metaphorical model does the best job of characterizing, for that person, the target experience.

Third, underlying metaphorical models cannot be studied in isolation from one another. In ongoing understanding, they frequently bear relationships to one another. Here I have given an example of metaphorical models which are mapped onto entailments of other metaphorical models by way of goal implementation. Elsewhere, Johnson (1982) has provided a hypothetical example of a different kind of mapping, which we might distinguish as substitution. While metaphorical models, as I have claimed here, are relatively stable understandings of experience, it often happens that one such model ceases to adequately capture experience for its user. Another model which shares multiple entailments with the earlier one may then be substituted for it; the shared entailments serve as bridges. If time allowed, I would give additional examples of such substitution from my material. For instance, the husband who conceptualized marriage as BEING A COUPLE felt that he and his wife were growing closer over time, and spoke of his more recent marital experience as BEING ONE PERSON. Given goal implementation, substitution, and other possible relationships between metaphorical models, it becomes critical to study the understanding process in the context of life story discourse.

References

Carbonell, Jaime, Jr.
1981 Metaphor: an inescapable phenomenon in natural language comprehension. Unpublished ms.

Johnson, Mark
1982 Metaphorical reasoning. Unpublished ms.

Lakoff, George and Mark Johnson
1980 Metaphors We Live By. Chicago: University of Chicago Press.

Quinn, Naomi
1981 Marriage is a do-it-yourself project: the organization of marital goals. In Proceedings of the Third Annual Conference of the Cognitive Science Society, Berkeley, California, August 19-21. Pp. 31-40.

Metaphoric Gestures
David McNeill
University of Chicago

An analysis of the internal structure of saying, for example, "that's ok" as based on pointing and saying "ok," suggests a form in which this (virtual) action could be expressed--namely, as a pointing gesture. Such a gesture would be regarded as a second manifestation of the internal action structure--the utterance of "that's ok" being the first (first in communicative importance). The same can be said of other utterances. The externalization of action structures in gestures offers a way of studying the internal organization of language actions that is separate from speech. The gesture and the speech can be compared in a relationship that is comparable in its ability to bring out details to triangulation.

Internal thoughts of actions--manipulations and movements of objects in the world--seem to play a metaphoric role in language actions. In producing speech a concept or meaning is shown through a (virtual) action--this imaginary manipulation or movement of objects. In the following example the concepts of pursuit and inaccessibility are presented in a complex gestural image of moving but non-closing objects. This image immediately presents a global and undivided picture of the conceptual content, while concurrently the content is segmented into words and arranged across time in the speech channel (the fact that the gesture image arises first shows that it is not a response to the words).

(1) Speech: they um wanted to get where Anansi was

Gesture: both hands held apart in the air, right hand flutters back and forth (where the underlining shows the temporal extent of the gesture).

The synthesis of thoughts on which this language action was based (as revealed in the gesture) was a (virtual) placement of two objects, one in motion, but without closure. This image shows directly the concepts of pursuit and inaccessibility. The utterance of "the wanted to get where Anansi was" is an expression of the same internal structure, as numerous detailed parallels of form between the speech and gesture channels show. For example, the participants (referred to by the pronoun and proper name) correspond to the two hands (that is, the gesture was two handed rather than one handed). The two hands were held apart at spatial extremes, and in the sentence appear at temporal extremes (rather than together as would have been possible in a frame such as "the sons [coreferent of "they"] and Anansi couldn't get together"); one participant is not in motion, and in the sentence is referred to in a stative locative construction ("where Anansi was"); the other participants are in motion, and in the sentence are referred to by the subject of a verb of motion ("they wanted to get"); and the motion of these participants in the gesture was of small extent and ineffectual, and in the sentence are referred to by the subject of the verb "want." All of these parallels are explicable if the gesture and utterance were joint manifestations of the same internal structure--a synthesis based on the idea of placement and movement of objects. This idea is a metaphor for pursuit and inaccessibility. (It is well to remind ourselves that the relationship between the structure of language actions and that of language objects--these being two completely different perspectives--is anything but clear; therefore it is not particularly interesting to ask how thoughts based on actions such as placement of objects translate into deep structures or other linguistic object configurations.)

Gesture evidence reveals a very widespread use of metaphoric thinking in performing language actions in which thoughts related to actions are used to show meanings of a non-action kind.

Mathematics discussions are accompanied by a flow of gesture which show mathematical ideas in the form of actions. The mathematical meaning of a dual is that each concept is replaced by its converse; for example, the dual of upward is downward. The following examples (2-4), taken from non-consecutive places in a technical mathematics discussion, each contain a gesture in which a hand rotates through the air from one orientation to the opposite orientation; the gestures therefore show the concept of a dual in the action realm.

(2) Speech: this gives complete duality

Gesture: right hand palm rotates upward

(3) Speech: when you dualize

Gesture: right hand palm rotates downward

(4) Speech: the powers of x kind of give a dual

Gesture: right hand palm rotates front to back

Another mathematical concept is that of a limit, and in the following examples the hands move toward some boundary marked by the other hand or a sudden stop; thus these gestures also are images of a mathematical concept in the action realm.

(5) Speech: it's an inverse limit

Gesture: right hand flattens; left hand moves up to

right hand

(6) Speech: the inverse limit of...(trails off)

Gesture: right hand goes down, then up as to a boundary

(7) Speech: which is a limit, a direct limit

Gesture: right hand moves down, then up as to a boundary

Example (5) also included a second gesture that showed the concept of an inverse:

(5') Speech: it's an inverse limit

Gesture: right hand moves in a tight loop

The concept of finiteness is shown by enclosing or pinching down on a space by curling the fingers and hands; thus here too is a mathematical concept in the action realm.

(8) Speech: through the finite pieces

Gesture: fingers curl inward

(9) Speech: to get the finite group scheme

Gesture: fingers curl inward

(10) Speech: some finite group functor

Gesture: forms a two handed bounded shape with palms facing and fingers curled

A rule of gesture production is that new movements indicate changes of meaning; and so a gesture can indicate the emergence in discourse of a new element of meaning ("information focus"). Thus in (2), for example, the new element was the concept of duality, and the other examples can be interpreted in a parallel way.

Utterances are structured to make salient the same elements of meaning. This is another parallel that suggests a common source for gesture and speech. In (2), "that gives complete duality" was structured and pronounced to achieve the same effect as the gesture: reference to the concept of duality was held off until the final sentence position (the position of the rheme) where it was given main stress, and was introduced in full lexical form. On the other hand, the sentence topic was announced first with a pronoun, and was weakly stressed. The transitive sentence form also enhanced the information focus of duality. Internally the model for (2) seems to have been that something (the sentence topic) was pushing forward the example the gesture demonstrated of duality (hence the use of "gives").

METAPHOR AND THE CONSTRUCTION OF REALITY

by George Lakoff

University of California at Berkeley

Johnson and I (1980) have argued that metaphors are essentially conceptual in nature, rather than linguistic, and that a metaphor provides a way of understanding one kind of thing in terms of another. Since we base our actions in our understanding, and since actions are real, it follows that reality, especially on the social, interpersonal, and emotional domains, is structured according to our metaphors. Though I think this is an essentially correct view, it is certainly oversimplified and in need of more detailed study.

Here are some of the questions that I think need to be answered:
- Which conceptual metaphors do we live by, and which do we use "merely" for the sake of understanding?
- What does it mean to live by a metaphor?
- Which metaphors do we believe, and which don't we believe?
- Are some metaphors more essential than others in defining a concept?

And finally:
- To what extent does a given metaphor "create" the structure of a concept it defines, and to what extent does it merely "decorate" an already given structure?

In addition, I will review very briefly some results on image-based metaphorical concepts. These results suggest to me that even spatial understanding may not be universal.

Symposium—Control of Arms

WHY IS IT EASY TO CONTROL YOUR ARMS?

Peter H. Greene
Computer Science Department
Illinois Institute of Technology
Chicago, Illinois 60616

How can our nervous systems control all the variables needed to guide our arms? How can we represent the abstract pattern of an action such as handwriting so that it may be realized in any of an infinity of variants--large, small, horizontal, vertical, with the hand weighted, or even by holding the pencil stationary and moving the paper? Most researchers who try to represent movements of artificial arms by means of computer programs have chosen, in the interest of supposed computational simplicity, to use the smallest number of "degrees of freedom", or independent joint movements that will allow desired hand movements. I will discuss the opposite idea: namely, the idea that a large number of "redundant" degrees of freedom, when used in the style that I will discuss, can simplify the control task, in that, if there are enough ways of moving, a recipe involving just a few of them can usually be found that will approximate any desired movement. In particular, the presence of "redundant" degrees of freedom allows us to rely more on ballistic (free-swinging) movements than is generally done in research on artificial arms, so that physics, rather than computation, accounts for much of the trajectory. Computations are required to set up the constraints defining and initializing a low-degree-of-freedom "virtual arm" in such a way that a satisfactory ballistic movement exists. Thus, a complicated physical arm behaves like a family of easily controlled virtual arms.

Among the points to be discussed: Using momentum saves energy, and it simplifies control. Movements may be controlled by sending new parameters to systems that control the muscles, rather than by controlling the muscles directly. The principal task in tracking a moving object, rather than being to minimize instantaneous errors, may be to synchronize an internal pattern generator to the movement. The present style of control identifies similar movements as cousins, rather than regarding them as unrelated computations. The same overt muscle movement can be more or less difficult, depending upon the higher up patterns in this hierarchy from which it is derived.

(See Greene, P.H. (1972), Problems of organization of motor systems, in Rosen, R. and Snell, F.M., eds., Progress in Theoretical Biology, Vol. 2, New York: Academic Press and Greene (1982), Why is it easy to control your arms? Journal of Motor Control, to appear.)

Internal Directional Reference Frames for Motor Coordination

C.C. Boylls

Rehabilitative Engineering
Research and Development Center

Palo Alto Veterans Administration Medical Center
Palo Alto, California 94304

Several decades ago, Graham Brown (11) found that the spontaneous walking of a high-decerebrate cat can be continuously transformed from rectilinear locomotion into either circling or uphill/downhill progression by appropriate changes of head position. The cat's performance thus carries with it an attribute of "spatial directionality" which can be independently regulated by the CNS; and the method of regulation relies, in this instance, upon postural biases created by tonic neck and labyrinthine reflexes.

Recently, experiments using decerebrate cats similar to Graham Brown's have indicated that activity within the olivocerebellar system of the brainstem is associated with postural alterations resembling those elicited from neck and labyrinths (4,5). These, too, bias the locomotor musculature so as to influence the overall directionality of walking in a wide variety of ways. However, there is one area in which the directional control exerted by the olivocerebellar system differs considerably from that seen by Graham Brown: It has "memory", in that a posture adopted by an animal as a function of olivocerebellar activity is retained for many tens of seconds after that activity ceases. By contrast, the postures of Graham Brown's animals reflect only the current position of the head, without any apparent recollection of previous positions. The directional skews associated with head movement can thus be changed in "real time" from step to step, while olivocerebellar skews establish an enduring postural context within which many steps (or other activities) may occur. It thus is tempting to hypothesize that the olivocerebellar system exists in the CNS to regulate, via postural mechanisms, an internal directional reference frame within which motor actions are elaborated and, perhaps, evaluated. But then, why should such a faculty exist?

The idea of an internal directional reference for movement was first derived theoretically from consideration of CNS mechanisms to simplify the controllable degrees of freedom in the skeletomotor system (2,8; P.H. Greene, this volume). The technique for doing this is to create functional dependencies (e.g., fixed ratios) amongst movement parameters affecting different joints, as is frequently encountered experimentally (10,9). One particular form of functional dependence employs so-called "muscle linkages" (3) of synergists at different joints, the activities of which covary in some prescribed manner (cf., ref. 12 for experimental examples). Actions carried out with such a linkage are characterized by a distinct directional skew that becomes quite apparent as the covarying parameters of the linkage are altered. Graphic illustrations of such a process may be seen in, for instance, Graham Brown-like changes in the coactivation of human leg musculature (elicited with galvanic labyrinthine stimulation) as a continuous function of neck position (13). Consequently, one might well see olivocerebellar directional biasing as just another way to parameterize muscle linkages and simplify the motor control process. But this would provide no facile explanation for the extended time-course of such biasing, nor would it define the conditions that presumably spur the olivocerebellar system into establishing a particular directional reference frame.

A speculative approach to the last question is suggested by neurophysiological studies of the olivocerebellar system and its role in regulating eye movement (ref. 1 for review). In brief, activation of the appropriate (anatomically) part of the system institutes a seconds-long nystagmus of the eyes seemingly equivalent to the olivocerebellar postural biasing of the skeletal muscles described above. This nystagmus also resembles the phenomenon of optokinetic after-nystagmus (OKAN) that occurs in humans and animals following exposure to whole-field motion of the visual world. It may come as no surprise, therefore, that the olivocerebellar system has proved to receive retinal image-motion cues which are nearly optimal for optokinetic eye movements. What is more interesting is that, in stationary human subjects, the development of OKAN is associated with illusory sensations of self-motion or "vection", which, in darkness following exposure to the moving visual stimulus, persist for prolonged periods of time (7). The rationale for this persistence, or "memory", would appear to involve an appreciation for momentum: The subject feels accelerated to some velocity by the moving visual world, and has no reason to feel decelerated when that world is no longer visible. While appropriate studies seem not to have been done, it seems reasonable to suppose that humans and animals experiencing vection will alter their motor behavior as a function of this sensation, just as they would were they experiencing actual self-motion. Because of the long time-course associated with vection, such motor adjustments will likely take the form of "static" postural biasing altering the directionality of movement. Might this be the sort of directional skewing produced by the olivocerebellar system? Might the perception of self-motion along particular trajectories be associated with the creation of olivocerebellar directional reference frames for movement?

The arguments above have been helped somewhat by the demonstration that vection sensations (and accompanying "OKAN") can be released by proprioceptive cues from the limbs (6)-- which, besides providing a role for the massive somatosensory input to the olivocerebellar apparatus, indicates that self-motion cues derive from multisensory processing. Those cues probably also owe themselves to knowledge of efferent command signals, since the quality of self-motion illusions depend upon a subject's assumptions about movement he or she is producing voluntarily. Fortunately, it appears possible to go back to Graham Brown's cat and its olivocerebellar system to see whether the directional skewing it participates in can be triggered by those conditions leading to vection in humans. Work is now underway toward that end.

References

1. Barmack, N.H. Immediate and sustained influences of

visual olivocerebellar activity on eye movement. In: Talbott, R.E., and Humphrey, D.R. (Ed.), Posture and Movement, Raven (New York), 1979: 123-168.

2. Bernstein, N.A. On the Construction of Movement, Medgiz (Moscow, 1947).

3. Boylls, C.C. A theory of cerebellar function with applications to locomotion. II. The relation of anterior lobe climbing fiber function to locomotor behavior in the cat. In: COINS Technical Report 76-1, Dept. Computer & Information Sciences, Univ. Massachusetts (Amherst), 1976.

4. Boylls, C.C. Prolonged alterations of muscle activity induced in locomoting premammillary cats by microstimulation of the inferior olive. Brain Res., 1978, 195: 445-450.

5. Boylls, C.C. Contributions to locomotor coordination of an olivocerebellar projection to the vermis in the cat: Experimental results and theoretical proposals.. In: Courville, J., Lamarre, Y., and de Montigny, C. (Ed.), The Inferior Olivary Nucleus: Anatomy and Physiology, Raven (New York), 1980: 321-348.

6. Brandt, T., Buchele, W., and Arnold, F. Arthrokinetic nystagmus and ego-motion sensation. Exp. Brain Res., 1977, 30: 331-338.

7. Brandt, T., Dichgans, J., and Koenig, E. Differential effects of central versus peripheral vision on egocentric and exocentric motion perception. Exp. Brain Res., 1973, 16: 476-491.

8. Greene, P.H. Problems of organization of motor systems. Prog. Theor. Biol., 1972, 2: 304-338.

9. Kelso, J.A.S., Holt, K.G., Rubin, P., and Kugler, P.N. Patterns of human interlimb coordination emerge from the properties of non-linear, limit cycle oscillatory processes: Theory and data. J. Mot. Behav., 1981, 13: 226-261.

10. Lacquaniti, F., and Soechting, J.F. Coordination of arm and wrist motion during a reaching task. J. Neurosci., 1982, 2: 399-408.

11. Lundberg, A., and Phillips, C.G. T. Graham Brown's film on locomotion in the decerebrate cat. J. Physiol. (Lond), 1973, 231: 90P-91P.

12. Nashner, L.M. Fixed patterns of rapid postural responses among leg muscles during stance. Exp. Brain Res., 1977, 30: 13-24.

13. Nashner, L.M., and Wolfson, P. Influence of head position and proprioceptive cues on short latency postural reflexes evoked by galvanic stimulation of the human labyrinth. Brain Res., 1974, 67: 255-268.

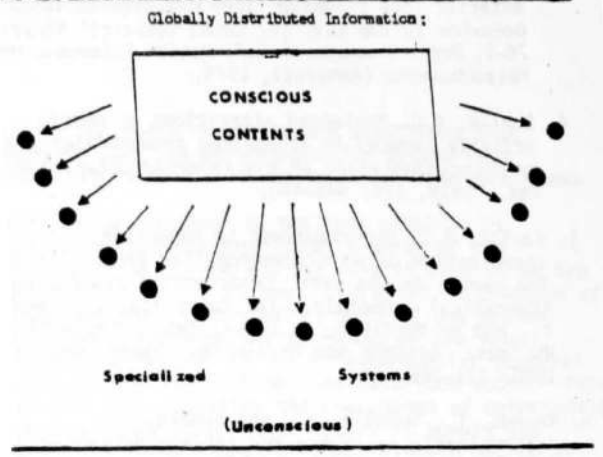
Conscious and unconscious components of intentional control.

Bernard J. Baars and Diane N. Kramer

State University of New York at Stony Brook.

How is intentional action controlled? Other papers in this symposium provide evidence for a style of motor control in which executives issue very general commands, which are interpreted "distributively" by intelligent specialized sub-systems, which are sensitive to local context. Likewise, there are classical suggestions that conscious components of intentional control serve an executive function, but without controlling motor systems in great detail: instead, the subsystems controlling actions interpret very simple conscious contents intelligently, with a view to local context (James, 1890). We suggest that there is much to be said for James' view of intentional control; further, his view fits a conception of conscious processes advanced by Baars (in press), suggesting that conscious representations are global, coherent, and informative in a nervous system consisting of distributed specialists which control all information processing details (Figure 1).

Figure 1



Note that conscious contents are globally available, but most detailed information processing is performed locally by a large set of specialized, distributed processors. The specialized processors maintain the processing initiative.

Table 1: Capability Constraints on a theory of conscious contents.

Conscious Processes	Unconscious processors
1. Computationally inefficient.	Highly efficient in specialized tasks.
2. Great range, & relational capacity.	Limited domains & relative autonomy.
3. Apparent unity, seriality, & limited capacity.	Very diverse, parallel, and together have great capacity.

Table 1 shows a set of widely-accepted facts about conscious vs. unconscious processes which fit this general view. Like conscious processes, entirely global processes are computationally inefficient because they require the cooperation or tacit consent of many other processes to remain global. They have great range of possible content since any specialist, or set of specialists has potential access to the global data base, and great relational capacity, for the same reason. Global representations, like conscious contents, have apparent unity because internal contradictions would imply competition between different processors, which would destabilize the global representation; hence any competing representations must be displayed serially, and the global component would seem to have limited capacity. Similarly, the unconscious processors of Table 1 resemble the specialized processors of Figure 1. Though this is only a first-approximation model of conscious vs. unconscious activity, it will serve as a basis for approaching conscious vs. unconscious components of intentional activity.

Now consider the facts shown in Table 2 about contrasts between conscious and unconscious aspects of intentional activities.

Table 2

Conscious components	Unconscious components (*)
Problem assignment Problem solution (Aha!)	Problem incubation
Goal representation Goal feedback	Goal execution Open-loop adjustment of future actions.
Biofeedback signal	System controlling biofeedback.
Seriality of non-automatic tasks.	Parallelism of automatic tasks.
Stimulus for reflexes and externally-driven automatic tasks.	Detailed control of reflexes and automatic tasks.
Intentional modulation of reflexes and automatic tasks.	

(*) Some of these may be momentarily conscious, but too briefly to be retrievable subsequently.

Note first that in classical problem-solving tasks, the stage of problem-assignment --- the accumulation of constraints on a possible solution --- is conscious; however, all the detailed pro-

cesses working toward a solution operate unconsciously, while the solution itself becomes conscious unexpectedly, as an "Aha!" experience. In intentional problem solving, the very fact that a goal is made conscious serves to trigger unconscious systems able to contribute to this goal. This fits the rough model of Figure 1, since distributed specialists can be triggered by a global display of a goal. These specialists then work locally on a solution, and can return a solution to the global display when they reach it. In the classical problem-solving case, the differences between conscious and unconscious parts are quite obvious; however, much the same components may also operate in other cases of intentional control, where they may occur much more quickly and less discretely.

For example in biofeedback training, a conscious feedback signal is triggered by an otherwise unconscious neural process. In itself, this is sufficient for intentional control of the unconscious process to develop. The model suggests that the feedback is "broadcast" globally, throughout the nervous system, so that one subsystem out of many millions that can control the feedback can "decide" to act whenever the feedback occurs. In this fashion, sensory feedback can come to control otherwise totally unrelated neural processes: thus, a feedback click can come to control a single motor unit (Basmajian, 1963), and the tast of saccharin can come to elicit suppression of immune function (Ader & Cohen, 1982).

The "executive ignorance" of conscious processes is not limited to new or exotic intentional control tasks. William James (1890), among others, has pointed out that "we" do not know in any detail how we do anything. One can account for this ignorance by assuming that we do not need to know anything: we can just know the goal consciously, and unconscious but very intelligent specialists will take care of execution of the goal.

Note also in Table 2 that feedback from an intentional action is conscious, a fact that presumably permits unconscious improvements in planning and execution to take place, in preparation for the next time that the action will be performed. This is especially true if there is a mismatch between the intended action and its performance.

But the case has so far been oversimplified. In fact, we cannot think of an action as being controlled by a single goal. Baars & Mattson (1981) maintain that an intention is indeed a multi-leveled goal structure, of which only a few goals tend to be conscious. The multi-leveled intention can be separated into presuppositions of the conscious goal, and subordinate systems involved in executing the conscious goal.

Further, practically all intentional actions consist of a continuous mixture of conscious and unconscious components. Generally speaking, most routine components tend to be largely unconscious, while those components that are new or involve some choice-point may be conscious. Thus, in skilled typing, we may be conscious of non-routine starting points of action, of input and output, and of attempts to override, modulate, or interrupt the typing task. Generally we seem to be unconscious of the mapping between letters and finger-strokes, of the details of motor control, and of highly repetitive input or output.

As we acquire proficiency in a task, it tends to become less and less conscious --- in terms of the model, it tends to be consigned to specialized, autonomous systems with fewer global messages. Schneider (1980) has found that tasks which are initially slow, serial and capacity-limited become increasingly fast, parallel and unlimited as they become automatic with practice. This is almost a perfec characterization of the difference between global and local processes in the current model.

Competition:

One of the most important properties of the model is that it permits competition; there is much reason to think that competition plays a central role in the control of intentional activity (Norman & Shallice, 1980). One can imagine a number of different kinds of competition in this model:

1. Conflicting intentions: intentions may be incompatible. In this, often the mismatching components seem to become conscious.

2. Conflict between superordinate and subordinate components of a single intention. This is typically the case with psychopathologies (see Table 3).

3. Conflict between an intention and its execution. Slips can be defined as actions that violate the actor's own expectations (Baars & Mattson, 1981). Slips often become conscious, perhaps because global broadcasting helps to recouple a previously decoupled goal component, whose absence permitted the slip to occur.

4. Conflict between intentions and external reality. And of course, sometimes the means needed to carry out an intention are unexpectedly unavailable.

Table 3

Perceived intentional vs. unintentional activities.

Intentional: Sense of some conscious control	Unintentional: Sense of no conscious control (*)
Most ordinary actions, thoughts, images, and feelings.	Actions: compulsions, undesired habits, slips, tics, speech defects, and addictions.
	Thoughts and images: phobic, obsessive, hallucinatory, anxiety-provoking, depressive.
	Feelings: anxious, depressive, etc.
Effect of "paradoxical intention" on unintentional activities	Resisted unintentional activities.
Success in well-known tasks	Failure in well-known tasks (TOT phenomenon)
Skeletal muscle control	Reflexes, autonomic functions, and automatic processes cued externally.
Internally motivated actions.	Externally coerced actions. Actions triggered by direct brain stimulation (Penfield & Roberts). Slips induced experimentally.
Activities whose pace is unforced.	Activities that are forced at a pace faster than normal.

(*) Some processes which do not yield a sense of

conscious control may in fact be triggered by brief conscious contents that cannot be retrieved.

We suggest the following general conclusions, based on the material presented in Table 3:

Intentional activities appear to be triggered by conscious contents. Intentions are violated not only when the action is unexpected, but also when the subordinate system appears to resist control --- e.g. when it takes longer to find a certain word than one expects. This suggests that intentions carry information about the typical duration and difficulty of a known task. Further, it also suggests that "mental effort" occurs not as a function of the complexity of a task, but rather, as a function of the degree of perceived resistance to the intention, compared to the expected duration and difficulty of the task. This view may also help explain the related case of perceived coercion (a case of unintentionalness which is not just a political fact, but also occurs very often in our educational system). Such perceived coercion from an outside source may bring about a great deal of internal competition between systems attempting to exert executive control in a way that is insensitive to the demands of the subordinate system. One implication is that intentions, too, have their own "ecology": a successful intention must fit into the system as a whole, or competition will occur which will increase the perceived effort in carrying out the intention.

References

Baars, B.J. Conscious contents provide the nervous system with coherent, global information. In R.J. Davidson, G. Schwartz, & D. Shapiro (eds.) Consciousness & self-regulation (Vol. 3). N.Y.: Plenum, in press.

Baars, B.J. & Mattson, M.E. Consciousness and intention: A framework and some evidence. Cognition & Brain Theory, 1981, 4(3), 247-263.

Basmajian, J.V. Control and training of individual motor units. Science, 1963, 141, 440-441.

James, W. The principles of psychology. N.Y.: Holt, 1890.

Schneider, W. & Fisk, A.D. Dual task automatic and controlled processing of temporal and spatial patterns. (Tech. Rep. 8002) University of Illinois, February, 1980.

Submitted Papers

How do Children Learn to Judge Grammaticality? A Psychologically Plausible Computer Model

Mallory Selfridge
Department of EE and CS
University of Connecticut
Storrs, CT. 06268

1.0 Introduction

If a young child is asked whether the sentence "ball me the throw" sounds "silly" or "ok", chances are the child will respond "silly." Encouraged to "fix it up," the child may well generate "throw me the ball." Such behavior was reported by Gleitman et al. [7] for children of two-and-a-half and five years. It implies that by these ages children have acquired at least some ability to judge a sentence's grammaticality. Further, Gleitman et al. report that by age five, children's judgements increase in sophistication. Thus children's ability to judge grammaticality apparently increases as they learn language.

Unfortunately, little is known about the mechanisms responsible for the development of such abilities. Pinker [10], reviewing language acquisition models, reports no work in this direction. Anderson's [1] model of language learning does not address learning to make grammaticality judgements. Recent research (e.g. [2,3,8]) on syntactic recognition and learning has not been integrated into a model of child learning. The question remains: "how do children learn to make grammaticality judgements?"

This paper addresses this question by proposing a three stage model, implemented and tested in the CHILD program [12,13,14,15]. During stage one CHILD knows word meanings but not syntax, and can understand sentences, but cannot tell that word order is incorrect. During the second stage, CHILD has learned active syntax, and notices incorrect word order for active sentences. During the third stage, CHILD learns passive syntax, and notices incorrect word order for both active and passive sentences. This progression corresponds generally to Gleitman et al.'s finding that as children learn more language their ability to make grammaticality judgements increases.

CHILD's mechanisms may provide part of the answer to the problem of how children learn to make grammaticality judgements of sentences with incorrect word order. These mechanisms have been developed to account for a number of different data about child language learning [14], and their extension to the problem of grammaticality judgements has been straightforward. The CHILD model suggests that children learn to make such judgements almost entirely as a side-effect of mechanisms whose primary function is directed elsewhere. This paper describes the CHILD program, and presents sample output. The question of learning to make grammaticality judgements is considered, and several predictions are described which may confirm or deny this account.

2.0 The CHILD Program

CHILD is a computer model of the development of language comprehension and generation abilities written in Franz LISP and currently running on a DEC VAX 11/780. It begins with world knowledge and language experiences similar to those received by children, and learns a subset of the word meaning and syntax which children learn. After learning, CHILD can correctly understand utterances which it previously misunderstood, and can generate English describing events it "observes."

CHILD's language comprehension process is a version of the CA program [4] which incorporates mechanisms derived from Wilks' [16] preference parsing. CHILD's analysis process combines Conceptual Dependency (CD) [11] word meanings to form a CD representing the meaning of the entire utterance. Understanding begins when the meanings of input words are placed in a short term memory. CHILD then retrieves semantic requirements associated with those slots but specific to that particular word. It searches the short term memory for a word meaning which best satisfies those features, and fills the empty slot with that meaning. The syntactic features are formed from the positional predicates PRECEDES and FOLLOWS. These relate the position of a candidate slot filler to either the word they were stored under, a filler of another slot in that word's meaning, or a lexical function word. Each slot in the meaning of a word has a collection of features describing where in the input a filler is expected to be. In order to understand different voices, these features are organized into disjunctive "feature sets." Each set characterizes one order in which slot fillers appear. During understanding, feature set selection is performed by considering which set most successfully characterizes the input.

CHILD learns syntax by acquiring syntactic features and build disjunctive feature sets. After having understood an utterance, CHILD examines a record of the input, and examines the meaning of every input word. It then examines every empty slot in each such meaning. It accesses the record of the input to find where in the input the filler for that slot occurred. It creates a description of this position using PRECEDES and FOLLOWS. CHILD must then decide whether this description constitutes a new feature set or should be merged with an existing feature set. CHILD's strategy is based on a suggestion by Iba [8]. CHILD compares the features extracted from the current input with any existing feature sets: the position description is merged with a previous set only if one is a subset of the other. Otherwise, the description is learned as a new feature set.

CHILD notices that a sentence is ungrammatical if any syntactic features within the selected feature set characterizing the position of a slot filler are not true of the position of that filler. CHILD uses these features to generate an explanation of why the sentence was ungrammatical, and uses its language generation abilities [6] to generate a correct version based on the sentence's understood meaning according to whatever word meaning and syntax it knows about at that stage of learning.

3.0 Learning to Make Grammaticality Judgements

The following example is edited from a complete

run of the program during which it learns meaning and syntax for all the words it knows. The example begins after CHILD has learned meanings for "throw", "me", "Child," "Mom," "on," "table," and "ball." For this example, the meaning of "throw" has been simplified from a complex CD into $THROW and processing of "the" has been ignored. As shown below, CHILD is initially given an ordinary sentence which it understands correctly. The second sentence has incorrect word order. CHILD understands this sentence correctly, but fails to notice the incorrect order.

```
CHILD hears "throw me the ball"
CHILD understands
($THROW ACTOR (CHILD) OBJECT (BALL1) TO (PARENT1))

CHILD hears "ball me the throw"
CHILD understands
($THROW ACTOR (CHILD) OBJECT (BALL1) TO (PARENT1))
```

Transition to the second stage occurs when CHILD learns active syntax for "throw." This occurs when it hears an example sentence whose interpretation is unambiguous, and has heard the word a number of times without modifying its meaning. Given this sentence, CHILD notes the positions of the fillers (summarized in linear order here), and stores them in a feature set under the word "throw."

```
CHILD hears: "throw me the ball"
CHILD's understanding is:
($THROW ACTOR (CHILD) OBJECT (BALL1) TO (PARENT1))

CHILD learns syntax of "throw"
order is: $THROW TO OBJECT

ATTEMPTING MERGE OF NEW FEATURES WITH EXISTING SET
NO EXISTING FEATURES SETS
CREATING NEW FEATURE SET
```

Having learned active syntax for "throw," CHILD uses this knowledge during stage 2 understanding. It notices when the word order of a sentence is incorrect, as the first sentence below shows. CHILD prints out the reasons it thought the sentence was incorrect, and generates a correct version from the understood meaning of the sentence. However, CHILD also decides that a passive sentence with correct order is incorrect, as the second sentence below demonstrates.

```
CHILD hears "ball me the throw"
CHILD understands
($THROW ACTOR (CHILD) OBJECT (BALL1) TO (PARENT1))

INCORRECT SENTENCE NOTICED:
"throw" should precede "ball"
"throw" should precede "me"
CORRECTION: "throw me the ball"

CHILD hears
"the ball was throw n on the table by Child"
CHILD understands
($THROW ACTOR (CHILD) OBJECT (BALL1)
        TO (TOP VAL (TABLE1)))

INCORRECT SENTENCE NOTICED:
"Child should precede "throw"
"throw" should precede "ball"
CORRECTION: "Child throw ball on table"
```

The transition to the third stage occurs when CHILD learns passive syntax for "throw," and creates a second feature set for the new syntactic features.

```
CHILD hears: "the ball was throw n to Mom by
  Child"
CHILD's understanding is:
($THROW ACTOR (CHILD) OBJECT (BALL1) TO (PARENT1))

CHILD learns syntax of "throw"
order is: OBJECT "was" $THROW "to" TO "by" ACTOR

ATTEMPTING MERGE OF NEW FEATURES WITH EXISTING SET
MERGE FAILS
CREATING NEW FEATURE SET
```

Once CHILD has learned passive syntax (reported in more detail in [15]), it can then judge passive sentences. It correctly judges the passive sentence which it previously judged incorrect. The second sentence below is an incorrect passive, and CHILD correctly understands it, prints out the reasons it was judged incorrect, and generates its corrected version.

```
CHILD hears
"the ball was throw n on the table by Child"
CHILD understands
($THROW ACTOR (CHILD) OBJECT (BALL1)
        TO (TOP VAL (TABLE1)))

CHILD hears: "the ball to Mom throw n by Child"
CHILD's understanding is:
($THROW ACTOR (CHILD) OBJECT (BALL1) TO (PARENT1))

INCORRECT SENTENCE NOTICED:
"the ball" should precede "was"
"thrown" should precede "to Mom"
CORRECTION: "the ball was thrown to Mom by Child"
```

As shown above, CHILD does progress through a series of stages which generally correspond to data reported by Gleitman et al., during which it learns to make increasingly accurate and complex grammaticality judgements. Initially knowing no syntax, all sentences are judged correct. After learning active syntax, it successfully judges active sentences, but judges passive sentences as if they should have been active. Upon learning passive syntax, CHILD judges both active and passive sentences correctly. In the complete run, CHILD also learns to understand noun phrases, prepositional phrases, and adverbial phrases, and learns to make judgements about sentences containing these constructions.

4.0 How Do Children Learn to Make Grammaticality Judgements?

CHILD's answer to this question depends upon a number of factors: a) the representation of language syntax as a set of independent features characterizing the position in the input where a slot filler may occur; b) learning of syntactic features while learning to understand; c) the evaluation of syntactic features and semantic preferences as a necessary part of understanding. Given these mechanisms, children make grammaticality judgements by analyzing syntactic violations occurring during understanding. They generate correct versions of incorrect sentences by applying their language generation ability to the understood meaning of that sentence. Thus children acquire the ability to make grammaticality judgements as a side effect of acquiring syntactic features needed for understanding.

This account of learning to make grammaticality judgements makes several predictions. First, this model predicts that people's judgements of incorrect sentences will not merely be "grammatical" or "not grammatical," but rather judgements as to the relative grammaticalness of a sentence. This prediction follows from CHILD's generation of a number of different reasons for a sentence's incorrectness. Second, this model predicts that as a child learns increasing amounts of syntax he will find certain sentences increasingly ungrammatical. This is because newly learned syntax becomes available to judge grammaticality, and thus the number of violated syntactic features increases. Third, this model predicts that before learning passive syntax children will judge non-reversible passive sentences to be ungrammatical. This is because at this stage they are using active syntax to understand passive sentences. Later, when they have learned passive syntax, they will no longer judge non-reversible passive sentences ungrammatical.

Clearly, this work has not completely solved the problem of how children learn to make grammaticality judgements, since there are certainly a large number of complex syntactic constructions which CHILD cannot handle. In addition, it is not even clear what exactly constitutes such judgements, since Gleitman et al. report that children think sentences are "silly" for a number of reasons not discussed here. It is hoped, however, that this approach will prove a promising direction for further research, since it is grounded in mechanisms which manifest and explain a number of other psychological data.

Acknowledgements

Thanks to Rich Cullingford for sponsoring this paper, and to Peter Selfridge, Oliver Selfridge, Don Dickerson, Jason Engleberg, and Marie Bienkowski for helpful discussions of this work and for commenting on drafts of this paper.

References

[1] Anderson, J.R. (1981). A Theory of Language Acquisition Based On General Learning Principles. Proc. 7th IJCAI, Vancouver, Canada.

[2] Baker, C.L. and McCarthy, J.J. (1981). The Logical problem of Language Acquisition. M.I.T. Press, Cambridge, Mass.

[3] Berwick, R.C. (1977). Learning Structural Descriptions of Grammar Rules from Examples. Proc. 5th IJCAI, Cambridge, Mass.

[4] Birnbaum, L., and Selfridge, M. (1981). Conceptualal Analysis of Natural Language, in Inside Computer Understanding: Five Programs plus Miniatures. Schank R. and Riesbeck C.K. (eds.), Lawrence Erlbaum Associates, Hillsdale, NJ.

[5] Chomsky, N. (1965). Aspects of the Theory of Syntax. M.I.T. Press, Cambridge, Mass.

[6] Cullingford, R.E., Krueger, M.W., Selfridge, M. and Bienkowsky, M. (1981). Automated Explanations as a Component of a CAD System. IEEE Trans. SM&C. Special Issue on Human Factors and User Assistance in CAD, December, 1981.

[7] Gleitman, L.R., Gleitman, H. and Shipley, E.F. (1972). The Emergence of the Child as Grammarian, Cognition, 1-2/3:1-164.

[8] Iba, G. (1979). Learning Disjunctive Concepts from Examples. M.I.T. A.I. Memo #548, M.I.T., Cambridge, Mass.

[9] Marcus, M. (1980). A Theory of Syntactic Recognition for Natural Language. M.I.T. Press, Cambridge, Mass.

[10] Pinker, S. (1979). Formal Models of Language Learning. Cognition, 7:217-283.

[11] Schank, R.C., (1973). Identification of Conceptualizations Underlying Natural Language. In R.C. Schank and K.M. Colby (eds.) Computer Models of Thought and Language. W.H. Freeman and Co., San Francisco.

[12] Selfridge, M. (1980). A Process Model of Language Acquisition. Computer Science Technical Report 172, Yale University, New Haven, Ct.

[13] Selfridge, M. (1981a). Why Do Children Say "Goed"? A Computer Model of Child Generation. Proc. Third Annual Meeting of the Cognitive Science Society. Berkeley, CA.

[14] Selfridge, M. (1981b). A Computer Model of Child Language Acquisition. Proc. 7th Int. Joint Conf. on Artificial Intelligence. Vancouver, Canada.

[15] Selfridge, M. (1982). Why Do Children Misunderstand Reversible Passives? The CHILD Program Learns to Understand Passive Sentences. Submitted to the 3rd Annual AAAI Conference, Pittsburgh, Penn.

[16] Wilks, Y., (1973). Parsing English II. In E. Charniak and Y. Wilks (eds.) Computational Semantics. North-Holland Publishing Co., NY, NY.

PATHFINDER: INVESTIGATING THE ACQUISITION OF COMMUNICATIVE CONVENTIONS

Robert Cummins
The University of Wisconsin--Milwaukee
Eric Dietrich
Martin Marietta Corporation

This work was supported in part by a grant from the National Science Foundation, and by the Institute of Cognitive Science, University of Colorado (Institute of Cognitive Science publication no.).

ABSTRACT

PATHFINDER is a system that solves coordination problems that require acquisition of a convention governing the intended meaning of a symbol. LEADER blazes a trail through a maze by leaving symbols in the various paths, and FOLLOWER must find LEADER by discovering the intended meanings of these blazes. PATHFINDER is the first step in a project to design a system that can solve a variety of coordination problems of the sort implicated in language acquisition. Solving certain coordination problems is communicating. Since coordination problem solution can become conventional (as David Lewis has shown), communication can become conventional, and that is language in its most general form. As conventions are acquired, more sophisitcated coordination problems can be solved, and more sophisticated conventions can be acquired. Eventually, it should be possible to acquire conventions governing identifiers and general terms, and this will enable use of a first order language via a recursive procedure adapted from Tarski by Cummins.

PATHFINDER: INVESTIGATING THE ACQUISITION OF COMMUNICATIVE CONVENTIONS

The PATHFINDER project is a study of the acquisition of the capacity to communicate by means of convention-governed symbols, and of the knowledge structures required for such communication. The project revolves around a series of PATHFINDER programs, each of which contains two programs--LEADER and FOLLOWER--which together solve coordination problems in a way that requires acquisition of conventions governing the meaning of a symbol. We begin by sketching the theoretical background, then turn to PATHFINDER itself.

In 1973, Jonathan Bennett (Bennett, 1973, 1976) outlined a two phase account of language acquisition based on the pioneering work of Grice on meaning (1957, 1969) and Lewis on conventions (1969). In phase one, he explains along Grician lines what we shall call pre-conventional communication: cases in which a speaker S performs some action and thereby communicates with an audience A in a way that doesn't depend on the prior existence of any shared rules or conventions. In phase two, he imports Lewis' account of conventions to show how pre-conventional cases could lead to the establishment of a convention between S and A with the result that S's act-type comes to have a conventional meaning. Since Bennett's work in this area has not received the attention it deserves outside of philosophy, (especially in AI and cognitive psychology) we begin with a brief review of his two-phase account.

Phase One: Pre-conventional Communication. Bennett takes from Grice the following conditional.

> (GC) If S utters E, intending thereby to get A to believe that p, and relies for the achievement of this upon the Grician Mechanism (GM), then S means by E that p.

Here is what we shall understand by the Grician Mechanism.

> (GM) A recognizes S's intention to get A to believe that p, and is led by that recognition, through trust in S, to believe that p.

This is a simplified version of Grice's more recent accounts, but we require only a rather crude sufficient condition at this stage of the account. Bennett claims that (GC) could be satisfied by pre-linguistic S and A, i.e., by S and A who share no conventional means of communication. We agree with this assessment for reasons that will emerge later. For now we shall simply assume that pre-linguistic S and A could satisfy (GC)--though perhaps only rarely and in rather special circumstances--and that (GC) does in fact formulate a sufficient condition for communication between S and A.

Phase Two: Conventionalization. The second phase of Bennett's account imports Lewis' treatment of conventions to show how a convention could emerge between S and A governing S's communicative actions. For present purposes, the crucial feature of Lewis's theory is this.

> (L) When a group achieves coordination in a certain situation by acting in a certain way, and they act that way because (i) they wish to achieve coordination, and (ii) each actor knows, and knows the others know, that that is how coordination has been achieved in the past, then the group has a convention governing that situation.

(L) applies to cases involving coordination of action, whereas our problem involves coordination between S's action and A's beliefs. But (L) is easily extended to accomodate this fact because the sorts of reasons A can have for adopting a belief so as to coordinate with S are the same sorts of reasons A will typically have for acting so as to coordinate with S. In particular, A can have as a reason for adopting the belief that S intends A to believe that p in uttering E the fact that A knows, and knows that S knows, that in the past S's intention in uttering E has been to get S to believe that p. If A is then led, through trust in S, to believe that p, we have a case that satisfies (GC) because S's utterance of E is governed by a convention existing between S and A. This yields the following account of conventional meaning.

> (CM) Utterance-type E conventionally means that p when uttered by S to audience A if (a) in the past, S has uttered tokens of E to A only when S meant that p, and (b) this fact is mutually known to S and A, and (c) because of this mutual knowledge it continues to be the case that when S utters tokens of E, S means, and is understood by A to mean, that p.

We can put the pre-conventional case and the conventional case together in an obvious way. Suppose S intends to get A to believe that a coconut is about to fall on A, and S goes through a certain performance that results in A recognizing S's intention and, via trust in S, adopting the belief that A is about to be hit by a coconut. Here we have a pre-conventional case in which communication occurs only because conditions are especially propitious, and because S's performance has a certain natural suggestiveness. Next time, however, the mechanism of convention will set in, and, as repetitions occur, the special conditions favoring the original success will no longer be necessary. S's performance can be streamlined by a process akin to stimulus substitution to the point where it need have no special features beyond the fact that A and S perceive it to be of the same type as its predecessors. Thus, the account allows for the fact that a sign may, so far as its physical characteristics go, have any meaning whatever.

Extending the Account. As it stands, the account just sketched hasn't a chance of being a full-scale theory of communicative conventions, for it begins and ends with sentence meanings--meanings have the form "that p" where p is a proposition. Since there cannot be infinitely many meaning conventions, it follows that the account just rehearsed runs afoul of the fact that a natural language contains infinitely many non-compound sentences having distinct meanings.

This defect has been repaired in Cummins (1978), by introducing Grician meanings for identifiers and general terms. Here are the relevant conditions.

> (ST) There is a convention whereby N refers to x in S's language if (a) in the past S has uttered N only when intending to identify x, and (b) this fact is mutually known by S and S's audience, and (c), because of this mutual knowledge it continues to happen that when S utters N S identifies x.
>
> (P) There is a convention whereby G means yellow in S's language if (a) in the past S has uttered G only when he/she/it meant yellow, and (b) this fact is mutually known to S and S's audience, and (c), because of this mutual knowledge it continues to happen that when S utters G, S means, and is understood to mean, yellow.

We can now state a relation between these meanings and satisfaction conditions, and import the standard recursion on the latter, to generate conventional meanings (though not meaning conventions) for an infinity of non-compound sentences.

> (S) 'The i-th member of the sequence f is red' gives the satisfaction condition for a token consisting of the general term G applied to the i-th variable iff the (or a) conventional maning of G is 'red'.

(S) allows us to go back and forth between satisfaction conditions and conventional meanings. If we start with cases for which conventions exist for the primitive general terms, we get satisfaction conditions for those terms by moving from the meaning to the satisfaction part. We can then use the standard recursion to get a satisfaction condition for any first order combination of the primitive general terms. Then, moving from the satisfaction part of (S) to the meaning part, we get conventional meanings, though not meaning conventions, for complex general terms. It is well-known that this sufficies to fix the truth-conditions for each sentence in a first-order language.

Investigating Convention Acquisition. The acquisition and use of communicative conventions has not been very extensively investigated by researchers in artificial intelligence or cognitive psychology, presumably because the requisite theoretical background has seemed lacking. However, putting Grice's account of communication together with Lewis' account of conventions yields a powerful theory of the acquisition of communicative conventions. Extending the account to apply to acquisition of conventions governing identifiers and general terms makes it possible to use the recursive apparatus of Tarski's theory of truth definitions to generate meaning conventions for every sentence in a first order language having a finite number of semantically primitive terms. The upshot is a theory of language acquisition for first order languages. This theory, however, is incomplete or vague at several critical points. (1) The theory tells us what it is to be a party to a communicative convention governing a symbol with a propositional meaning, but it does not tell us how humans can or do actually solve primitive communicative convention acquisition problems. (2) The theory tells us what it is to be a party to a communicative convention governing an identifier or general term, but it does not tell us how humans can or do acquire such conventions on the basis of simpler shared communicative conventions, viz., conventions governing symbols with propositional meanings.

We propose to meet point (1) by adding the hypothesis (i) that primitive communication problems can be solved, and appropriate conventions acquired, in the course of solving simple coordination problems that contain the communicative problems as sub-problems. The problem analyzed by PATHFINDER is just such a containing problem. We propose to meet point (2) by adding two hypotheses: (ii) that the power of a group of agents to solve coordination problems increases as that group acquires communicative conventions; (iii) that solving relatively more complex containing coordination problems enables agents to acquire relatively more sophisticated communicative conventions. It is these three hypotheses that the PATHFINDER PROJECT is primarily designed to investigate.

PATHFINDER: Embedding Communication Problems in other Coordination Problems. Pre-linguistic communication problems are difficult to solve in part because propositional attitudes are hidden. It is difficult for a speaker-audience pair to determine whether or not they have succeeded. This difficulty can be overcome by embedding primitive communication problems in other non-communicative coordination problems that are more tractable. If S and A are engaged in some cooperative activity, the success or failure of their efforts to communicate will be more or less obviously reflected in the success or failure of that activity.

In PATHFINDER, LEADER and FOLLOWER must solve such an embedded coordination problem. LEADER blazes a trail through a maze by leaving symbols in the various paths, and FOLLOWER must find LEADER by discovering the intended meanings of these blazes: LEADER must enable FOLLOWER to find LEADER. In the process, they must solve a primitive communication problem. For example, in the level-one version of PATHFINDER, FOLLOWER may learn that when LEADER marked a path "Y", LEADER meant that that path is to be avoided. Suppose FOLLOWER locates LEADER by avoiding paths marked "Y". Then LEADER and FOLLOWER will have solved their main coordination problem, and they will have solved a primitive communication

problem as well. Most importently, however, they will have solved a primitive convention acquisition problem: both know that "Y" means "avoid this path". This convention can be used in the solution of other related coordination problems, thereby increasing the power of LEADER and FOLLOWER to solve such problems, and hence increasing their power to acquire other conventions. For example, it is evidently easier for FOLLOWER to grasp an identifer in the context of an already understood instruction. "Avoid Y at zz," links use of the identifier to solving the embedding coordination problem (find LEADER), thereby making it possible for LEADER and FOLLOWER to recognize successful communication, and hence to acquire a convention governing use of the identifier. Conventions are a special kind of knowledge that increase capacity to solve coordination problems far more effectively than other types of shared knowledge. Advanced LEADER-FOLLOWER pairs will come to share conventions governing such things as the identifiers, general terms, and syntactic rules of a relatively sophisticated language.

Preliminary research has suggested a list of parameters of two types, intrinsic and contextual, the values of which define a relative level of sophistication. The coordination problems analyzed by PATHFINDER are significantly different from each other depending on the type of maze FOLLOWER faces (intrinsic parameters) and the amount and type of knowledge, including conventions, shared by LEADER and FOLLOWER (contextual parameters). This is especially significant given the hpypothesis that the capacity of two parties (LEADER and FOLLOWER, SPEAKER and AUDIENCE) to solve coordination problems should increase as simple problems are solved and conventions are acquired for future use.

Intrinsic Parameters. FOLLOWER will eventually have to face mazes that vary in at least the following ways: (i) number of branches per node; (ii) number of symbols per branch (including blanks); (iii) complexity of symbols--e.g., context sensitivity and reference to other parts of the maze; (iv) noise--e.g., symbol-like objects in the maze not left by leader.

Contextual parameters. To solve the coordination problem set by a relatively general maze, LEADER and FOLLOWER will have to share some knowledge. The amount and type of shared knowledge are contextual parameters of the coordination problem, for they specify the cognitive context in which the coordination problem is attacked. These include: (i) previously acquired conventions, if any, (ii) mutual knowledge of capacities--e.g., can LEADER cut down a tree, and does LEADER know FOLLOWER knows this? (iii) mutual knowledge of what is likely to be a natural rather than an artefactual feature--e.g., that pine cones are noise in a forest, but possible blazes in a building; (iv) mutual antecedent knowledge of the terrritory; (v) mutual knowledge of behavioral and cognitive tendencies. These parameters are best thought of as "passed" to LEADER and FOLLOWER from containing systems that specify the goals (blaze trail; find LEADER), contain records of mutual knowledge, and handle general reasoning and decision making, including when to give up, or to give up trying hard and just "try something" (a common strategy in communication).

The level-one version of PATHFINDER (which has already been implemented), involves a maze in which all branching is binary, there is at most one symbol per branch, and noise is limited by the assumption that only the symbols encountered at the first node are significant. In a level-one maze, FOLLOWER faces a relatively simple but non-trivial task. A maze that is general along all four dimensions specified above will evidently require a highly "experienced" LEADER-FOLLOWER team, a team that, we suspect, will have to share several powerful conventions to be effective.

Summary. The PATHFINDER project is designed to investigate the following strategy for language acquisition. S and A, given some shared knowledge and goals, but no shared conventional means of communication, solve a coordination problem such as that faced by LEADER and FOLLOWER. Several successes produce a shared convention. Now that S and A share a convention, they can solve more diffecult coordination problems, hence acquire more sophisticated conventions. Eventually, S and A will be able to acquire conventions governing identifiers and general terms, and hence, by a recursive process, a first order language. Since solving certain coordination problems is communicating, and coordination problem solution can become conventional, communication can become conventional, and that is language. Standard approaches to the problem of symbolic communication have emphasized acquisition of knowledge of a language. Yet it seems clear that learning a language is neither necessary nor sufficient for communication. Knowledge of a language is a means to understanding a speaker, or communicating with an audience. Language use and understanding is not likely to be properly understood if it is studied independently of the cognitive task that motivates it. The present project, in emphasizing the acquisition of communicative conventions, focuses on the cognitive task which language learning subserves and thereby avoids studying language acquisition "out of context".

REFERENCES

Bennett, Jonathan (1973). "The Meaning-Nominalist Strategy." Foundations of Language 10, 141-168.

Bennett, Jonathan (1976). Linguistic Behavior. Cambridge: Cambridge University Press.

Cummins, Robert (1978). "Intention, Meaning and Truth-Conditions." Philosophical Studies 35, 345-360.

Grice, H. P. (1957). "Meaning." The Philosophical Review 66, 377-388.

Grice, H. P. (1969). "Utterer's Meaning and Intentions." The Philosophical Review 78, 147-177.

Lewis, David (1969). Convention. Cambridge: Cambridge University Press.

PLAY CONSIDERED AS A STRATEGY FOR KNOWLEDGE ACQUISITION

Paul D. Scott
Department of Computer and Communication Sciences
University of Michigan
April 1982

1: INTRODUCTION

If you ask a layman what he means by the term 'play' he will probably reply "activities which are useless but fun" or something very similar. If you ask a developmental psychologist the same question you will probably get much the same answer although he is likely to phrase it differently:-

> "Play consists of behaviors and behavioral sequences that are organism dominated rather than stimulus dominated, behaviors that appear to be intrinsically motivated and apparently performed 'for their own sake' and that are conducted with relative relaxation and positive affect."
> Weisler and McCall (1976)

Patterns of behavior which appear to have no external purpose but are nevertheless enjoyable for the participant present something of a biological paradox. The majority of activities which are accompanied by positive affect clearly promote, either directly or indirectly, the participant's homeostatic or reproductive goals. An adequate theory of play must resolve this paradox by attributing a function to play.

A number of theories have been advanced which attempt to do this by suggesting what the organism may gain by engaging in play. Space does not permit a discussion of the relative merits of these theories but see Weisler and McCall (1976) and Gilmore (1966) for reviews. Fortunately one particular theory appears to enjoy almost universal support. This we shall call the 'Cognitive Development Hypothesis'. Its basic premise is that the organism learns something through the process of play which is of value in later life. This theory has been advanced in a bewildering variety of forms which largely reflect the enormous range of things which a child learns. Taken together these various theories amount to a claim that play is the fundamental learning strategy by which children acquire mastery of themselves and of the perceptual, motor, cognitive and social skills which they will need throughout life.

The cognitive development hypothesis provides an explanation of the function of play and hence resolves the paradox. It is very widely accepted by developmental psychologists, primatologists, pediatricians and laymen. Strangely it has received little acknowledgment from learning theorists. Thus a large and reputable text on learning theory (Hilgard and Bower 1975) contains no index reference for play. Piaget does assign a relatively minor role to play in his model but regards it as a particular case of assimilation rather than a fundamental learning strategy. Play has been equally ignored by artificial intelligence researchers interested in machine learning. I am not aware of any program which explicitly incorporates play as a learning activity although I think it would be fair to describe the behavior of AM (Lenat 1976) as playing with numbers. Otherwise AI programs seem to be based on the assumption that learning must be either a classroom experience (learning with a teacher) or an apprenticeship (learning while doing the task). This paper is intended to exhort both learning theorists and AI workers to take play more seriously.

Although the cognitive development hypothesis provides an explanation of the function of play it does not constitute a complete theory. Such a theory must provide an account of how play activity is instigated, motivated and rewarded. It must explain the content and structure of play activities. The cognitive development hypothesis only provides a framework within which more complete theories may be developed. The rest of this paper is devoted to sketching the outlines of one such theory.

2: A THEORY OF PLAY

If play is a method of building a cognitive representation then any theory of play must make some assumptions about the nature of the representation which is built. I therefore begin the development of the theory with the following postulate:-

> Play is an activity directed towards building a representation of the world in terms of the organism's abilities to do things to or with the entities which it encounters in the world.

This hypothesis makes a strong claim about what is learned during play. It asserts that the organism is attempting to discover what it can do rather than what it should do. That is, it is not primarily concerned with learning what actions have desirable outcomes. It is of course possible, and indeed probable, that the organism will obtain information about what it should do as a side effect of trying to discover what it can do, but the claim made in the hypothesis is that such information is not the goal of play behavior. Note that this does not imply that the organism will not be trying to determine the consequences of its actions but only that it will not be directly concerned with the values of those consequences.

This form of representation in which the world is modelled in terms of how it relates to the organism's behavioral capabilities has some obvious merits. For example, it is an essential prerequisite for any kind of problem solving behavior since it enables the organism to generate alternative courses of action in a

given situation. However, since it is most readily understood in terms of simple motor responses to a given event there is a serious danger of underestimating its power and generality. It is therefore worth pointing out that it strongly resembles Gibson's notion of 'affordances' (Gibson 1977). It is also closely related to the pragmatic theory of meaning due to Peirce (1878) and subsequently elaborated by James, Dewey and Mead among others. For this reason we shall refer to it as a 'pragmatic representation'. Object-based programming languages such as SIMULA and SMALLTALK represent entities using what is essentially a pragmatic representation.

3: IMPLICATIONS OF THE HYPOTHESIS

We now explore some of the implications of the hypothesis that play is a strategy for building a pragmatic representation of the world. In executing an ordinary goal oriented task the organism is attempting to effect some change of state in its world. In doing this it uses knowledge of the properties of the world. In play the organism is attempting to effect some change of state within its own representation of that world. In doing this it will use knowledge regarding that representation. Thus it can be seen that the goals of play are metagoals and hence that play involves access to metaknowledge.

What sort of metaknowledge would be relevant for the development of a pragmatic representation? If the organism is to discover what it can do then it presumably needs to have some representation of what it does not know it can do. That is the metaknowledge must represent the organism's ignorance. Such a representation could be used to determine the course of play behavior. Thus the organism would in effect conduct experiments whose purpose is to reduce its own ignorance of its capablities in a manner loosely analagous with scientific research.

The introduction of the concept of metaknowledge raises the spectre of an infinite regress. Where does the metaknowledge come from? Is it necessary to play at playing in order to discover how to play? The threat of an endless regress can be avoided if the same activities which provide information for the pragmatic representation of the world also provide the information needed to build a model of the organism's ignorance. This constraint is not only satisfiable but also explains one of the basic empirical findings regarding play and exploratory behavior: the probability that a child will play or explore is related by an inverted-U curve to the novelty of a situation. In a highly familiar situation the child will have a detailed pragmatic representation and correspondingly low ignorance and thus there is little to be gained by play. Conversely in a totally unfamiliar situation the child will have virtually no pragmatic representation and hence have no knowledge of its own ignorance. In such circumstances he or she would essentially not know how to play. Only in the intermediate case in which a partial pragmatic representation exists is the child able to construct potentially useful play activities.

4: A SIMPLE IMPLEMENTATION

In order to clarify the ideas discussed in the preceding section by providing a concrete example and to demonstrate that such hypotheses can indeed lead to a successful learning program, I shall now describe a very simple concept learning program which learns by playing.

The organism in this case is a LISP program called PAN. PAN operates in a simple blocksworld type of environment. In this world are numerous objects which each have the properties of color,size,texture and shape. Each of these properties may take one of several discrete values. PAN is able to apply three types of action to these objects. It can push them, kick them and pick them up. However these actions will only result in the object moving in certain cases. For example the operation of picking up might only result in the object moving if the object were small. Initially PAN does not know what classes of objects its three kinds of action will succeed on. Thus PAN's task is to discover the equivalence classes of kickable objects, pushable objects and objects which may be picked up. It must experiment entirely without external guidance until it is confident that it can predict the applicability of an action to an object.

PAN does this by developing a class hierarchy. Initially it possesses only one class - the class 'Things'. All objects are instances of this class and all actions are initially attached to this class. As part of the attachment of an action to a class PAN stores an estimate of the probability that the action can be applied to a member of that class. This probability estimate is revised every time PAN tries to apply that action to an instance of that class. If this probability estimate is very large or very small then PAN is relatively certain about the applicability of the action to instances of the class and hence has no need to conduct further experiments. If however the probability is in the region of 0.5 then PAN is highly uncertain and further development of the class hierarchy is needed. The actual measure of uncertainty used in the system is Shannon's information function (Shannon and Weaver, 1949). In fact any function of the probability which was unimodal in the interval 0 to 1 with a maximum at 0.5 and a value of 0 at 0 and 1 would serve. The use of the Shannon function has the advantage of allowing one to interpret it as the informational value of a new subclass rather than being a meaningless number. The initial probability estimate assigned to each action for its attachment to the class Things is 0.5 and hence they each have an uncertainty of 1.

A cycle of the system is called an experiment. In each experiment the system finds the action which is attached to a class with highest uncertainty. An instance of that class is selected and the action applied. If the action is successful then, apart from the increase in the estimated probability, nothing else happens and the system begins a new experiment.

If the action is unsuccessful then one of two processes may occur: a new subclass may be created or the action may be detached from the class. If the uncertainty exceeds a certain

threshold then PAN will attempt to construct a subclass of the class in which the action has just failed and then attach the action to the new subclass. (The action remains attached to the original class). It does this by repeatedly selecting instances of the original class until it finds one on which the action succeeds. It then selects a random attribute of that instance, for example its color or its size, and uses that as the criterion for membership of the new subclass. The action is then attached to the new subclass with an initial probability estimate which is identical to the current probability estimate for the attachment of the action to the original class.

This newly created subclass may or may not contain a higher percentage of objects to which the action may be applied. If it does then the subclass is clearly useful and hence will be retained. If it does not then the probability estimate will eventually fall below that of the corresponding estimate in the parent class. When this happens the action is detached from the subclass. If any class has no actions attached then it is removed. Hence only useful classes are retained. In this way the system develops a hierarchy of classes as it attempts to reduce its uncertainty. The system is able to learn both conjunctive and disjunctive concepts and will eventually reach a stage when all uncertainties are below threshold. In this situation PAN announces that it is bored and halts. Note the system does not necessarily find a minimal set of classes to represent the concepts it is discovering. This could be done at the expense of more elaborate rules for modifying the hierarchy. It does however achieve a correct if redundant representation. In this respect its behavior resembles that of human beings.

The above account is simplified in one respect. Once subclasses have been constructed any given object may be an instance not only of a given class but also of one or more of its subclasses. Thus, if PAN is doing an experiment which involves applying an action to an instance of some class, the particular instance selected may also be a member of a subclass to which the action is also attached. In these circumstances the experiment is effectively transferred to the subclass which has the highest probability estimate. The result of the experiment modifies the probability estimates of both the subclass and the parent class. However a second probability estimate is also kept for each attachment which is a measure of the proportion of attempted applications which were not passed down to a subclass. This second probability estimate, called usage, is multiplied by the Shannon information function in determining the uncertainty. This is analogous to Shannon's measure of the entropy of an information source.

The reason for this modification is that if it were omitted the uncertainty of parent classes would remain high even when the appropriate subclasses had been constructed. This would lead to endless redundant experimentation. The modification described ensures that a class with successful subclasses will have low uncertainty values despite not having probability esimates close to 1.

As indicated earlier PAN is only intended as a demonstration that the play theory can be used as the basis of a learning system which works without the assistance of a teacher. We are developing a much larger version of the system in which objects may possess relational attributes and actions may change those relations. PAN is however only a simple instantiation of the use of a play based learning strategy.

The pragmatic representation takes the form of a class hierarchy with actions attached to classes. The metaknowledge of its own ignorance takes the form of the associated uncertainties. The same experiments which lead to alterations in the pragmatic representation also change the representation of ignorance.

Because PAN operates in a very restricted universe it eventually learns all that can be learned. Generally we should not expect this to happen. As the pragmatic representation becomes richer the organism has more things to be uncertain about. Hence the process of building the representation becomes a never ending search for something even better while retaining the best that has been achieved so far.

5: REFERENCES

Bruner,J.S., A.Jolly and K.Sylva 1976
"Play - Its Role in Development and Evolution"
Penguin Books Ltd., Harmondsworth, England

Gibson,J.J. 1977
"The Theory of Affordances"
In "Perceiving, Acting and Knowing: Toward an Ecological Psychology" Eds. R.Shaw and J.Bransford, Erblaum, pp 67-82

Gilmore,J.B. 1966
"Play: A Special Behaviour"
In "Current Research in Motivation", Ed. R.N. Haber, Holt, Rinehart and Winston, pp 343-354

Lenat,D. 1976
"AM: An Artificial Intelligence Approach to Discovery in Mathematics as Heuristic Search"
Doctoral dissertation, Stanford University, July 1976.

Peirce,C.S. 1878
"How to Make Our Ideas Clear"
Popular Science Monthly, January 1878, pp 286-302
Reprinted in "Charles S. Peirce: Selected Writings"
Ed. P. P. Wiener, Dover, 1966

Shannon,C.E. and W.Weaver 1949
"The Mathematical Theory of Communication"
University of Illinois Press, Urbana, Chicago, London

Weisler,A. and R.B.McCall 1976
"Exploration and Play - Resume and Redirection"
American Psychologist, 31, pp 492-508

An Experimental Architecture that supports Non-Temporal Prediction

Paul Robertson
University of Texas at Dallas
Natural Sciences & Mathematics
Box 688, Richardson
Texas 75080, USA

Abstract

A constructive theory of memory organisation has been developed, based upon the principle of non-temporal prediction. The theory predicts much of the experimental findings on recall and forgetting and provides a computational foundation for some of the intuitive notions of the society of mind theory. This paper describes an experimental architecture that is being used to study this form of learning. The architecture is a highly distributed system that achieves 'structural' learning through the application of a particularly powerful form of natural constraint.

Keywords - Non-Temporal Prediction, Distributed Problem solving, Society of Mind, Models of Learning, Skill Acquisition.

Introduction

Progress in VLSI techniques along with the emergence of some highly distributed architectures such as Fahlman [1] and Hillis [2] has awakened an interest in examining what can be done with certain architectures based on simple 'neuron like' processors, such as Hinton [3 4] and Feldman [5]. A theory of learning based on the principle of non-temporal prediction has been developed that is completely data-driven [6 7]. In this paper, we describe an experimental architecture that is being used to study this form of learning.

Non-Temporal Prediction

Learning and memory can be viewed as mechanisms for the acquisition of knowledge. Knowledge itself can be viewed as a means of predicting events in the world. Our survival is in a large part dependant on our ability to 'predict' the world. It is supposed that learning has evolved to meet this need. Making predictions about the world can be classified into two broad categories. First, there is the class of predictions that are time related. An understanding of 'Gravity' might be classified in this way, to understand 'gravity' is to predict that when a thing is dropped it will fall to the ground (or the class of predictions of which that is a simple example). This form of prediction is time related because the two defining events (the dropping and the hitting on the floor) are disparate in time. The second category, to which this paper is specifically addressed, concerns predictions that are unrelated to time. This kind of prediction concerns the classification of events. Here, learning the concept of an 'arch' (say) is making a prediction about what objects constitute 'arch'. When examples of arches that conform to this prediction are encountered they will be recognised as such, just as dropping an object that subsequently falls to the ground is recognised as indicating the presence of 'gravity'. The difference, is that the second category is unrelated to time. There are several reasons why it is useful to make this form of distinction.

(1) Many theories of learning and forgetting are based upon the notion of trace decay. Recency explains certain observable phenomenon, but is difficult to justify computationally and gives rise to some serious problems when dealing with predictive situations of vastly disparate times.

(2) Many of the effects for which recency was proposed can be adequately explained without reference to 'time' or 'trace decay'.

(3) Many problems that at first appear to be temporal in nature can be expressed in terms of the non-temporal paradigm. It is not known whether all situations can be transformed in this way. It may be that learning for 'temporal' situations is itself a learned strategy, there is some evidence to support this conjecture.

(4) It is possible to solve the problem of non-temporal prediction computationally in terms of a highly distributed architecture of simple processors.

Learning by Modification

The notion that learning usually takes the form of modifying an existing skill is intuitively attractive. Many attempts at capturing this intuition computationally have been tried, STRIPS [8] employed an augmented triangle table that allowed old plans to be 'modified' to suit new situations, an idea recently extended by Carbonell [9]. Minsky [10] discussed a form of learning in which new agents arise by 'splitting off' from old ones, with only small changes and essentially the same data connections. The mechanism presented in this paper follows the spirit of Minsky's 'agent splitting' but differs in detail. The architecture presented differs in that instead of splitting a single process (by copying) and then modifying the copy, it supports multiple copies of (almost) identical agents. Learning involves taking a 'suitable subset' of these agents and modifying it. Before describing the architecture itself, we should make a few points regarding the significance of this difference.

(1) It seems likely that natural systems such as the human brain can support this form of 'redundency'.

(2) Having multiple copies introduces a degree of 'fault tollerence', in particular, the 'Grandmother Problem' does not arise.

(3) Most significantly, having many copies means that a data driven mechanism can be utilized to achieve the 'split' instead of needing a top down decision to split.

Understanding Discontinuous Changes in Capability

Instead of having a single agent that can perform a given task, the architecture supports many such agents. We will refer to a set of similar agents as a process-set. The agents of a process-set compete to influence the state of the system. Each agent provides its own prognosis and some indication of how reliable it believes this prognosis to be (based on a simple probabilistic analysis). One agent's prognosis will be chosen as the most credible alternative. The computation of credibility will also be computed on the basis of a simple probabilistic analysis. Instead of hypothesizing that when a thing is learned its strength gradually increases, or when it is forgotten, it gradually decreases (trace decay), this model of learning distinguishes several phases of learning. First, the agent is generated in isolation (we will demonstrate one algorithm for agent creation when expounding the details of the architecture). Then, the agent must be refined

(discover its own boundaries and be able to accurately compute the reliability of its own prognosis). Finally, the agent must be discovered by other agents already in the system. This final stage is one in which the agents credibility is computed as the result of a probabilistic analysis, and corresponds closely to the notion of forming K-lines expounded by Minsky [11]. When a new and necessary agent is created, its success causes its credibility to rise until enough samples have been obtained to raise its credibility to a level above that of the previous 'favorite' agent for this task. At this point, the new agent will suddenly be used in place of the previous favorite, giving rise to an observable discontinuous change in performance.

The Experimental Architecture

The experimental architecture can be described at several levels. At one level, is the general system topology defined by a number of intuitive connectivity restrictions described in [7] and in more detail in [6]. Space prevents a discussion of this aspect of the architecture. The heirarchy can be decomposed into neighborhoods of agents that will, for the purposes of this paper be totally connected (the overall heirarchy allows the connectivity complexity to be kept linear despite the total connectivity within neighborhoods, furthermore, the connectivity within a neighborhood can be relaxed [2] without loss of generality). A neighborhood contains two computationally distinct components. The processors, that may be programmed to compute a predictive rule, and the creators that program processors for the purposes of generating new agents (learning) and replenishing process-set size when process splitting has resulted in an insufficient process-set cardinality (housekeeping). We will discuss the creator and the processor objects seperately. A programmed processor will be refered to as an agent.

The Anatomy of a Neighborhood

Consider a neighborhood to be a two dimensional sheet of processing elements. Each processor in the region recieves an input from outside the neighborhood, being totally connected each processor also recieves inputs from the outputs of every other processor in the neighborhood.

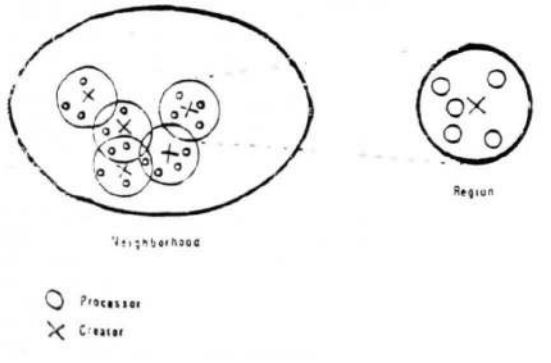


Figure 1

The neighborhood itself is divided into smaller overlaping 'regions' (See Figure 1). Each Region contains a single creator and a large number of processors. The creator has access to all local inputs to the processors within the region, and the outputs of each processor in the region. The creator can cause one or more of the processors in its region to be re-programmed.

Computation performed by a Creator

The creator monitors both the inputs local to the region and the number of processors that respond to the input. If too few processors respond to an input, the creator selects the processors that are least successful and re-programs them so as to increase the process-set cardinality. The creator is continually performing the following sequence of computations.

(1) Compute the activity of the inputs to the region. This involves counting the number of active inputs locally. Let the activity be denoted by activity.

(2) Compute the response size. This involves counting the number of processors in the region that responded to the inputs. Let the response size be denoted by response.

(3) Compute the expected response size. In the present system, the expected response size is a linear function of the activity.

(4) If response<expected, re-program response-expected processors. This involves choosing the required number of processors, the least successful ones are chosen first. Each processor keeps a record of its success. In our implementation, each region keeps a sorted list of processors, when n new processors are required, the first n are taken from this sorted list. In a truely parallel system such as might be found in Biological systems, this process can be achieved simply by broadcasting a re-program command to all processors and using a system of inhibition to prevent re-programming of the better processors (for a development of this idea see [6]).

Processing Inputs

It is convenient to describe the operation of the processors in two stages. First, how each input to a processor is handled on an individual basis, and second how these inputs are combined to form a prognosis.

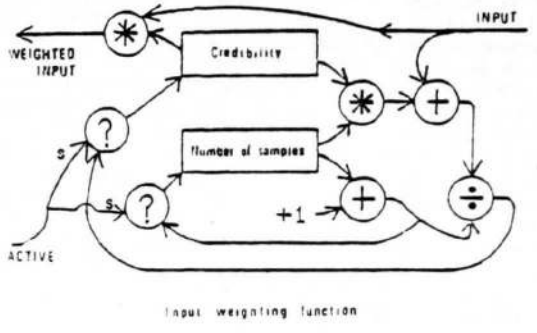


Figure 2

Each input to a processor is processed by an input weighting function. Figure 2 illustrates the function of this process. Each input weighting function (corresponding to $input_i$) samples its input whenever the process is active. In this way, the input weighting function computes for its input, the credibility that that input is indicative of the event being diagnosed -- the probability that the input will be active when the event is diagnosed $P(input_i \mid this.agent.active)$.

Other Processor functions.

Once the inputs have been weighted according to their credibility, they can be combined to form the prognosis.

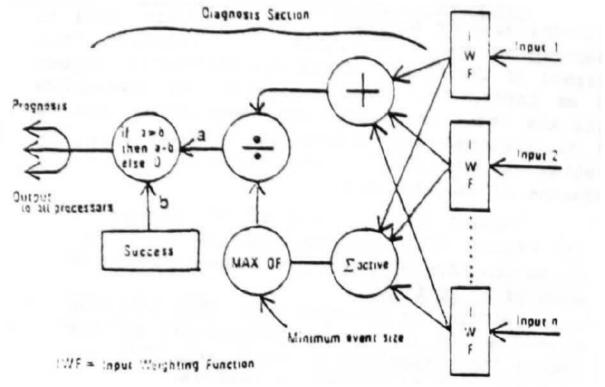


Figure 3

Figure 3 illustrates the basic operation of the diagnosis part of a processor. The value of success is adjusted whenever the agent is active. Space prevents further development of this idea here, however, low success values indicate failure in implementing a predictive rule, and such processors will be re-programmed by their creator when a new agent is required. The inherent limitations of simple Linear Threshold devices such as the prognosis function, are used as a powerful natural constraint. This guarentees that most agents that are created will eventually die (success will fall until it is eventually re-programmed). This gives rise to a very ecenomical use of processors without the need for a knowledge driven resource (processor) allocation system (these ideas are developed in detail in [6]).

Conclusion

Due to a lack of space, many significant details and much of the theory had to be omitted. Experiments with a LISP based implementation of the system outlined in this paper have been encouraging. Complex structural descriptions can been learned by the system. The system is robust in that usually, no agent is so important that its removal will be critical (due to duplication), and a high degree of noise can be tollerated. An analysis of the systems noise immunity can be found in [6]. It is interesting that as the regions approach saturation (most processors are successfully programmed as agents), it becomes increasingly difficult to learn a new rule. This is because, before a new agent can achieve a respectable success it is re-programmed by its creator because it is still the least successful agent. Only intensive training will result in the new agent being learned, and this will be at the cost of one of the other successful agents. Full details of the architecture, and justification of its design can be found in [6].

References

1. Fahlman,S.E.
NETL: A System for Representing and Using Real-World Knowledge The M.I.T. Press. Cambridge Massachusetts 1979. ISBN 0-262-0609-8

2. Hillis,W.D.
The Connection Machine
M.I.T. AI Memo 646 September 1981.

3. Hinton,G.
A Parallel Computation That Assigns Canonical Object-Based Frames of Reference
Proceedings of IJCAI-7 1981.

4. Hinton,G.F. & Anderson,J.A. (eds)
Parallel models of associative memory.
Hillsdale, NJ: Erlbaum, 1981.

5. Feldman,J.A.
A Connectionist Model of Visual Memory
In [4] above.

6. Robertson,P.
Process Dependant Localized Memory
University of Texas at Dallas Technical Report.

7. Robertson,P.
Non-Temporal Prediction: A Distributed System For Concept Acquisition
Proceedings of the Fourth National Conference of the CSCSI/SCEIO 1982

8. Fikes,R.E. & Nilsson,N.J.
STRIPS: A new Approach to the Application of Theorem Proving to Problem Solving
Artificial Intelligence Journal, Vol. 2, no.3/4, 1971

9. Carbonell,J.G
A Computational Model of Analogical Problem Solving
Proceedings of IJCAI-7. 1981.

10. Minsky,M.
Plain talk about Neurodevelopmental Epistemology
Proceedings of IJCAI-5. 1977.

11. Minsky,M.
K-Lines: A Theory of Memory
In 'Perspectives on Cognitive Science' Donald A. Norman ed.

THE LOGIC OF EVENTS

John M. Morris
Measurement Concept Corporation
Rome NY 13440

Some of the earliest work in the logic of events appears in Hempel [2]. Here are some examples of what he meant by "event": "the first solar eclipse of the twentieth century," "the eruption of Mt. Vesuvius in A.D. 79," "the assassination of Leon Trotsky," "the stock market crash of 1929." The events are whatever these phrases refer to.

Events occur in both time and space, but the edges of the event may be fuzzy. An event like the collapse of the German economy during the 1920s or an increase in tension between Russia and China is not the sort of thing that can be confined to a definite region of space-time. Still, even though the location is vague or fuzzy, it always makes sense to ask <u>where</u> and <u>when</u> an event is located. The German banks, bankers, and householders that fell victim to the economic collapse were located in Germany; and the increase in tension between Russia and China includes editorials, posters, speeches, military movements, and the hearts and minds of people at definite points within the two countries. Similarly, it makes sense to inquire <u>when</u> an event occurs, even when the time boundaries are fuzzy. So we can always include a place and time reference in our descriptions of events, even though the edges of the events may be blurred.

A major problem in the development of a logic of events has been a criterion of identity for events, that is a way of telling when two descriptions refer to the same event. A single set of objects in a single space-time segment may be involved in an indefinitely large number of events. A Russian soldier near the Chinese border squeezes the trigger of his rifle. Among the many events which occur are these: (1) various neurological and physiological events in the Russian's body, together with physical processes associated with the firing of the rifle, and the resulting physiological processes in the body of the Chinese soldier who is killed by the bullet; (2) an attack on a Chinese outpost; (3) from a psychological point of view, a Russian soldier's expression of his boredom, frustration, and contempt for the Chinese; (4) the first incident in a major Russian-Chinese war.

Some people, like Anscombe, would prefer to say that only one event has occurred and that we have given four different descriptions of it. Goldman and others have shown that these cannot be regarded as a single event [1]. His proof, which is very simple, is this: We may say that the Russian, in this example, <u>expressed</u> his boredom by firing his rifle; we say that the shooting <u>constituted</u> an attack on the Chinese outpost; and we say that the killing <u>became</u> an international incident because of later reactions to it. We would not speak in this way if all of these were descriptions of the same event, because the converse of these statements would not be true. We would not say that the soldier fired his rifle by expressing his boredom, or that an attack on the outpost constituted the shooting, or that an international incident became a killing. If (1), (2), (3), and (4) above were identical, then relationships among them should be symmetrical; but they are not. For this reason they are not descriptions of the same event.

The important thing is that it will be impossible to specify an event unambiguously simply by specifying the objects and the portions of space and time in and to which it occurred. Since an indefinitely large number of events may occur at the same point in space and time, we need additional specifications in order to describe an event uniquely.

Distinguishing among events is important for current events analysis, because different events will have different consequences. The psychological state of an isolated Russian soldier is likely to be unimportant to the current affairs historian; but the outbreak of a war along the Russian-Chinese border is of major importance. An effective system for current events analysis will identify the event in terms of its relevance to the historian's goals.

Suppose that, following the incident, a Chinese radio broadcast is heard to characterize the shooting as "inhuman butchery" and to describe the incident in other emotionally loaded terms. We can say (1) that the Chinese reported on the shooting, and (2) that the Chinese attacked the Russians as "butchers." Precisely the same broadcast, at precisely the same time, used the same set of words to perform both of these actions. But the event reported as (2) is more significant for the historian than the event reported as (1). From the historian's point of view (1) and (2) are different events.

In the symbolism developed by Jaegwon Kim [3,4] an event is represented by an expression of the form:

$$[(x_1, \ldots, x_n, t), P^n]$$

where $(x_1, \ldots, x_n)$ is an ordered n-tuple of concrete objects, P^n is an n-adic empirical attribute, and t is the time at which $(x_1, \ldots, x_n)$ is said to exemplify the attribute P^n. The n-tuple of objects may be written in vector notation as X_n. The event is said to "exist" if and only if X_n does exemplify P^n at time t. (The place can be included among the x_i.)

Thus $[(x_1,x_2,t),P^2]$ might signify the event of an Israeli F-4 Phantom-II aircraft flying over the Suez Canal at 4:06 a.m. on August 4, 1982. Here, x_1 represents the aircraft, x_2 the Suez Canal, t the time, and P^2 the attribute of overflying. (The superscript "2" indicates that it is a two-place predicate.) It may seem somewhat strange to speak of an event like "overflying" as an attribute, but this generalization makes the symbolism applicable to states, conditions, and other qualities, as

well as to events.

A problem of particular importance for the designer of an event logic will be that of determining when two descriptions refer to the same event. In the example just given, when we receive a dozen reports of an F-4 flight over the Suez Canal, we will want to know whether there was just one flight or a dozen flights. Goldman and Kim propose a rather strong criterion of identity for two events: [(x,t),P] = [(y,t'),Q] if and only if x=y, t=t', and P=Q. This makes "flies over the Suez Canal" a different event from "threatens Egyptian frontiers." From a pragmatic point of view, the role of these two descriptions in an information system will be different, and we will take them as representing different events, even though the physical objects and their raw, physical motions are the same.

The description of the flight as a "threat" depends on the context of world events in which it takes place. Although the flight is located in the area of the canal, its significance is not located there at all. The significance of the flight is in the various government officials whose attitudes make it a threat. It would not be a threat if it were not for these attitudes. The claim that the threat is located only along the flight path is what Whitehead called the "fallacy of simple location".

A complete analysis of the logic of events will provide us with rules for going from one event description to another. We will want to know, for example, how to go from "Israeli plane flies over Suez Canal" to "Israel threatens Egyptian frontier." Border violations are events that can, in the aggregate, provide evidence for a current historian that tension is rising between two countries.

To show how the logic works, consider the following hypothetic event. Let us suppose that a Soviet officer at the Chinese border, one General Sayev Andronovich, is promoted to Field Marshall. In itself, this event does not have any clear significance for the historian. However, if we add the information that Andronovich is noted for his outspoken anti-Chinese attitudes, then his promotion becomes a significant predictor for future Soviet-Chinese relations. At least two events have taken place: (1) a Soviet officer named Andronovich has been promoted; and (2) anti-Chinese attitudes have been encouraged in the USSR.

Now, if we know that Andronovich is anti-Chinese in attitude, then we know that he belongs to the class of anti-Chinese Soviet officials. Our event logic should permit us to say that anything which happens to Andronovich is also an event which happens to an anti-Chinese official of the USSR. From this, it should be possible to derive the more general event, in which anti-Chinese attitudes have been encouraged. Finally, from this event, it should be possible to predict deterioration of Soviet-Chinese relations. The role of the logical apparatus is to provide the hypotheses upon which the historian can predict the deterioration of relations.

In an automated system for current events analysis a central problem will be to determine, from a general description of an event, which properties are going to be significant -- which properties are "constitutive" of the particular event, and which are merely "exemplified" by the event. It is just conceivable, for instance, that the historian is collecting the names of Soviet officers that begin with the letter "A" -- for some obscure reason we can only guess at -- and the important information is the first letter of the name of the new Field Marshall. (This would be part of the historian's "user view," the viewpoint from which he or she would want to look at the data.) The first letter of the name would be constitutive of the significant event (in the sense that it would be that which makes it signficant), and the political attitudes of the Marshall would then be nothing more than irrelevant noise.

The problem is in distinguishing the significant or constitutive features of an event. For human observers there is little difficulty in locating just those features of an event which are relevant to their interests. One fascinating characteristic of human perception is the way in which humans fail to notice elements in a situation which have no interest for them. For an automated information system, however, the problem of relevance becomes acute, because the machine has no interests of its own. We must be able to tell the machine how to locate those features in the information which will be useful in discriminating among relevant patterns of events [5].

In summary, the problem for analysis is determining those features, among the infinite number of features which can be extracted from the world around us, which will be significant for the goals of the current historian -- such as the detection of a potential world conflict.

REFERENCES

1. Goldman, Alvin I., A Theory of Human Action, Englewood Cliffs: Prentice-Hall, 1970.

2. Hempel, Carl G., Aspects of Scientific Explanation, New York: Free Press, 1965.

3. Kim, Jaegwon, "Events and Their Descriptions: Some Considerations," Essays in Honor of Carl G. Hempel, Nicholas Rescher, et. al., editors, Dordrecht: D. Reidel Publishing Co., 1970, pp. 198-215.

4. Kim, Jaegwon, "Causation, Nomic Subsumption, and the Concept of Event," The Journal of Philosophy, Vol. LXX, no. 8, April 1973, pp. 217-236.

5. Morris, John M., "The Need for Context in Event Identity," Third Annual Conference of the Cognitive Science Society, 1981, pp. 197-199.

FUZZY SEMANTIC NETWORKS: A NEW KNOWLEDGE REPRESENTATION STRUCTURE

BY:

DOUGLAS D. DANKEL II
KENNETH W. SPRAGUE

COMPUTER AND INFORMATION SCIENCES
UNIVERSITY OF FLORIDA
GAINESVILLE, FL 32611

ABSTRACT

This paper introduces a new method of knowledge representation called a fuzzy semantic network (FUSEN). FUSENs were created to model continuous or fuzzy knowledge using concepts from artificial intelligence, fuzzy set theory, and cognitive psychology.

FUSENs have the ability to model three theories from cognitive psychology: the theory of natural categories, the family resemblance theory, and the feature-set theory. They can also perform as most of the knowledge structures from artificial intelligence and as a fuzzy set structure. Presented is their structure and several examples illustrating their use.

INTRODUCTION

To have a complete understanding of an entity one must be aware of how it acts, what rules apply to it, and in what situations one might expect to find it. For example, it is possible to describe the color, shape, size, and subparts of a 'dog'. It is easy to define the sets to which 'dog' belongs and the memebers of the set called 'dog'. But, the concept of 'dog' is not complete unless one knows what 'dog's do and how they act. There should be specific memories of 'dog's. There should be anticipations of what to expect from 'dog's in general and from specific 'dog's in particular. There must also be an understanding of time, space, and the physical reality in which 'dog's operate. A complete concept of a 'dog' includes all of this knowledge.

FUSENs divide this complex knowledge into four separate classes: entities and categories; actions and processes; literal and deep sentences; and rules and hypotheses. This paper examines the first of these classes and briefly discusses the relationships between FUSENs and three theories from cognitive psychology: natural categories, family resemblance theory, and feature-set theory.

STRUCTURE

Figure 1 shows the graphical representation of FUSENs. The owner label defines the owner of a head node and the type label defines the association existing between the node. The weights represent the association strengths between nodes. A head node can be associated with any number of sub-nodes. Each instance of a head node and its sub-nodes is called a fuse. All nodes of a fuse can be sub-nodes or head nodes of other fuses.

Figure 2 is a fuse representing a set of attributes for the category 'fruit'. This is determinded by examining the head node name, 'fruit'; and the type label '(attrib)'. The type label is a reserved work, denoted by the surrounding parentheses, describing the relationship between the sub-nodes and the head node. '(Attrib)' defines all the sub-nodes as attributes of the head node name 'fruit'. The owner label defines the parent node(s) of the head node. This label resolves any ambiguity created when two or more fuses have the same head node name. For example, if two fuses have the head node name of 'color', one would look at the owner label to see what they referenced. There could be fuses concerned with automobile colors, leaf colors, or colors is general. In Figure 2 the owner label is '()' or null. This means this fuse is about 'fruit' in general.

Each sub-node is a different attribute of 'fruit'. The weights associated with each sub-node reflects how strongly that particular attribute is associated with 'fruit'. The link labels define the domain over which the sub-node is defined. In Figure 2 'red' and 'yellow' are defined as colors of 'fruit'.

The weights are viewed as frequency counts. In Figure 2 the head node weights of 137 states that 137 instances of 'fruit' have been observed. The ratio of the sub-node's weight to the head node weight is that sub-node's association strength. 'Red' has an association strength of 66/137 or 48.2%.

Figure 3 shows a fuse representing a set of apples attributes. The type label is '(attrib)', so the syntax of this fuse is the same as that of Figure 2.

NATURAL CATEGORIES

The theory of natural categories was developed by Rosch [ANDE80]. Natural categories are levels of abstraction that people seem to naturally develop and use. Rosch feels categorization occurs to go beyond insignificant individual differences and to obtain the most information from the smallest amount of categorization.

Figures 2 and 3 can be used as an example of natural categories. According to these figures, a certain object that is small, red, and sweet can be seen as an apple or a piece of fruit. Since these attributes match both the 'apple' and the 'fruit' fuses a computer algorithm would say the object is both an apple and a piece of fruit, which is correct. But, in communicating with humans, the algorithm will have to pick the most appropriate level of abstraction or as Rosch called it, the 'basic' level.

The way the algorithm can find the basic level is to look at the head node weight. The highest weight is the most frequently conceptualized concept or the basic level. In this example the object would be called an 'apple'.

FAMILY RESEMBLANCE THEORY

The family resemblance theory was also developed by Rosch [ANDE80]. This theory states

that every category is defined by an open-ended set of attributes or features. Natural categories have no fixed boundaries. For any particular category there might not be even one attribute in common with all the category members. An entity is judged to be a good member of a category if it has many attributes overlapping with the attributes of the category.

The FUSEN structure models this theory very well. The 'fruit' and 'apple' fuses show how the concept is defined by a set of attributes. The number of sub-nodes and their weights are dynamic and can constantly change as new examples of the category are observed. If a green fruit is observed, the sub-node 'green' with a weight of 1 will be added to the 'fruit' attribute fuse. In addition the 'fruit' head node weight will be incremented by 1.

FEATURE-SET THEORY

Feature-set theory [ANDE80] assumes people recall how frequently they have seen all the various attributes of a concept. The more frequently seen attributes have a higher correlation or association strength with the category.

This is exactly how fuses work. Figures 2 and 3 show two categories. The association strengths for each sub-node reflects how strongly it is associated with the head node. Notice that 'red' is more strongly associated with 'apple' than 'fruit', and 'tart' is more strongly associated with 'fruit'.

OTHER METHODS OF KNOWLEDGE REPRESENTATION

Sprague [SPRA82] has shown how fuses can also perform as many other knowledge structures. In particular he discusses production rules, semantic networks, expert knowledge systems, frame theory, fuzzy sets, and stimulus-response theory.

SUMMARY

This paper briefly introduces a new method of knowledge representation called a fuzzy semantic network. The theory is based on the idea that knowledge can be represented by the associations between symbols and that these symbols and associations can be explicitly represented by a semantic network. Using semantic networks as a base, a general method of knowledge representation was developed to include ideas from many areas: artificial intelligence, mathematics, psychology. It is hoped that when the complete syntax is developed FUSENs will be able to represent most any kind of semantic knowledge.

REFERENCES

[ANDE80] Anderson, J.R. 1980. Cognitive Psychology and Its Implications. San Francisco, CA: W.H. Freeman.

[SPRA82] Sprague, K.W. 1982. 'Fuzzy Semantic Networks'. Gainesville, FL: MS thesis University of Florida

Figures 2 and 3 on following page.

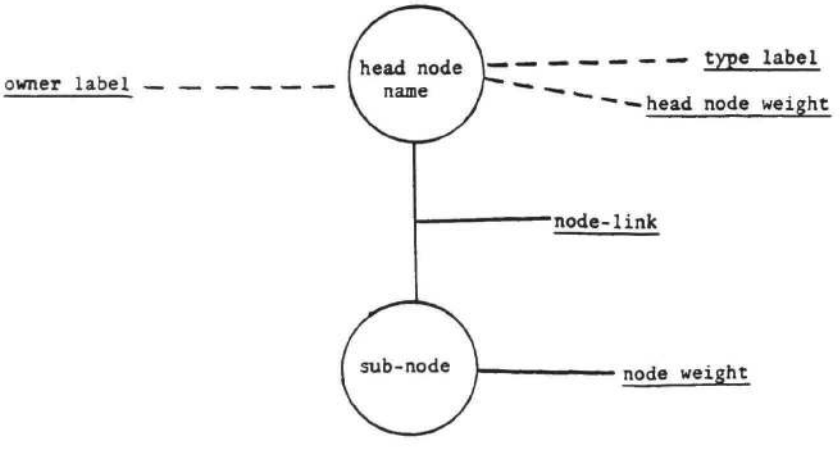


FIGURE 1. Diagram of FUSEN structure

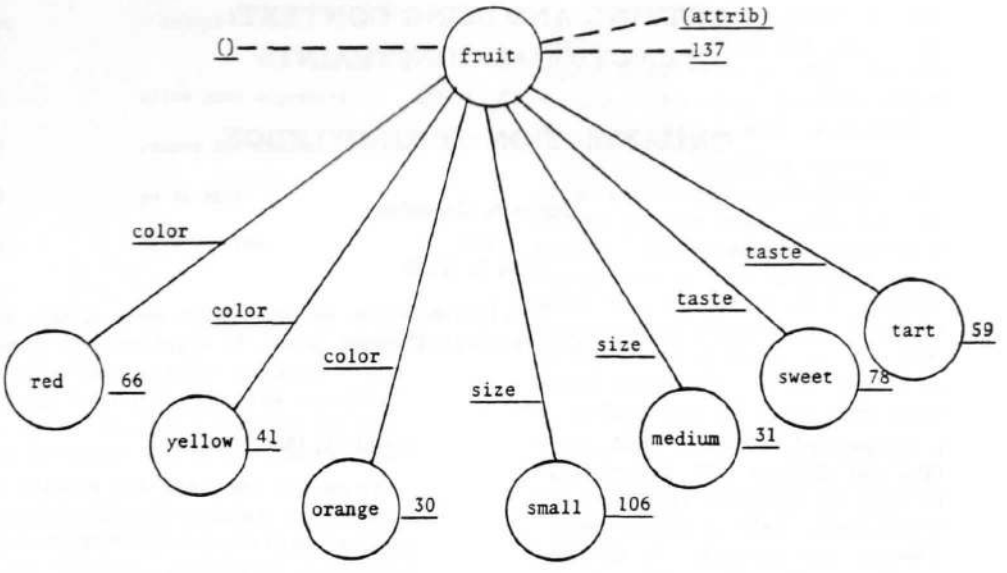


Figure 2. Example of fruit attribute fuse

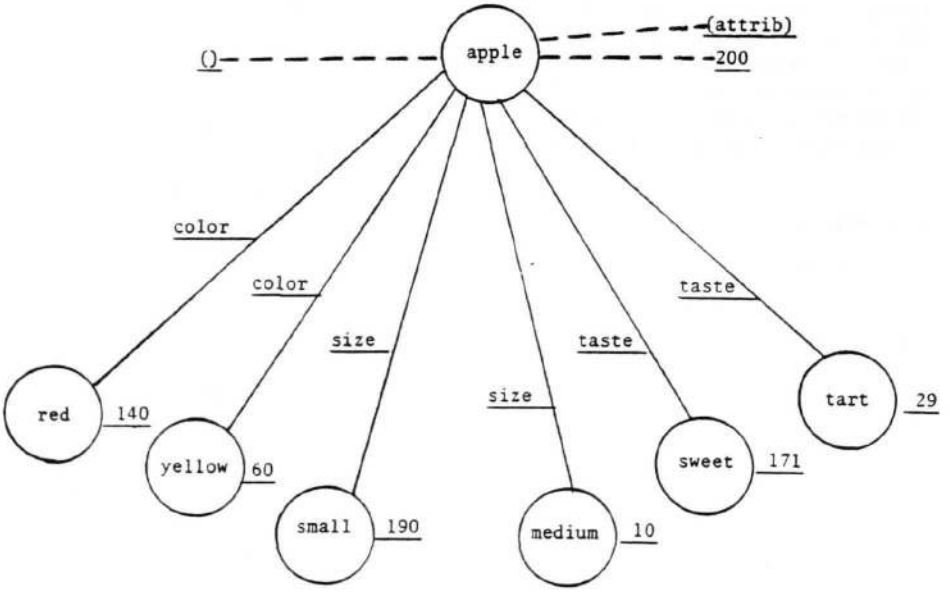


Figure 3. Example of apple attribute fuse

GETTING AND USING CONTEXT: FUNCTIONAL CONSTRAINTS ON THE ORGANIZATION OF KNOWLEDGE

James A. Galambos
and
John B. Black

Yale University
Cognitive Science Program

Studies of text comprehension (Bower, Black, and Turner 1979, Mandler and Johnson 1977, Schank and Abelson, 1977) have relied on the notion of a script or schema. A *script* represents world knowledge about common activities, events, and situations. It includes information about the components of these activities and the relations among the components. In this paper we examine scripts for common activities (e.g., cashing a check, or going to restaurants) as they exist prior to their instantiation in prose. The questions addressed in this paper are: How is the script knowledge structure accessed? and once the script is accessed how are its components made available? In other words, since context is so important in comprehension, we want to know how we get a context and once we have it, how does it help?

Schank and Abelson discuss how the script knowledge structure is activated during the comprehension of narrative. Clearly the easiest way to invoke a particular knowledge structure is to refer to it by name. Thus if the narrative explicitly mentions a situation, the retrieval of the knowledge structure should be straightforward. This can be done by a title of a passage, or by setting statements. We are interested in cases where the context is not given explicitly. Implicit reference to the activity can be made in a number of ways. For instance a goal mentioned in the narrative can serve as access cue for the script typically involved in accomplishing that goal. We are concerned with a different case where the presentation of one of the *actions* in the script leads to the accessing of the script itself. Thus on encountering the sentence:

John walked through the door

and saw the head waiter.

in a narrative, the restaurant script should be activated to contextualize subsequent sentences.

We test the claim that component actions will serve as access cues for their scripts if those actions are *distinctive* to the script. An action is distinctive to a script if that action is performed in few if any other scripts. Thus, for the restaurant script, the action SEE THE HEAD WAITER is highly distinctive, since it occurs in few if any other activities. The action of walking through the door occurs in so many activities that it is extremely low in distinctiveness to the restaurant script. This aspect of script structure has been developed and examined in Galambos, 1981 and 1982, and Galambos and Black, 1981.

We are also concerned with how the components of a script become available when the script is accessed. The question here is whether accessing the script makes all its components immediately available or whether some components have a more prominent status. In other experiments (Galambos and Rips, 1982), we have defined a measure of prominence called *centrality*. The centrality of an action is a measure of the importance of the action to the performance of the main goals of the activity. For example, in the restaurant activity the action EAT THE MEAL is highly central. Our hypothesis is that central actions should have a greater availability than less important actions when using an accessed context to aid comprehension.

Note that it is possible to select actions in such a way that these two dimensions are independent. The distinctive seeing the head waiter action is not particularly central to dining at restaurants, and the central eat the meal action is not particularly distinctive (since eating can occur in many other contexts; a plane, at home, a picnic, etc.). In terms of these dimensions our hypotheses are that the distinctiveness of an action should determine whether or not the script is accessed. The centrality of an action should influence whether the action becomes available when the script has been accessed. We designed a reaction time experiment in order to test these hypotheses.

The subjects' task was to decide whether or not two presented action phrases were components of the same activity. The first phrase was presented on a CRT screen for 1500 msec. This phrase then disappeared, and the second phrase was presented. The second phrase remained on the screen until the subject responded. The response latency was measured from the onset of the second phrase to the subject's response.

Four actions from each of 22 activities were chosen to sample the combinations of high and low levels of both centrality and distinctiveness. Thus from each activity one action (Hi-C/Lo-D) was high in centrality in the activity and low in distinctiveness, a second action (Lo-C/Hi-D) was low in centrality and high in distinctiveness. The third action (Hi-C/Hi-D) high in both centrality and distinctiveness, and the fourth was Lo-C/Lo-D. For example the four actions selected in the activity of cashing a check were:

Action Type	Action
Hi-C/Lo-D	write your signature
Lo-C/Hi-D	record the amount
Hi-C/Hi-D	go to bank
Lo-C/Lo-D	wait in line

Twelve pairs of actions were constructed for each activity by combining the four types of actions in all pairs at each order. These twelve conditions were equated for length and word frequency. The sequential presentation order of the two actions matched the real order of the actions for exactly half of the trials in each condition.

Stimuli were constructed for each subject so that all 12 conditions and all 22 activities were equally represented, but each action was presented only once. There were an equal number of negative trials using actions not involved in the positives. Twenty-four Yale undergraduates participated in the experiment.

The mean RTs for each of the twelve (positive) conditions were:

Condition	Mean
Hi-C/Hi-D --> Hi-C/Lo-D	873
Lo-C/Hi-D --> Hi-C/Hi-D	880
Lo-C/Hi-D --> Hi-C/Lo-D	964
Hi-C/Hi-D --> Lo-C/Hi-D	896
Hi-C/Hi-D --> Lo-C/Lo-D	1059
Lo-C/Hi-D --> Lo-C/Lo-D	963
Hi-C/Lo-D --> Hi-C/Hi-D	986
Hi-C/Lo-D --> Lo-C/Hi-D	1124
Hi-C/Lo-D --> Lo-C/Lo-D	1081
Lo-C/Lo-D --> Hi-C/Hi-D	1193
Lo-C/Lo-D --> Hi-C/Lo-D	1013
Lo-C/Lo-D --> Lo-C/Hi-D	1073

The nomenclature here is perspicuous; for example the first entry indicates that a highly central and highly distinctive action was presented in the first position followed (after 1.5 seconds) by a highly central but non-distinctive action, and the mean reaction time was 873 msec.

If we are right that distinctive actions access their script, then conditions where a distinctive action (Lo-C/Hi-D or Hi-C/Hi-D) is presented first should facilitate the response. This is because the script should be accessed in the 1.5 seconds before the second action is presented. Having the appropriate context should speed the interpretation and processing of the second action, as well as simplify the sameness decision. When the first action is not distinctive (Hi-C/Lo-D or Lo-C/Lo-D), then the script is not accessed and subjects must try to access a contextualizing structure when the second action is presented. This prediction is equivalent to a comparison of the first six and the last six means above. The first six contained a distinctive action in the first position (_ _ _/Hi-D). The prediction was confirmed. The difference between the two sets of means was significant [$min\ F'(1,35) = 7.31$, $p < .02$]. The context accessed by a distinctive first action does help subjects to confirm that the second action is in the same script.

Our second prediction involves the centrality of the second actions following distinctive first actions. Central actions are the main goals and components of the activity. This prominence should be represented in the organization of the underlying knowledge structure. When the script is accessed by a distinctive first action, central second actions should be confirmed more quickly as components of that script compared with less central second actions. This prediction is tested by a comparison of the first three and second three means in the list above (_ _ _/Hi-D --> Hi-C/_ _ _ vs. _ _ _/Hi-D --> Lo-C/_ _ _). In this case the *Min F'* was not significant but the F for the subjects was 4.83 which was significant at the .04 level for one and 23 degrees of freedom [for materials, $F(1,21) = 2.03$, $p < .18$]. Thus the claim that central actions are more available than non-central actions when the script is accessed also received a certain amount of support.

It is possible to examine more fine-grained predictions for these data. Perhaps the purest test of our assumptions can be obtained by comparing conditions Lo-C/Hi-D --> Hi-C/Lo-D and Hi-C/Lo-D --> Lo-C/Hi-D. This compares the same actions in different presentation order. Clearly the preferred order is when the distinctive (non-central) action is presented before the central (non-distinctive) action. The first action accesses the script and since the centrality of the second action makes it more available for confirmation. The reversed order should be much more difficult since the script is not accessed by the first action and second action is not prominent in the script. There is a very large difference (160 msec) in favor of the optimal order of these two action types. The point is that the optimal order is facilitative because it exploits the functional organization of the knowledge structure.

The results of this experiment indicate the presence of two functional constraints on the organization of knowledge about common activities. Knowledge structures (like the scripts examined here) are used to provide context to better understand experience. This implies that the knowledge structures can be quickly accessed when the need for them becomes apparent. When an isolated action is encountered it is necessary to find a context into which it fits. The organization of knowledge structures must reflect this necessity. The distinctiveness of an action to a script can be represented as a link to the superordinate script concept. If a distinctive action is encountered, then this link can be traversed and the script concept retrieved. If the action is not distinctive then either the retrieval path is unavailable or too many available scripts are accessed and the context is ambiguous. Distinctive actions then provide one way to find an unambiguous context. Our results demonstrate that distinctiveness is a relevant structural characteristic in the functional organization of knowledge structures for common activities.

A second functional constraint is that knowledge structures must organize information in such a way as to have the necessary components available for utilization

by the comprehension processes. In other words, having a context means (among other things) being able to generate predictions about subsequent input in order to lessen the processing load when that input is encountered. This constraint would be satisfied if a list of all information that could possibly be relevant to the context were activated when the context was retrieved. Alternatively, since some of the information in a contextualizing knowledge structure is likely to be more relevant, it might be that this more relevant information is more available or more easily accessed. Such relevant information might include the main goals of the activity and the most important actions in the performance of those goals. If the comprehension system can keep only a limited amount of information about a context available for prediction, then this information is probably the best sort to have. For instance, if the restaurant context is involved in a narrative then it is a very good prediction that subsequent input will include something about the action of eating. Our results indicate that this more central information does benefit from a greater availability once the context is accessed. Here again we have demonstrated an important aspect of the functional organization of knowledge structures.

In conclusion, we take this research to be a beginning in the specification the functional organization of information in knowledge structures for common activities. Furthermore, we think our results outline a theory of getting and using context in order to understand experience.

Acknowledgments

We are grateful to Robert Abelson, Kate Ehrlich, Brian Reiser, Scott Robertson, and William Salter for their help. This research was supported by a grant from the Systems Development Foundation.

References

Bower, G. H., Black, J. B., & Turner, T. J. Scripts in memory for text. *Cognitive Psychology*, 1979, **11**, 177-220.

Galambos, J.A. *Question answering and the plan structure of routine activities*. Paper presented to the American Educational Research Association Annual Meeting. New York, New York, March 1982.

Galambos, J.A. *Question-Answering and the Structure of Event Knowledge*. Paper presented to American Psychological Association. Washington, D.C., August, 1982.

Galambos, J.A. & Black, J.B. Why do we do what we do? *Proceedings of the Third Conference of the Cognitive Science Society*. Berkeley, California, 1981.

Galambos, J.A. & Rips, L.J. Memory for routines: just one thing after another? To appear in *Journal of Verbal Learning and Verbal Behavior*, August 1982, **22**, no. 4.

Mandler, J.M. & Johnson, N.S. Remembrance of things parsed: Story structure and recall. *Cognitive Psychology*, 1977, **9**, 11-151.

Schank, R. C., & Abelson, R. P. *Scripts, plans, goals, and understanding*. Hillsdale, N.J.: Erlbaum, 1977.

Conceptual Combination and Fuzzy Set Theory

Edward E. Smith
Bolt Beranek and Newman Inc.
and
Daniel N. Osherson
Massachusetts Institute of Technology

Conceptual combination is the process by which people combine existent simple concepts (e.g., brown and apple) into novel combinations (e.g., brown-apple). As a possible formalism for conceptual combination, most proponents of prototype concepts endorse fuzzy-set theory (e.g., Zadeh, 1965). Osherson and Smith (1981), however, argue that the amalgamation of fuzzy-set-theory and prototype concepts is fraught with problems.

Some fuzzy-set theory. A key notion in fuzzy-set theory is that of a characteristic function, which maps entities into numbers in a way that indicates the degree to which the entity is a member of some set or concept. To illustrate, consider the characteristic function, c_F, which measures degree of membership in the concept fish (F). When applied to any creature x, $c_F(x)$ yields a number between 0 and 1, where the larger $c_F(x)$, the more x belongs to F. Thus, our pet guppy may not be very typical of fish, so it gets a characteristic-function value of .80. Our pet dog will get a very low value, say .05. If we now consider pets (P), and its characteristic function c_P, then our guppy and dog might be assigned the values .70 and .90.

The issue of conceptual combination has often been reduced to a question about characteristic functions: namely, given that concepts P and F are combined to form the complex concept P&F, how do we specify P&F's characteristic function ($c_{P\&F}(x)$) on the basis of those of P and F ($c_P(x)$ and $c_F(x)$)? The answer from fuzzy set theory is that $c_{P\&F}(x)$ is the minimum of $c_P(x)$ and $c_F(x)$.

Applying this min rule to our pet guppy, g, yields

$$c_{P\&F}(g) = \min(c_P(g), c_F(g))$$

$$= \min(.70, .80) = .70$$

This says that our guppy is less typical of pet fish than it is of fish. And therein lies the problem. For as Osherson and Smith (1981) point out, intuition suggests that a guppy will be more typical of the conjunction pet fish than of either constituent, pet or fish. Osherson and Smith argue that this pet-fish example is just one of an indefinite number of counterexamples to the min rule.

Rationale for the present work. There are two problems with the Osherson and Smith (1981) counterexamples. First, they rest only on Osherson's and Smith's intuitions; such claims need to be tested against typicality ratings of naive subjects. Second, there is no indication of the generality of the failure of fuzzy-set theory; perhaps Osherson and Smith's counterexamples are of a few types in some underlying taxonomy of conjunctions, where other types might conform to the theory. To deal with these problems, we first present a taxonomy of adjective-noun conjunctions, and then describe some relevant experimental work.

An initial taxonomy of adjective-noun conjunctions. All counterexamples of the Osherson-Smith variety, such as pet-fish and brown-apple, have the following characteristics: the adjective concept (i.e., the property denoted by the adjective) is relevant to the noun concept (i.e., the object denoted by the noun) and negatively diagnostic of it; e.g., being brown is relevant to whether an object is an apple, and counts against it. More precisely, an adjective is negatively diagnostic of a noun to the extent that knowing that the adjective is a true description of some object increases the probability that the noun is a false description of that object, and knowing that the adjective is false of some object increases the probability that the noun is true of that object. An adjective is positively diagnostic of a noun to the extent that knowing that the adjective is true (false) of some object increases the probability that the noun is true (false) of that object. And an adjective is nondiagnostic of a noun to the extent that knowing that the adjective is true (false) of some object has no bearing on whether the noun is true or false of that object. Thus, in sliced-apple the adjective is largely nondiagnostic; in red apple the adjective is positively diagnostic; and in brown apple the adjective is negatively diagnostic.

In addition to the relation between the constituents, we also considered the degree to which the conjunction provides a true description of an object that is to be categorized. To keep things simple, we consider only the degree to which the to-be-categorized object manifests the property denoted by the adjective in the conjunction, and we let the object take either a high or low value on this

property. This gives a total of six cases, presented in Table 1.

Table 1

Initial Taxonomy of Adjective-Noun Conjunctions

Relation of Adjective Concept to Noun Concept	Degree to Which Object Manifests Property: High	Low
Nondiagnostic	(1) unsliced apple — object is unsliced	(2) unsliced apple — object is sliced
Positively Diagnostic	(3) red-apple — object is red	(4) red-apple — object is brown
Negatively Diagnostic	(5) brown-apple — object is brown	(6) brown-apple — object is red

Consider now how people might judge the typicality of various objects vis a via the different kinds of conjunctions in Table 1. In Case 1, since the constituent concepts are relatively independent of one another, people might separately judge the extent to which an object is an instance of the adjective concept and of the noun concept, and then combine the outcomes of these two distinct judgements into an overall typicality rating. Since this is the key idea behind fuzzy-set theory, some variant of the theory might prove adequate for Case 1. In contrast, Case 5, where the adjective is negatively diagnostic of the noun, captures the counterexamples used by Osherson and Smith (1981). Here, intuition suggests that an object with a high value on the property (e.g., an apple that is indeed brown) will be rated more typical of the conjunction (brown-apple) than of either constituent (brown or apple). The outcomes for the remaining Cases (2, 3, 4, and 6) might fall somewhere inbetween these extremes.

An experiment to test the taxonomy. For each of 48 pictured objects, one group of 20 subjects rated the object's typicality with respect to an adjective concept (e.g., red, brown, sliced), a second group of 20 subjects rated its typicality vis a vis a noun concept (e.g., apple), and a third group of 20 rated its typicality with respect to an adjective-noun conjunction (e.g., red apple, brown apple, sliced apple). The adjective-noun conjunctions were such that all six cases of our taxonomy were tested.

In the Noun group, on each trial the experimenter spoke the name of a noun, then a pictured object appeared and subjects rated how good an example it was of the noun concept. Each picture was presented once. In the Adjective group, on each trial the experimenter spoke the name of an adjective, then a pictured object appeared and subjects rated how good an example the pictured property was of the adjective concept. Now, each picture was presented twice, once with an adjective denoting a property that the pictured object had a high value on, and once with an adjective denoting a property that the picture had a low value on.; e.g., the picture of a red apple and that of a brown apple were presented once with "red" and once with "brown." In the Adj-Noun group, on each trial the experimenter spoke the names of an adjective and noun, then a picture was presented and subjects rated how good an example the pictured object was of the conjunctive concept. Each picture was presented twice, once with a conjunction whose adjective denoted a property the picture had a high value on, and once with a conjunction whose adjective denoted a property that the picture had a low value on; e.g., the picture of a red apple was presented once with "red apple" and once with "brown apple." All subjects had ten seconds to make a judgement, the judgements being made on a 10-point scale, where higher numbers indicated better examples.

The top half of Table 2 contains the data for the three cases of the taxonomy where the object has a high value on the property denoted by the adjective. For Case 1, we expected the minimum rule to work. The results are otherwise: the conjunction's typicality clearly exceed the minimum of its constituents. A comparable deviation from the min rule also occurred in Case 3. For Case 5, where we expected the largest violations of the min rule, the conjunctions' typicality exceeds the minimum value of the constituents by virtually half the scale! For all three cases, the deviation from the min rule is significant by a sign test.

Table 2

Typicality Ratings for Three Groups, Separately for Each Case

Cases	Adjective Rating	Noun Rating	Adj-Noun Rating	Adj-Noun Minus Minimum
a. Object Has High Value on Property				
1: Nondiagnostic	8.71	7.25	8.65	1.40
3: Positively Diagnostic	8.50	7.81	8.87	1.06
5: Negatively Diagnostic	6.93	3.54	8.52	4.98
b. Object Has Low Value on Property				
2: Nondiagnostic	.45	7.25	.52	.07
4: Positively Diagnostic	.02	3.54	.10	.08
6: Negatively Diagnostic	.81	7.81	.39	-.42

As for alternatives rules within fuzzy-set theory, none seem to do a better job. Gougin's (1969) multiplicative rule suggests that the conjunction's typicality rating should be less than the minimum value of the constituents, which is even wronger than the min rule. Another alternative is that the conjunction's typicality value be the average of its constituents, but this too is violated by the data (see Table 2). The best-fitting post hoc rule is that the conjunction's typicality is the maximum of its constituents. The max rule works well for Cases 1 and 3 but fails for Case 5; and it is not really a serious possibility in fuzzy-set theory for if conjunctive concepts are represented by a maximum then there is no obvious way to represent

disjunctive concepts.

The bottom half of Table 2 contains the results for cases where the pictured object had a low value on the property denoted by the adjective. For all three cases the min rule works well, but only because subjects in the Adjective and Adj-Noun groups judged the pictured objects to be nonmembers of the relevant concepts. Thus, when presented a picture of a brown apple and asked to judge its typicality of red or of red-apple, most subjects gave it 0 ratings. This floor-effect, which prevents us from taking the data in the bottom of Table 2 as a sensitive test of the min rule, reflects a poor choice of how to experimentally implement the extent to which an object instantiates the property denoted by the adjective. Thus, for the concept red, had we used pictures of red apples and reddish-brown apples, we might not have obtained so many 0 ratings for the concepts red and red-apple. This change has been made in our subsequent experiments.

In conclusion, for cases where an object "fits" a concept well, fuzzy set theory fails to provide an adequate account of conceptual combination.

References

Osherson, D.N., & Smith, E.E. On the adequacy of prototype theory as a theory of concepts. Cognition, 1981, 9, 35-58.

Zadeh, L.A. Fuzzy sets. Information and Control, 1965, 8, 338-353.

Natural Language Processing Using Spreading Activation and Lateral Inhibition

Jordan Pollack & David Waltz
Coordinated Science Laboratory
University of Illinois

Abstract

The knowledge needed to process natural language comes from many sources. While the knowledge itself may be broken up modularly, into knowledge of syntax, semantics, etc., the actual processing should be completely integrated. This form of processing is not easily amenable to the type of processing done by serial "von Neumann" computers. This work in progress is an investigation of the use of a spreading activation and lateral inhibition network as a mechanism for integrated natural language processing.

This work was supported in part by the Office of Naval Research under contract N00014-75-C-0612.

INTRODUCTION

It has long been thought that the modular decomposibility of language knowledge into syntax, semantics and pragmatics implied that language processing could be similarly decomposed; that natural language could be processed by first parsing the syntax, then fleshing out the meaning of a syntactic derivation tree, and finally (if we could ever get to this point!) attempting to interpret the speaker's intentions. Nowadays, it has become apparent that this processing is integrated in humans [Marslen-Wilson, 1980], and that it should, thus, also be in computer models [Schank & Birnbaum, 1980; DeJong, 1980]. However, the natural inclination of von Neumann computers to run one-step at a time presents a severe roadblock to the kind of integration needed for NLP.

What is needed is an integration mechanism sensitive to interpretation pressures from several directions. A promising approach would seem to be the use of a quantitative spreading activation / lateral inhibition network. This kind of network, similar in conception to relaxation techniques for low-level vision, and to neural network models, works through the iterative adjustment of real-valued node weights.

PREVIOUS AND RELATED WORK

The term "spreading activation" is almost as overworked as the term "frame," but most systems which spread activation do it in one of two ways: As marker passing intersection search [Quillian, 1968; Collins & Quillian, 1972; Fahlman, 1980], in which a parallel intersection search is simulated by binary marking of adjacent nodes in a breadth-first manner, or as quantitative weight balancing, [Ortony, 1974; McClelland & Rumelhart, 1981], in which activation energies assigned to all nodes are iteratively adjusted, based on local activation energies and strength of connections. One of the well-known dangers of spreading activation is its potential for overkill; an intersection search, under certain circumstances, may generate too many useless intersections, and quantitative adjustment may result in "heat death," where every node becomes activated. (A solution for this latter form of activation involves the use of decay, dampening factors, or the spread of negative energy - lateral inhibition.) Nonetheless, both forms of spreading activation display interesting behavior.

For example, the previously mentioned work by Collins and Quillian showed how spreading activation could account for aspects of human memory priming, while Fahlman's work demonstrated that many forms of problem solving could be simplified when an intersection search was computationally "free." Ortony, on the other hand, built a system for schema selection using damped activation, and McClelland and Rumelhart effected a close simulation of experimental results on human letter and word perception in context.

Other work in parallel approaches to natural language processing has been done by Small [1981] and Rieger [1977] in which the traditional practice of breaking down knowledge into syntax and semantics was turned on its head, and knowledge of all kinds was distributed to individual "word experts"; by Hendler & Phillips [1981] who are working on an ACTOR-based [Hewitt,1976] NLP system; and by Gigley [1982] in which a neurolinguistically-inspired NLP system capable of simulating aphasic behavior was built.

NATURAL LANGUAGE PROCESSING USING AN ACTIVATION/INHIBITION NETWORK

The authors of this paper are presently building a NLP system in which the knowledge sources are modular, but the processing is fully integrated. The integration mechanism is an activation/inhibition network similar in nature to the one used by McClelland and Rumelhart and described below.

An activation/inhibition network is a weighted directed graph, where node weights represent activation levels, and link weights represent strength of activation if positive, or strength of inhibition if negative. The process of spreading activation / lateral inhibition involves iterative recomputation of activation levels. At each cycle, every node receives a contribution from each of its neighboring nodes equivalent to the neighbor's activation level multiplied by the weight of the intervening link. This contribution (scaled to range between -1 and 1) causes a proportional change in the activation level of the node; a contribution of 1 zaps the node up to its (predefined) maximum activation level while a contribution of -1 saps the node of all its strength. Eventually, a static condition is reached where some nodes reach their maximum or minimum strength, while the rest of them receive contributions of 0. (For a complete mathematical formulation see Pollack [1982b].)

NETWORK CONSTRUCTION

An activation/inhibition network such as this

can smoothly model the flow of quantitative constraints up and down a multilevel system. For natural language processing, the main problem becomes how to build such a multilevel network. We feel that a proper network can be built through the judicious instantiation of network fragments which are represented in standard knowledge representation structures, such as frames [Minsky, 1975].

The frames in our system contain the knowledge of syntax, of semantic features, and of case roles, organized to efficiently generate pieces of network on demand. These frames are richly interconnected with activation and inhibition links, and constitute the general knowledge base of the system. When sentences are input, a temporary network is constructed out of fragments stored within lexically accessed frames. These fragments are organized into a network by the same sort of breadth-first operation used in a chart parser [Kay, 1973]. The resulting network has activation links between phrase markers and their constituents, and inhibition links between pairs of phrases that have common constituents. (So far, we have done the network building by hand.)

In more detail, the required actions are as follows:

First, there is breadth-first instantiation of nodes representing phrase markers, case roles, and expectations for other nodes. These expectations are triggered when lexical items or grammatical constituents are encountered, and consist of simple feature patterns to match and connection procedures to be carried out if the match occurs. Secondly, there is pattern-based connection whereby if a newly instantiated node matches a pattern, specific linkages are made. As an example of these these two processes, if a node of type NP is instantiated, it will then cause the instantiation of an expectation that a VP will occur; if a VP is found, an S is generated and connected to both the NP and VP. Of course, if more than one candidate for a pattern shows up, the two candidates are connected with an inhibition link, so one will eventually be eliminated.

The activation and inhibition processes reinforce nodes that are supported by activation links and inhibit those which are not, so, for example, expectations that are not quickly fulfilled will die. Furthermore, activation and inhibition are also happening in the background frame system by a purely word associative scheme, which helps prime good word senses (and aids in schema selection). Finally, nodes which become inhibited below a certain point are garbage collected thus keeping the active network as small as possible.

EXAMPLE OF OPERATION

Some preliminary results are presented here which demonstrate the feasibility of the activation/inhibition approach to NLP. As mentioned above, since the system is in its early stages, the networks presented were built by hand. We demonstrate how the system reacts to syntactic ambiguity, how a lexical preference can affect its behavior, and finally how semantic constraints can be integrated.

Consider, then, the following sentence, which, in the absence of any semantic knowledge, is syntactically ambiguous due to the lexical ambiguity of "up":

John ate up the street.

The hand-built network for this sentence is shown in figure 1 with arrows denoting activation links, and circles denoting inhibition links (following McClelland & Rumelhart). Note that each node in this network is suffixed by two numbers which denote the "span" [Hobbs, 1974] or sequence of words, of that node.

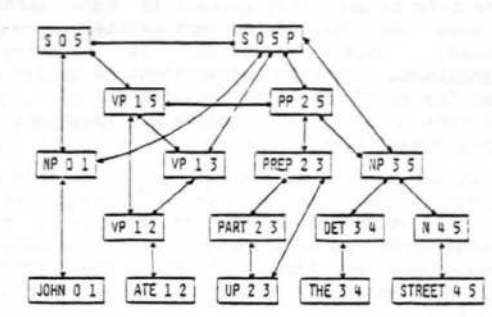


FIGURE 1 - SYNTAX ACTIVATION/INHIBITION NETWORK FOR "JOHN EATS UP THE STREET"

One would expect a robust NLP system to be confused by ambiguity but then to gracefully resolve it. This is indeed what happens. Figure 2 contains a graph of the activation levels over time for all the nodes in the network. Each node is depicted by a single letter, and each activation cycle by a horizontal row in the graph. When a letter traces a path to the left, it is being inhibited and when it moves to the right, it is being activated.

The most interesting node pairs to watch are B and C, the mutually inhibitory sentences, and G and F, the mutually inhibitory verb phrases:

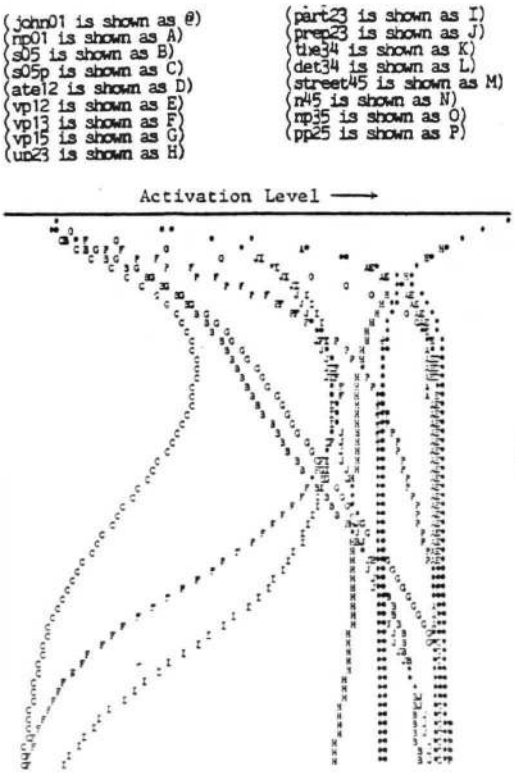


Figure 2 - "Confused"

B=(John) (ate (up the street))
C=(John) (ate up) (the street)
G=(ate (up the street))
F=(ate up)

The system is confused at first: B is more heavily weighted than C, so the sentence with the preposition is selected, while F is more strongly activated than G, so the verb-particle phrase is selected. This selection is, obviously, inconsistent. But then, after about 30 cycles, the system "decides" ("Look Ma, no homunculus!") on a consistent reading of "up" as a preposition, and weights G more heavily than F.

In the absence of semantic preferences (e.g. a preference for interpreting "street" as a location), syntactic preferences can play a role. Certain words do have lexical tendencies, as, for instance, the word "does", which is most often a verb, but which is also a plural noun, meaning several female deer.

Figure 3 demonstrates the sensitivity of an activation/inhibition network to syntactic preferences. The link strength from "up" to "particle" has been increased, corresponding to a lexical preference. Notice that the phrases related to interpreting "up" as a preposition (B, G, J, and P) become inhibited much more quickly this time.

However, when humans process this sentence, they also take into account the knowledge that "street" is a good candidate for a location, but a bad candidate for the object of eating. The next example demonstrates the sensitivity of our NLP approach to this semantic knowledge. Four nodes have been added and connected into the network. The verb phrase "ate" is linked to "ate-loc" and "ate-obj," and the verb phrase "ate up" is linked to

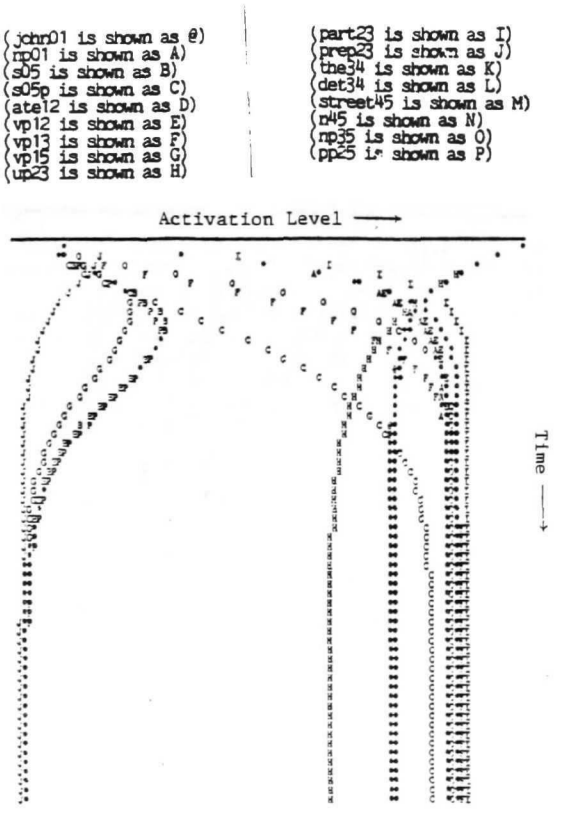


Figure 3 - Syntactic Preference for "Up" as Particle

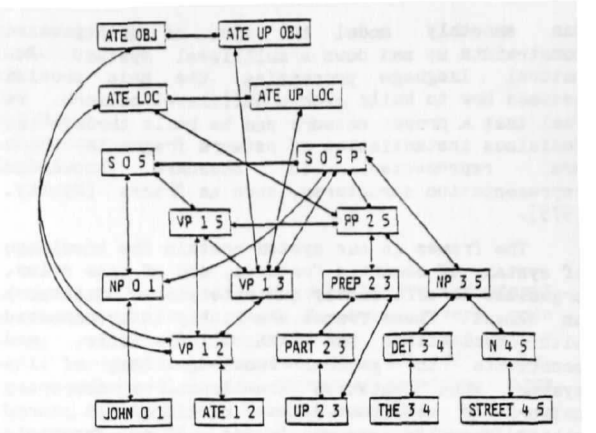


FIGURE 4 - SEMANTICALLY AUGMENTED NETWORK

"ate-up-loc" and "ate-up-obj." These nodes represent "cases" [Fillmore, 1968] of their respective nodes and are a subset of those that would be instantiated by our system. The pattern-matching connection component would connect the prepositional phrase "up the street" to "ate-loc" based on its span and on inherited features from "up" and "street".

The modified network is shown in figure 4, and figure 5 graphs the response of the activation/inhibition network to this new information. As one can see, after 15 cycles, all nodes related to interpreting "up" as a particle are being rapidly inhibited. (T, S, C, F, and I).

PROSPECTS

The results given above are interesting in that they demonstrate the sensitivity of activation/inhibition networks to slight

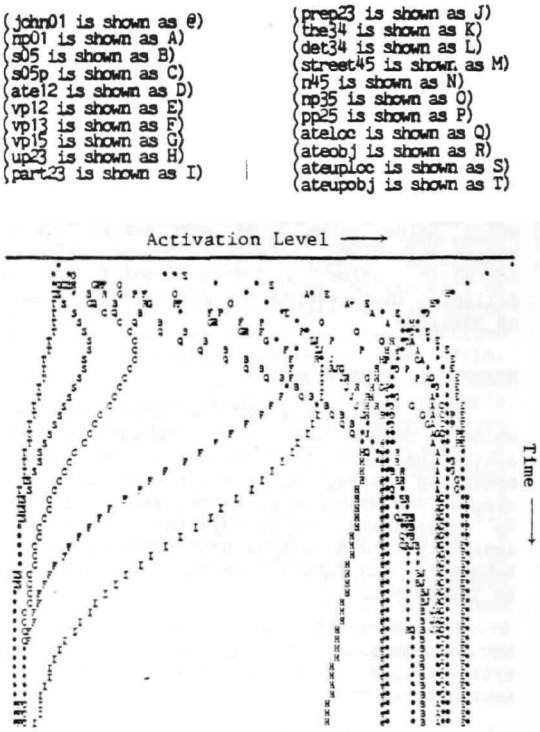


Figure 5 - Added Semantics

differences in knowledge. Currently we are working to complete the automatic instantiation and connection components of the system.

The use of a parallel and decentralized decision process can be brought to bear on many other interesting problems in NLP as well. For instance, there are indications that the timing and volume of spoken language both play useful roles in disambiguation [Wales and Toner, 1979]. A system based on activation and inhibition could be designed for sensitivity to these clues, since time is, after all, a crucial element in the activation/inhibition process.

Furthermore, the processing of garden path sentences, which are an interesting but not well-understood phenomenon in natural language, could quite possibly be handled by an activation/inhibition network. Marcus [1979] built a parser which attempted to account for garden-path sentences as a result of memory limitations. Unfortunately, there are garden path sentences his parser could (though shouldn't) handle [Milne, 1980], such as:

The prime number few.

Within the framework of activation/inhibition networks, garden path sentences would be accounted for by irreversible inhibition of expectations.

Also we have recently begun to consider ways of integrating a novel form of knowledge representation, "event shape diagrams" [Waltz 1982], to model certain kinds of metaphor understanding and adverbial modification. As an example, these methods should allow us to interpret sentences such as:

Robbie's metal legs ate up the space between himself and Susie.

as meaning a kind of PTRANS [Schank 1975].

Finally, a practical system based on activation/inhibition networks could be the starting point for new computing architectures. In this vein, [Pollack,1982] has designed a VLSI cell for parallel simulation of activation/inhibition networks, thus showing that a programmable set of logical connections (i.e. links) can be run on a machine with fixed and regular physical connections (i.e. wires).

CONCLUSION

The processing of natural language requires the sensitive integration of multiple sources of knowledge. A mechanism very likely to achieve this integration is an activation/inhibition network.

REFERENCES

Collins, A. and M.R. Quillian, 'Experiments on semantic memory and language comprehension' in L.W. Gregg (Ed.) Cognition in Learning and Memory, Wiley, New York, 1972.

Church, K. and R. Patil, "Coping with Syntactic Ambiguity or How to Put the Block in the Box on the Table," presented at 56th Linguistic Society of America meeting, December, 1981.

DeJong, G., "Prediction and Substantiation: A new approach for Natural Language Processing" Cognitive Science V.3 #.3 pp. 251-273, 1980.

Fahlman, S.E., NETL: A system for Representing and Using Real-World Knowledge, MIT Press, 1979.

Gigley, H., "Neurolinguistically Based Modeling of Natural Language Processing," Doctoral Thesis (in preparation), University of Mass, Amherst, 1982.

Hendler, J. and B. Phillips, "A Flexible Control Structure for the Conceptual Analysis of Natural Language Using Message-Passing," Technical Report 08-81-03, Texas Instruments, 1981.

Hewitt, C., "Viewing Control Structures as Patterns of Passing Messages," MIT AI Memo 410, 1976.

Hobbs, J.R., "A Metalanguage for Expressing Grammatical Restrictions in Nodal Spans Parsing of Natural Language.", Report NSO-2,Courant Institute, NYU, January 1974

Kay, M., "The MIND System", in Rustin (Ed.) Natural Language Processing, Algorithmics Press, New York, 1973. Marcus, M.P., A Theory of Syntactic Recognition for Natural Language, MIT Press, 1980.

Marslen-Wilson, W. and L. K. Tyler, "The Temporal Structure of Spoken Language Understanding," Cognition V.8 #.1 pp. 1-72, 1980.

Milne, R., "Using Determinism to Predict Garden Paths," DAI Research Paper 142, University of Edinburgh, 1980.

Minsky, M., "A framework for Representing Knowledge", in Winston (Ed.) The Psychology of Computer Vision Mcgraw Hill, New York, 1975.

McClelland, J.L. and D.E. Rumelhart, "An Interactive Activation Model of the Effect of Context in Perception", Technical reports 91 & 95, Center for Human Information Processing, UCSD, 1980.

Ortony, A., "SAPIENS: Spreading Activation Processing of Information Enclosed in Associative Network Structures", unpublished, 1976.

Pollack, J., "An Activation/Inhibition Network VLSI Cell", WP #31, Advanced Automation Group, Coordinated Science Laboratory, Urbana, January, 1982

Pollack, J., "An Activation/Inhibition Approach to Natural Language Processing", WP #35, Advanced Automation Group, Coordinated Science Laboratory, Urbana, April, 1982

Schank, R.C., "The Primitive ACTS of Conceptual Dependency", in Schank & Nash-Webber (Eds.) Theoretical Issues in NLP, ACL, Arlington, Va. 1975.

Schank, R.C. and L. Birnbaum, "Memory, Meaning, and Syntax," Research Report 189, Yale C.S. Department, November 1980.

Rieger, C., "Viewing Parsing as Word Sense Discrimination," in R. Dingwall (Ed.) A Survey of Linguistic Science Greylock, 1977.

Wales, R. and H. Toner, "Intonation and Ambiguity", in W.E. Cooper and C.T. Walker (Eds.) Sentence Processing: Psycholinguistic Studies Presented to Merrill Garrett, Erlbaum, New Jersey, 1979.

Waltz, D.L., "Event Shape Diagrams", To Appear in Proc. NCAI, Pittsburgh, August, 1982.

USING THE DANCE TO INVESTIGATE THE PRAGMATIC/SEMANTIC BOUNDARY BETWEEN ARTIFICIAL AND NATURAL LANGUAGES

Laura Silver
University of Pittsburgh
Lawrence J. Mazlack
University of Cincinnati

0.0 ABSTRACT

This work addresses the pragmatic and semantic distinctions between natural and artificial languages by the development of a context-free generative grammar to describe motions in modern dance. The dance is a particularly good vehicle as it conveys meaning, but is undescribed by a generative grammar. Whether or not a grammar describing dance motion can be considered to be for a natural or artificial language is unclear.

1.0 INTRODUCTION

There are two different kinds of languages: natural and artificial. Artificial languages have been developed to deal with formal systems of man-created knowledge. Natural languages enable naturally arising entities to deal with their environment. Generally, artificial languages deal only with truth or knowledge that is specific to their artificial environment.

Both the written and spoken forms of human speech are universally considered to be languages. Animals as well as humans appear to communicate with each other through body motions. Whether or not body motion should be considered a language is open to debate. Some workers believe that the term "language" should be narrowly defined to include only signaling systems which are capable of manipulating abstractions. Others, would consider any origized system of signaling to be a language.

It is agreed that whatever a language is, its construction and interpretation is constrained by a specification mechanism. In languages, the construction specification is called a grammar.

Precisely how humans come to know the grammar of a language is unknown. One group of workers holds that it is learned. The other group, believes that the capability is innate. Irregardless of how men come to know the structures of spoken language, they certainly are capable of learning the grammars of artificial languages, for example, automata.

Both artificial and natural language can carry meaning; i.e., have semanticity. However, the semantic information represented by artificial languages appears to be of a different type than that of the information carried by a natural language.

In order to develop an understanding of the pragmatic and semantic differences between natural and artificial languages, a generative grammer is being developed to represent dance generation. The developed grammar is artificial, that described appears to communicate naturally. Whether or not the dance is a language is open to question as the tokens of the dance are never abstractions.

The problem is to understand the nature of language: of how humans perceive, understand and represent their world in their semiotic system. It is of further interest to develop an understanding of the relationship between the natural language system and the artificial language systems also developed by humans. These systems are intentionally created in order to represent systematically in systems of signs other perceptions in the symbolic manner or representation. Systems of signs can be represented as systems of signification where perceptions exist within the plane of content and are represented by the symbolic plane of expression. Artificial language systems such as mathematics and logic, are usually referred to as symbol systems, however they too are language systems and function as semiotic systems, as they are formal systems of signification.

In order to extend the analysis of information distinctions between the semantics of natural and artificial systems have to be clarified as do the distinctions between information and pragmatics.

2.0 BACKGROUND

Language, communication and information are three tightly interwoven concepts. The problem of information representation and communication is the focus of this work.

2.1 Language

Language is a process or symbolization that enables signification of some thing by representing it by something else. The "thing" represented has an existential space-time reality; the representation is an abstraction of the reality. Meaning is derived from the relationship between the physical and the symbolic

2.1.1 Meaning

Language provides the capability to functionally relate symbolized meanings. However, language is more than individual relationships among the meanings. Words, which are symbols, recursively become things themselves as they are utilized. As things themselves, they can be used symbolically to express or represent concepts as the next order or abstraction. Signs, symbols, words, tokens, pictographs are the tangible products of the interrelationship between the thought and the referent. This interface provides an operational definition for the nature of the concept of meaning, the property of language defined as "semanticity." That is, "the property of being able to convey meaning" [LYON 79].

2.1.2 Semiology: An Analytic Tool

De Saussure defined language as the Semiotic system; i.e., the science of signs. The sign is a subsystem, or a component of the system of language. The principle of "signification" indicates the relationship between the thing signified (the signified) and the things signifying it (the

signifier). Signifiers exist within the "plane of expression" and signifieds exist with the "plane of content." This relationship expressed by the sign as:

sign = (signifier, signified)

which is a specific relation between the plane of physical reality or "content" and symbolic reality or "expression." More generally, the sign is defined as:

sign = (plane of expression, plane of content)

A language is considered to be comprised of a set or system of signs.

Semiology aims to take in any system of signs, whatever their substance, and limits; images, gestures, musical sounds, objects, and the complex associations of all these...constitute, if not languages, at least systems of signification" [Bart 9]. Seminology will be used as tool in the analysis conducted by their work.

2.2 Natural and Artificial Language

Whether or not artifically constructed languages, or language schemas, can be considered as language "proper" is not central to this work.

Semiology, although initially concerned with natural signalling or communication systems set the stage for the analysis of any system of signs, whether they be natural languages or artificial language systems.

In discussing languages, Carnap states

> "so long as we are concerned with building this language, and not with its application and interpretation respecting a given theory, the signs of our language remain uninterpreted. Strictly speaking, what we construct is not a language but a schema or skeleton of a language: out of this schema we can produce at need a proper language (conceived as an instrument of communication) by interpretation of certain signs." [CARN].

Cherry discusses the difference between the natural and artificial kinds of languages.

> By 'language' we shall mean those organically developed systems, whether spoken or scribed, by which humans transmit messages; but the work 'cipher,' or 'code,' will be used to mean any invented, self-consistent system, whereby one set of symbols may be transformed into another for certain special stated purposes" [CHER 93, 94].

This difference between "language" and "code" can be understood not as a difference in structure, but as a difference in development. The concept of language generally implies an organic or natural development, and consequently referred to as "natural" language. The concept of code implies an intentional development, and consequently if referred to as "artificial" language systems.

2.3 Analyzing Language

The language being observed is usually called the object-language. The language used to discuss the object-language is called the metalanguage. The semiotic of the object language is formulated in the metalanguage system. Carnap identified the semiotic analysis of the object language into the three components of syntax, semantics, and pragmatics.

The terms syntax, semantics, and pragmatics are somewhat ambiguously applied. Part of the ambiguity of these terms is a function of whether the analysis of the object language is either the natural or artificial form of language.

According to Carnap, syntax "attends strictly to the expressions and their forms." However, "syntax may include rules which determine certain logical relations between sentences, e.g., the relation of derivability" [CARN 79]. The inclusion of the property of derivability in the syntactic component blurs the boundary between syntax and semantics.

2.4 Differing Semantics: Descriptive and Logical

Both natural and artificial languages contain the components of syntax, semantics and pragmatics. In the artificial language system, semantics refers only to the expressions and their designations without reference to any particular external system. In the natural language system, semantics includes the analysis of meaning by pointing to referents in the extentional world. In essence, there are two kinds of semantics, which can be understood as depending on either context-sensitive or context-free grammar. In artificially or "logically" constructed language systems, the grammar is context-free. In natural language, the grammer is context-sensitive. This latter form of semantics could be referred to as descriptive semantics, following the terminological distinction that "descriptive linguists" do the analysis. Their analyses are context-sensitive, in that they include the pragmatic component of meaning. In contrast, the form of semantics pertaining to the artificially or logically constructed language system can be labeled logical-semantics.

2.5 Communication and Information

Languages can be both naturally developed and artificially created. At the semiotic metalevel form of analysis both forms of language are treated as object level languages as both fulfill the need to signify; i.e., to represent perceptions and abstractions. This process of signification is more generally known as communication, where the language serves as an instrument of communication.

In the analysis of the problems inherent in communication, workers such as Shannon and Weaver have identified three levels of difficulties which complicate the problem of identifying information, particularly at the semantic level: (a) accuracy of symbol transmission, (b) communication of meaning, and (c) effectiveness (how conduct is affected). These three levels are all concerned with the concept that is labeled information, yet which "information" applies is not consistent for all three levels. Level A uses "information" as the amount of signal transmission, where Levels B and C the "information" is the semantic and pragmatic sense.

3.0 PROBLEM DOMAIN

In order to develop a context-free information representation a domain other than human verbal communication had to be selected. Verbal communication is too context-sensitive. Rather than working with the ambiguities of human verbal communication where it is difficult not to be pragmatic, or with information system design where the objective is to be pragmatic, the information system of human movement communication was selected.

Just as linguists have attempted to develop the notation for natural language grammars, seeking to represent the logical-semantic component, a similar grammatical structure of human movement can be developed. Generally, the domain of human communication is categorized into the verbal and the non-verbal. The verbal includes both the verbal and written forms of natural language. The non-verbal includes everything that is not verbal communication. Within this large category of non-verbal, the domain of human movement communication has been selected in order to construct a formal grammar representing the logical-semantic component of human movement information.

3.1 Purpose

This research investigates the semantic component of the artificial language system. The concern addressed is the clarification between the descriptive-semantics with the context-sensitive grammar (CSG) representation, and the logical-semantics with the context-free grammar (CFG) representation. The purpose is to illustrate the separation of the logical-semantic from the pragmatic, in order to demonstrate that it is possible to separate the information structure from potential meaning. The CFG is a template, providing the structure for the set of possible constructions any eventual user could select in order to represent any intended meaning. Prior to the representation of meaning a structure has to be defined whereby meaning representation can be made possible. Just as the information system can be viewed as both a process and referred to as a thing, a grammar can also. It is a template and therefore a thing, but it is a dynamic processing structure.

3.2 Existing Systems Representing Human Movement

The representational systems for human movement are data systems in that they are bound to some pragmatic component and that they are context-sensitive. Each has a basic set of symbols representing units that the user needed to represent. Although each representation system identifies a variety of syntactical units, no logical-semantic or grammar has yet been developed. Thus there is as yet no representation for the process of human movement information.

3.2.1 Notations

Labanotation is one of the most widely used representational systems for notating human movement [DNOT] [HUTC]. The notations are syntactic representations specifying syntactic units: direction, level, timing, and areas of the body, which are represented by unique symbols. It is not possible to use the system for anything but the description of the movement in the units provided by the initial symbol set. There are no structures or rules indicating relations among the syntactical units to generate more complex units. Consequently there is no representation by a grammar expressing the information system that is communicated by the movement.

Other notation systems are similar [SILV]. The Eschol-Watchman, the Benesh, Kinesics, Choreometrics, to name only a few, only differ in the specific particularization of the syntactic representation.

Why is there no system for representing the human body and its movement apart from any context? Perhaps because in the development of the representational systems, distinctions between the syntax and the logical-semantic were never clearly understood.

3.2.2 Models and Simulations

The objectives of various designers of models and simulations of human movement have been to extend the representational facilities of the human by mechanizing the laborious task of describing and computing problems in human movement. The goal was to develop a computer graphic display of a human model [POTT] [BILL].

Another area of research was the development of an interactive graphic editor for Labanotation [BROW]. The objective was to use the computer to facilitate the laborious process of hand writing Labanotation. This work was extended as part of the development of a graphic simulation for human motion [BALD] [TRAC].

3.2.3 Recognizing What To Know

The visual aspects of the perception of movement are essential in the design of mobile robots and the context-sensitive forms of representations are usefull. However, a context-free form of representation is preferable prior to any context-sensitive (i.e. applied) form of representation. For example, the visual aspect can be specified as a context-sensitive situation, which subsequently can be defined using a context-free grammatical structure. Research on this problem is important not only for the solution to problems in movement understanding, representation and generation, but also to illustrate the context-sensitivity of systems.

If we wish to represent the dynamic process of information, research must be done on abstracting the information from the pragmatics of use of that information. The structure of the process of information must be represented prior to the pragmatic application of the information.

3.4 Separating Logical and Semantic Descriptive Structures

Human movement and human verbalization both have the association of meaning with the sensorial transmittable component of movement and of speech that is transferred as a product of the information system. Where natural language has the symbolic representational facility of the written form, providing another channel for the transfer of the information, the movement notations do not. Just as the information process of human verbalization has been grammatically coded, the aspect of the problem that first needs to be addressed is the definition of the logical-semantic, i.e. the formal representation of the grammatical structure to code the information transferred via human

movement.

Before the grammar can be context-sensitive, it needs a context-free form. The problem is to develop a way to write movement information such that context-sensitive meanings can then be communicated in a written symbolic form.

3.4 Communication Systems: Human Movement Compared to Natural Language

The semiotic system of human movement communication was selected as the domain in which to investigate representation of a natural activity in an artificial language.

Adequate representation of the generative structure of the semiotic system seems to be the necessary and sufficient conditions which linguists, anthropologists and philosophers require for "language" identification. Where natural language encompasses both verbal and written forms of the sounds and their meanings, movement language exists only with what can be equated with the verbal level of natural languages. A comparison of this difference would be equating the notations for movement with the phonological orthographic representations of the sounds of natural languages. Each natural language has particular orthographic symbols necessary to represent the sounds of that language, just as each form of movement has developed notational symbols to represent the visual perception of movements particular to that form. However, where verbal language has not only the particular phonologogical representation, it further has a representational form which is called the written from where the meaning in the experiential form can symbolically be represented.

4. INVESTIGATIVE STRUCTURE

The specific problem addressed is the development of a prototype for information representation, by experiment with representions of the logical-semantic structure of human movement information using the BNF form of the context-free grammar as the analytic tool.

It is posited that the situation in natural language representation is analogous to problems in information system design. Before the grammatical structure is constructed, comprised of the vocabulary elements of the system and the set of relations among them, particular referents to the units are assigned, building the context-sensivitity into the initial design of the system. The idea of the context-free form of representation preceeding any context-sensitive representation is the direction of this research.

4.1 Role or The Grammar

The grammar itself is a representational template, in that it does not contain the meaning, but rather provides a structure. The grammar is a process in that it is used as a template. It is a commodity in that it is a tool constructed for analytical and representational purposes. It is tied to a particular form of representation, but it is relatively context-free. Any language system is a particular form of a semiotic system useful to communicate a range of meanings, and sensitive to that range. (Whorf defined this concept as "linguistic relativity" [WHOR].) Yet, the same semiotic system is context-free, in that it has the feature of productivity, and can generate valid expressions to represent new meanings even in the extensional world to which it is bound.

Looked at in this way, a grammar exists at the meta-level, providing a form of analysis for an information structure for any possible object-level expression that is generated in that language system.

BNF was chosen to represent the grammar. It provides a method of notation with the capability to code information that is dense and non-linear. BNF also lends itself to consistency verification.

4.2 Scope: Context-Free Representation of An Information System

In the analysis of the communication or semiotic system of human movement that is to be represented, only the logical-semantic form of the semantic component will be considered in order to illustrate that context-free representation is possible when the coding is only of the information system rather than including the pragmatics of the communication system.

This reception of data as information by the receiver is the pragmatic component, which is added to the input data from the sender. Meaning is the result of the contextual processing of data given some information input.

In order to develop a context-free grammar for the logical-semantic of human movement information, a non-purposeful context needs to be examined, i.e., where the movement is not intended to communicate any meaning but where the units of movement are learned for the production of movement itself, which subsequently can be used in various contexts to communicate a variety of meanings.

4.3 Dance Units

Dance instructors teach the units of the movement language without any intended transfer of information other than how to produce the units of movement. The vocabulary of movement that is used for dance is a complex series of units, which are deriveable in terms of initial units, plus rules for connecting the various units. These more complex units are referred to as "combinations." The units and the combinations are the information communicated in dance instruction.

4.4 The Goal: A Movement Semantic

This methodology formally can be represented as an operation of the logical structure of the BNF grammar, operating upon the selected scope of the verbal channel of the domain of the information system of human movement yielding as a product a grammatical representation of the logical-semantic component of the information system.

The movement semantic will be the grammar derived from the operation of the template processing the logical-semantic structure of the information into a representational form. This product will represent the results of research of the representation of the dynamic structure of the information process in a grammatical context-free form. The form of representation is that of a formal logical system.

4.5 Verification

A form of logical verification can be accomplished

by using LEX, the lexical analyzer, and YACC, the compiler compiler of the UNIX operating system. One of the advantages of using the BNF notation is that the movement semantic being developed and the code that LEX recognizes are both in the context-free form which is based upon the BNF notation.

4.6 Project Summary

The project will: 1) represent a portion of the logical-semantic information structure of a selected domain of human movement information, 2) represent a prototype for a written code of a representational rather than an experimental human movement language where 3) the symbolic representation of human movement information is accomplished using a dramatical rather than a descriptive template. The aim is to define a subset of human movement information code that meets these criteria of the logical or artificial system, such that it can be used without the problems of contradictory and ambigious expressions that are inherent, for example, in natural language systems.

5.0 REFERENCES

BADL Badler, N. J., Smoliar, S. W., "Digital Representation of Human Movement," Computing Surveys, Vol. 11, No. 1, March 1979.

BART Barthes, Roland, Elements of Semiology, Layers, Smith (trans.), Hill and Wang, New York, 1968, c1964 Elements de Semiologie.

BILL Billings, M.P., Yucker, W. R., "The Computerized Anatomical Man (CAM) Model," NASA-CR-134043, MDC-G4655, CNT: NAS9-13228, Issue 23, 1970-71.

BROW Brown, M. D., Smoliar, S. W., "A Graphic Editor for Labanotation," Computer Graphics, Vol. 10, No. 2, Summer 1976.

CARN Carnap, Rudolph, Introduction to Symbolic Logic and Its Applications, Meyer, W. H., Wilkinson, J. (trans.), Dover Publications, Inc., New York, 1958, c1954 Einfubrung in die symbolische logik, Springer.

CHER Cherry, Colin, On Human Communication, MIT Press, Cambridge, Massachusetts, 1980, c1957.

DNOT The Dance Notation Bureau, Courses and Programs, New York, 1980.

HUTC Hutchinson, A., Labanotation, Theather Arts Books, New York, 1977.

LABN Laban, R., The Language of Movement: A Guide to Choreutics, Plays, Inc., Boston, 1974, c1941.

LYON Lyons, J., Semantics, Cambridge University Press, Cambridge, 1977.

OTTE Otten, K., "Basis for a Science of Information," Information Science: Search for Identity, Debons, A. (ed.), Marcel Decker, New York, 1974.

POTT Potter, T. E., Willmert, K. D., "Three-Dimensional Display Model," Office of Naval Research, July 1975.

SAVA Savage, G. J., Officer, J. M., "CHOREO: An Interactive Computer Model for Dance," International Journal of Man Machine Studies, Vol. 10, 1978.

SHAN Shannon, C., Weaver, W., The Mathematical Theory of Communication, University of Illinois Press, Urbana, Chicago, 1980, c1949.

SILV Silver, L. D., "Towards a Movement Lanugage: On the Representation of Movement Knowledge," manuscript, Interdisciplinary Department of Information Science, University of Pittsburgh, Pittsburgh, Pennsylvania, 1981.

TRAC Tracton, W. P., "GEL: A Graphic Editor for Labanotation with an Associated Data Structure," Movement Project Report No. 15, The Moore School of Electrical Engineering, The University of Pennsylvania, Philadelphia, August 1979.

WHOR Whorf, B. L., Language, Thought and Reality, The MIT Press, Cambridge, Massachusetts, 1979, c1956.

WHAT CAN PHILOSOPHY CONTRIBUTE TO THE STUDY OF NATURAL LANGUAGE PROCESSING?

Martin Ringle
Computer Science
Department
Vassar College
Poughkeepsie, NY
12601

For the past twenty years philosophers have observed the development of research in natural language processing (NLP) and have offered periodic critiques of both its methods and its goals. (See Bar-Hillel, 1964; Matson, 1976; Dreyfus, 1978; Searle, 1980; and Odell, 1981.) Much of the criticism has proven to be valuable and artificial intelligence workers such as Winograd (1980) and Woods (1981) have acknowledged the positive influence of philosophical input to their work.

A great deal of philosophical criticism of natural language processing (and of artificial intelligence in general) however, rests, on misconceptions about the actual goals and claims of this research. This is partly due to the fact that NLP workers have not explicitly established a set of methods and aims for their work; it is also due to the fact that there are actually a number of different goals which motivate NLP research.

The purpose of this paper is to spell out the different objectives in the field of natural language processing in order to identify the places where philosophical criticism is legitimate and useful as well as those areas where it is inappropriate. Hopefully, this analysis will be valuable to philosophers and AI workers alike.

Research in natural language processing can easily be misconstrued to be a concerted effort towards a single goal. In the simplest terms, this goal would be the implementation of a system whose linguistic powers matched those of a literate, native user of a natural language such as English or French. In fact, however, research in natural language processing is a loose amalgam of projects aimed at a variety of goals. Even though common research requirements exist, such as the development of techniques for parsing, inference, memory organization, and so forth, presuppositions, methodologies, and criteria of success differ in significant ways from one project to the next. It is somewhat misleading, therefore, to appraise or to criticize the theoretical foundations of natural language processing as a single enterprise. Yet some philosophers (e.g., Odell, 1981) have assumed that the principal goal of NLP is the unified goal just mentioned, and have proceeded to question the plausibility of the research on that ground alone.

Consider the following formulations of the aims of natural language processing research:

1. To design systems which will allow a user to perform some traditional operation(s) on a computer (such as database query) without thereby requiring the user to learn an artificial language or a set of formal constraints which must be applied to the use of natural language.

2. To design systems which will be capable of processing textual material in order to produce accurate summaries, reliable translations, (from one natural language to another) or stylistically acceptable prose.

3. To design systems which will permit the user to initiate and direct a dialogue, in natural language, in a particular topic domain, with the latitude and fluency available in ordinary human dialogue.

4. To design systems which will persuasively exhibit the full range of human linguistic abilities, such as reading, translating, paraphrasing, interrogating, conversing, and so on.

5. To design systems which are able to use and understand natural language in precisely the same way that people do.

6. To design systems whose workings provide us with an explanatory model of the structures and processes responsible for human language use and understanding.

The first four formulations involve pragmatic goals, the fifth represents an epistemological goal, and the sixth an explanatory goal. The vast majority of efforts in natural language processing fall into the category of pragmatic goals. (See Waltz, 1982 for descriptive surveys of recent NLP projects.) Most, in fact, are examples of the first or Type 1, goal. Systems such as LIFER (Hendrix, 1977), ROBOT (Harris, 1979) and LUNAR (Woods, Kaplan, and Nash-Weber, 1972), for instance, provide natural language front-ends which are used principally for database query. The research aims which motivate the construction of these systems (and others like them) are relatively modest insofar as the use of natural language is constrained by topic, vocabulary, syntactic breadth and user dialogue goals.

The Type 2 goal is slightly more ambitious, since the analysis, generation, or translation of text may require a system to deal with a broad range of topics, a large vocabulary, complex syntactic constructions, and the intentions of an author (or reader) which may be less than obvious. Progress towards this goal has not been as substantial as progress towards the first goal, but there are programs which can analyze and paraphrase text (DeJong, 1982), produce modest translations from one natural language to another (Wilks, 1973) and generate moderately smooth English prose from an internal semantic representation (Mann & Moore, 1981).

Serious efforts towards the Type 3 goal are very few in number and have appeared only within the past five years.

Systems in this category include SRI's TDUS (Robinson, 1980) and BBN's HWIM (Bruce, 1982). Neither these, nor other systems of this sort, have achieved a level of combined reliability and efficiency which would make them suitable for broad implementation. However, there has recently been

a great deal of attention turned towards this area and a greater effort to achieve this goal can be expected in the near future.

The Type 4 goal is one which has been popularized in science fiction and the lay press, but it is not cited by AI researchers as the rationale for any serious NLP programming effort. This is not to say, of course, that AI workers have not entertained the idea of such a goal as a backdrop for their activities. In the proper perspective, such a goal is analogous to the one which underlies physics (and the natural sciences) namely, the eventual discovery of all lawful relationships among natural objects. Physics, after all, is dedicated to the objective of ultimately explaining the universe in terms of quantitative laws. One does not, however, invoke this goal as the aim of any particular research project. Moreover, it would be absurd to try to criticize a particular line of research in physics by attempting to show that this long-range goal is untenable. Even if the universe is not ultimately knowable in terms of the principles of physics, the enterprise still provides us with an ever-increasing understanding of natural phenomena. The same holds true for research in natural language processing: Even if the long-range goal is unattainable --- and that remains to be shown---this does not affect the plausibility of the other three pragmatic goals nor does it invalidate the knowledge of natural language processing derived from programs designed to achieve those goals.

The fifth goal raises a completely different set of questions. Here we are concerned with the status of the performance rather than with the performance itself. In the case of pragmatic goals, the criterion of success is the degree to which a system is able to deal effectively with linguistic input (or output). The phrase "deal effectively with" may be interpreted differently for different applications, but in general it implies that the system is able to carry out a function which would involve use and understanding of natural language if performed by a human. The claim is not made, however, that the system actually uses or understands natural language itself. We can appreciate the point of this last statement by considering the following question: Can the pragmatic goals be pursued without pursuing the epistemological goal as well?

Some AI researchers would undoubtedly say 'yes' in answer to this question and would point to the success of natural language interfaces such as the one used in MYCIN (Shortliffe, 1976), which are not generally characterized as "language understanding" systems. Cautious researchers, such as Winograd (1973) and Leitner (1977) have emphasized the epistemological limitations of their programs by putting the word "understand" in quotation marks when using it to refer to their natural language systems.

Other researchers, however, freely speak of their programs as natural language understanders. Schank and Riesbeck, for example, go one step further and argue that natural language programs must be directed towards genuine understanding:

> Computer programs that attempt to replicate understanding without simulating the human understanding process are doomed to failure when it comes to very complex processes. Nowhere has this been clearer than in natural language processing (Schank & Riesbeck, 1981, p. 2).

The point that Schank and Riesbeck make is a crucial one. If we are concerned with a Type 1 pragmatic goal, then genuine understanding is probably superfluous. A Type 1 interface can be limited to such a well-defined area of natural language that we can design systems to "deal effectively with" the range of anticipated linguistic input by means of deterministic production rules, discrimination nets, or similar methods. But if we are interested in Type 2, 3, or 4 pragmatic goals, then we must accept the fact that the potential for novelty, diversity, and deviant usage of linguistic inputs may be so great that a system would be effective under such conditions only if it were able to process the meanings of those inputs. And this implies that it must be able to genuinely understand natural language.

It follows, then, that while the Type 5 goal may be irrelevant to the majority of pragmatic systems of the present (and recent past), it is essentially related to the development of the more ambitious pragmatic systems of the future. It is in this context that philosophical evaluations of natural language processing become relevant: by analyzing the conceptual requirements of genuine language understanding, the philosopher can illuminate the theoretical conditions which an NLP system must meet. Moreover, unless these conditions are met, the epistemological goal cannot be achieved and thus the more ambitious pragmatic goals cannot be realized. Whether or not AI workers explicitly view the Type 5 goal as a motivating force in their research, therefore, they must acknowledge its indirect relevance if they intend to pursue a Type 2, 3, or 4 pragmatic goal.

The Type 6 goal is one which has drawn a great deal of attention in artificial intelligence due to statements such as the following:

> We consider the theory and model of semantic nets to be a computational theory of superficial verbal understanding in humans (Simmons, 1973, p. 63).
>
> . . .[W]e shall describe a model of human language understanding that forms the basis for a set of computer programs. . .(Schank, 1973, p. 187).

Both of these statements were published nearly ten years ago and since then there has been a considerable change in the claims made for the psychological significance of AI programs. Nevertheless, some AI researchers (especially members of the Yale Group) still view the explanatory goal (Type 6) as a primary one, and some philosophers (e.g., T. Simon, 1979) still find the view to be worthy of criticism.

An argument to demonstrate the relevance of the Type 6 goal to the rest of natural language processing might go something like this:

> Genuine understanding is necessary for any natural language understanding system capable of achieving Type 2, 3, or 4 pragmatic goals.
> Genuine understanding can be achieved only by processing language in the same way that humans process language.
> A system which does things in the same way as humans do them can serve as a model for explaining human language processing.
> Therefore: Pursuit of goal Types 2 - 5 entails pursuit of goal Type 6.

There are, however, several problems with such an argument. The second premise asserts a "process-product" identity relation which is very

much open to dispute. There are numerous instances (e.g., the synthesis of urea) where an artificial process results in a substance, event, or function which is identical to a natural substance, event, or function in every respect save its mode of origin. It has yet to be shown that a cognitive ability, such as the understanding of natural language, can be produced only by employing exactly the same processes and structures which are involved in human language understanding. Indeed, it has yet to be conclusively shown that all human beings understand language by means of exactly the same processes and structures.

However, even if we accept the second premise ---under some interpretation of the phrase "in the same way"---the conclusion still does not follow. A program for natural language processing is not, itself, an explanation of anything. In order to be viewed as explanatory, the details of a program---its variables and data structures, its control structures, and so on---must be interpreted with respect to human processes and structures. Any program can be legitimately interpreted in a variety of ways, few of which will bear any relation to the concerns of human psychology. The explanatory value of a natural language program, therefore, is not inherent in the program itself but arises, rather, from the use which can be made of it by someone who is concerned with cognitive modeling. The use of AI programs to theorize about human language processes, in fact, is not AI research at all. It is a tool of cognitive psychology or, if one prefers, a methodological heuristic for a multidisciplinary investigation of phenomena such as discourse comprehension, text comprehension, and so forth. It does not follow, therefore, that research in natural language processing entails explanatory goals of Type 6; consequently, philosophical objections to AI programs as theories of human language abilities are irrelevant to the plausibility of AI research in natural language.

Of all the types of goals ascribed to natural language research, then, philosophical evaluation is directly pertinent only to the epistemological goal formulated as Type 5, above. More specifically, the only valid judgment philosophy can provide is one which says that "the concept of natural language understanding entails X, hence a system must (be, do or have) X or it will not be capable of natural language understanding." The only valid objection philosophy can make to natural language processing is that "a computer, in principle, cannot (be, do or have) X."

REFERENCES

Bar-Hillel, Y. "The Present State of Automatic Translation of Language." In F. L. Alt (ed) Advances in Computers, New York, Academic Press, 1964.

Bruce, B. C. "Natural Communication Between Person and Computer." In Lehnert & Ringle, pp. 55-88.

De Jong, G. "An Overview of the FRUMP System." In Lehnert & Ringle, pp. 149-176.

Dreyfus, H. WHAT COMPUTERS CAN'T DO, Revised Edition, New York, Harper & Row, 1978.

Harris, L. "Experience with ROBOT in Twelve Commercial Natural Language Database Query Applications," IJCAI Proceedings, 1979, 6: 365-368.

Hendrix, G. "The LIFER Manual," SRI Technical Note No. 138, 1977.

Lehnert, W. & Ringle, M. (eds) STRATEGIES FOR NATURAL LANGUAGE PROCESSING, Hillsdale, NJ, LEA, Inc., 1982.

Leitner, H. "The Determination and Conceptual Structuring of Restricted Domains of Discourse for 'Intelligent' Interactive Systems," SIGART Newsletter, 1977, 61: 51-52.

Mann, W. C. & Moore, J. A. "Computer Generation of Multiparagraph English Text," American Journal of Computational Linguistics, 1981, 7: 17-29.

Matson, W. SENTIENCE, Berkeley, California, University of California Press, 1976.

Odell, S. J. "Are Natural Language Interfaces Possible?" *IBM Systems Research Technical Report* TR73-024, 1981.

Robinson, A. "Understanding Natural Language Utterances in Dialogs About Tasks," SRI Technical Note No. 210, 1980.

Schank, R. "Identification of Conceptualizations Underlying Natural Language." In Schank & Colby, pp. 187-247.

Schank, R. & Colby, K. (eds) COMPUTERS MODELS OF THOUGHT AND LANGUAGE, San Francisco, W.H. Freeman, 1973.

Schank, R. & Riesbeck, C., INSIDE COMPUTER UNDERSTANDING, Hillsdale, NJ, LEA, Inc., 1981.

Searle, J. "Minds, Brains, and Programs," The Behavioral and Brain Sciences, 1980, 3: 417-457.

Shortliffe, E. H. COMPUTER-BASED MEDICAL CONSULTATIONS: MYCIN, New York, North Holland, 1976.

Simmons, R. "Semantic Networks: Their Computation and Use for Understanding English Sentences." In Schank & Colby, pp. 63-113.

Simon, T. W. "Philosophical Objections to Programs as Theories." In M. Ringle (ed) PHILOSOPHICAL PERSPECTIVES IN ARTIFICIAL INTELLIGENCE, New Jersey, Humanities Press, 1979.

Waltz, D. "The State of the Art in Natural Language Processing." In Lehnert & Ringle, pp. 3-32.

Wilks, Y. "An Artificial Approach to Machine Translation." In Schank & Colby, pp. 114-151.

Winograd, T. "A Procedural Model of Language Understanding." In Schank & Colby, pp. 152-186.

Winograd, T. "What Does it Mean to Understand Language?," Cognitive Science, 1980, 4: 209-241.

Woods, W., Kaplan, R. & Nash-Weber, B. "The Lunar Sciences Natural Language Information System: Final Report," BBN Report No. 2378, 1972.

RECOGNIZING HUMOR IN NEWSPAPER CARTOONS BY RESOLVING AMBIGUITIES THROUGH PRAGMATICS

Lawrence Mazlack
Noemi M. Paz

ABSTRACT

Newspaper cartoons can graphically display the results of ambiguity in human speech. The result can be unexpected and funny. Captioned cartoons derive their humor from a sudden incongruity which can be made to follow by a human being who can automatically use stored world knowledge to resolve the ambiguous situation.

Likewise, computer analysis of natural language statements also needs to successfully resolve ambiguous situations. Computerized understanding of dialogue that takes place between humans must not only include syntactical and semantical analysis, but also pragmatical analysis. Pragmatics consists of an understanding of the speaker's intentions, the context of the utterance, and social implications of polite human communication.

Computer techniques have already developed been use restricted world knowledge in resolving ambiguous language use. This paper illustrates how these techniques can be used in resolving ambiguous situations arising in cartoons.

1. THE GENERAL ROLE OF PRAGMATICS IN NATURAL LANGUAGE UNDERSTANDING

Within linguistic theory, the study of language use can be called pragmatics. One defintion of pragmatics developed by Charles Morris (1946) is that pragmatics can be characterized by the relationship between signs and their human users. Signs fall into three classes: icons, indices, and symbols. Pragmatics relates directly to signs that are indices because indices can only be understood when they are actually used.

The meanings of indices can be found by describing rules for relating the sign to a context. These are pragmatic rules which are in essence "action" rules for "finding" relationships. The set of structures developed for describing these rules are called pragmatic-semantic "trees" and divide into three catesories:

1. Performatives - which describe the speaker's intention or goal in using a sentence as a question, a command, etc.

2. Presuppositions - assumptions about the context that are necessary to make that sentence verifiable, or appropriate, or both.

3. Conversational Postulates - a class of presuppositions concerning the nature of human dialogue which can be referred to as discourse codes of conduct. (Bates,1976)

For semantic-pragmatic structures to operate, it is not enough to determine the meanings of individual words. Other types of information must be accessed. In order to select between competing meanings, knowledge is required about the grammatical functions represented by particular word orders in the natural language sentence. Also, knowledge about the "real world" (presuppositions) is needed; i.e., the context in which the utterance took place.

Along with contextual knowledge, a semantic-pragmatic structure needs to account for "speech acts" (performatives). Speech acts demonstrate the speaker's goal. They can be a command, a question, a statement, etc. In other words, the associated meaning and the implied action must be understood. The theory must account for the fact that the listener or reader of the statement understands this double "meaning". (Bates, 1976)

Next, the semantic-pragmatic structure must explain the speaker's ability to understand sequences of language which should mean one thing but clearly mean another (conversational postulates). It is assumed that normal human beings who enter into a conversation have agreed to be cooperative. This means speakers will tell each other the truth, that they will only offer information assumed to be new and relevant to the listener, and will only request information which they sincerely want. This represents a set of standard rules. Deviations from this "code of conduct" will be seen as violations.

2. SUPPLYING PRAGMATICS FOR COMPUTER ANALYSIS

There are several question answering systems that make use of various techniques for including pragmatic analysis in understanding a natural language. As these techniques are described, a cartoon will be analyzed to illustrate how pragmatic analysis could be used to disambiguate the situation. The characters that correspond to the computer and those that correspond to a human in a man-machine dialogue will be identified. These cartoons are found in the Appendix.

2.1 COOPERATIVE DIALOGUE

The CO-OP System (Kaplan, 1979) is a question answering data base system that follows the "codes of conduct" presented earlier. Its objective is to provide cooperative responses from a natural language data base query. Some examples from this system follow.

CO-OP is able to determine from a

question not only what information is required, i.e. the direct, literal, and correct response, but also that the questioner is unaware of highly pertinent facts not explicitly requested in the question. A heuristic used by the system is knowledge of such facts frequently makes asking the question unnecessary, because they entail an answer. The system action is to ignore the question and provide the pertinent fact. For example:

> Question: How many students failed CSE110 in Spring, 77?
>
> System's answer is: CSE110 was not given in Spring '77.

The answer of "zero" would not have been cooperative.

The user who posed the above question presumed that the CSE110 class was taught in the Spring of 1977. The system on finding that this presumption was false responds with a "corrective indirect response" by supplying the negated presumption.

Cartoons often lead to funny results when their statements are ambiguous. In the cartoon TIGER (appendix, fig. 1) there is an example of a cooperative response in the answer to the question: "Did he catch him?". Prior to this question the human in the dialogue only knows there is a chase going on and that there are two participants: Stripe and Mrs. Parker's cat. The situation is ambiguous because we do not know who is chasing whom. Here the common human presumption is dogs usually chase cats and therefore Stripe must be a dog. The computer system on finding this presumption to be false, could respond with a corrective indirect response: "Nope, Stripe got away" rather than with the direct answer of "no".

Another type of cooperative response the CO-OP system replies with is a suggestive indirect response. Following the "codes of conduct", it is appropriate for an answer to contain relevant information likely to be requested in a follow-up question. A heuristic used here is to change the focus of the original question and to respond with a direct answer to the original question, but with the focus changed. The focus is that aspect of the question which is most likely to shift in a follow-up question.

The BC cartoon (appendix, fig. 2) could have used a suggestive indirect response. In this cartoon, a census taker (the human) is asking questions of a subject (the computer). A common presumption here is the first question the census taker will ask is the subject's name. The computer needs to realize the ambiguous situation created by the question: "Name, please". If a change of focus is analyzed the question could be seen as two questions:

> 1) What is your (subject's) name, please?
>
> 2) What is your dog's name, please?

The computer on realizing the possible ambiguous situation could then respond:

> 1) Fido is my dog's name.
> OR
> 2) "Computer" is my name.

2.2 RESTRICTED DOMAIN OF DISCOURSE

Another data base system, ROBOT (Harris, 1978), uses the data base itself to find the use of words in the question, to build expectations, and to resolve ambiguities. The system interprets input based on what makes sense within its limited world model, the data base.

In processing ambiguous statements, several interpretations may arise. A heuristic used is unintentional interpretations of input questions are usually not false for the specific domain, but have a vacuous response (Coles, 1972). To use this heuristic these interpretations are posed to the data base as queries. If all interpretations fail to find a response to the question, then the answer is "there aren't any". This negative answer assumes that the dialogue will only be about information contained in the data base. If more than one interpretation can be answered successfully, then the system enters into a clarification dialogue, just as humans would have to do when faced with an ambiguous question. If exactly one interpretation that is found, then the system responds using this interpretation for the question.

In the cartoon GARFIELD (appendix, fig. 3) the human speaker is asking Garfield to play with Nermal. The following ambiguous situation is created:

> 1) "play with Nermal" means that Nermal is a toy
>
> OR
>
> 2) "play with Nermal" means that Nermal and Garfield should play together.

The computer system could resolve this ambiguous situation by searching Toy and Friend domains for the entry Nermal. On not finding Nermal in the Toy domain but in the Friend domain the ambiguous situation is resolved.

The LIFER System (Hendrix, 1978) has capabilities for extending the natural language subset that is understood by the system. Users may employ easy-to-understand notions such as synonyms and paraphrases to extend the language. The users can then ask questions about information contained in the data base using their own natural language "style". In this way "utterances" by the speaker (user) can be understood by the listener (computer).

In the cartoon WIZARD OF ID (appendix, fig. 4) the human is stating to the Sire (computer) that "our records show there is a dip in unemployment" and is perhaps implying the question "what do we do next?". The ambiguous situation here is

that "a dip" could describe either a foolish person or a downward trend in a statistic or figure. To resolve this ambiguous situation the human could have entered the paraphrase:

> "Dip in unemployment" is a paraphrase of "temporary decline in the unemployment statistic".

2.3 WORLD KNOWLEDGE WITH FRAMES OR SCRIPTS

Many other systems have included world knowledge and information related to the dialogue with a user in a frame or script. Frames, simply put, are just highly structured sets for keeping pragmatics. When an action is carried out, some canonical description is stored in the frame that will permit the program to reconstruct the context in which the event took place. Frames carry over to the subsequent statements and conventions that mark anaphora or presupposition link the program to slots in the current or past frames that will resolve the reference.

Frames not only include syntactic information, eg. subject, object, prepositional phrases, but also semantic and pragmatic facts which provide various reasons, motivations and purposes not explicitly stated.

Scripts are like frames in that they also have empty slots that are filled with the context from a dialogue or text. However, scripts provide world knowledge about common experiences or situations in terms of Schank's conceptual dependency primitives. A text is understood by mapping sentences into actions or primitive acts as described in the script. Unstated facts described in a script but not in the sentence are assumed to be true. This provides a "background" or world knowledge for understanding and reasoning. (Schank 1975).

The BEETLE (appendix, fig. 5) cartoon can be used to illustrate the frame and the script concepts. Here the relative pronoun must be resolved in the phrase "that gun". The two possibilities are:

> 1) shoot at that gun, i.e. use that gun as a target
>
> 2) use that gun to shoot with and shoot at the original target.

The ambiguous situation can be resolved by using world knowledge about target practice. For example, it is helpful to know that targets should not be expensive, useful things, i.e. a gun, and that a target is not located near a human being. This type of information can be provided by demons in the case of frames or by the reason or goals statement in the case of scripts.

2.4 RECOGNITION OF HUMOR

Humor due to ambiguous statements requires the reader to recognize that an ambiguous interpretation has occurred. The humorous interpretation is the unexpected one. Through the use of pragmatic analysis, humorous interpretations can be recognized as well as generated.

SUMMARY

Cartoons can graphically represent the humor due to ambiguities in human speech. It may be possible to recognize humorous ambiguities using already existing techniques. Three levels of the use of pragmatics have been described. The first is to resolve double meanings by restricting the domain of discourse, eliminating the occurrence of double meanings. This device is employed by LIFER in restricting the language and by ROBOT by restricting the objects in the domain to only those that appear in the data base.

The second level of the use of pragmatics involves making the domain larger rather than restricting it. Frames and scripts are used to involve more world knowledge in the natural language analysis so as to disambiguate based on what is common occurrences in a given situation. The third level involves an ability to deduce from complex frames and scripts a purpose and then acting in agreement with that purpose. CO-OP is capable of determing the questioner's motive, and if necessary, posing for itself a question more in keeping with the questioner's motive than the original question.

Pragmatic devices used include proscribing context, enlarging context, and deducing motivation from context.

REFERENCES

Barr, A. 1980. "Natural Language Understanding", AI Magazine, Vol. 1, No. 1, Spring.

Bates, E. 1976. Language and Context: The Acquisition of Pragmatics, Academic Press, New York.

Intelligence: Can Computer Think?, Boyd & Fraser Publishing Co. San Francisco.

Coles, L.S. 1972. "Syntax Directed Interpretation of Natural Language", in Representation and Meaning: Experiments with Information Processing Systems. H.A. Simon and L. Siklossy (eds.) 44-66.

Elgin, S.H. 1979. What Is Linguistics?, 2nd ed., Prentice-Hall, Inc., Englewood Cliffs, NJ.

Goldstein, I. and Papert S. 1976. "Artificial Intelligence, Language and the Study of Knowledge", MIT, AI Memo. 337, March.

Grosz, B. 1980. "Utterance and Objective: Issues in Natural Language Communication", AI Magazine, Vol. 1 No. 1, Spring.

Harris, L.R. 1978. "Using the Data Base Itself as a Semantic Component to Aid in the Parsing of Natural Language Data Base Queries", Dartmouth College, TR 77-2, October.

Hendrix, G., Sacerdoti, E., Sagalowicz, D., and Slocum, J. 1978. Developing a natural language interface to complex data. ACM transactions of database

systems 3: 105-147.
Kaplan, S. 1979. cooperative responses from a portable natural language data base query system. doctoral dissertation dept of computer and information science, university of pennsylvania.
Mazlack, L.J. 1979. "Some Considerations in Mapping Natural Languages Queries onto Data Base Systems". Working Paper, September.
Schank, R.G. 1975. "The Structure of Episodes in Memory". Representation and Understanding Studies in Cognitive Science, D.G. Bobrow and A. Collins (eds.), Academic Press, New York.
Siklossy,L. and Simon, H.A. 1972. "Some Semantic Methods for Language in Representation and Meaning: Experiments with Information Processing Systems. H.A. Simon and L. Siklossy (eds.) 44-66.
Siklossy, L. 1978. impertinent question-answering systems: justification and theory. ACM proceedings annual conference Washing, d.c. vol 1 39-44.
Waltz, d July, 1978. an english language question answering system for a large relational database. CACM vol 21 no 7, 526-539.

APPENDIX

FIGURE 1

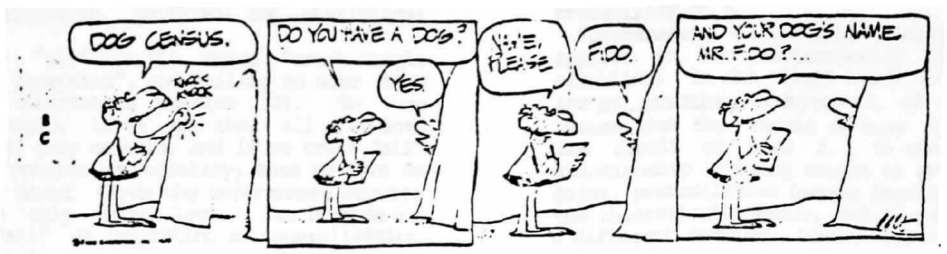


FIGURE 2

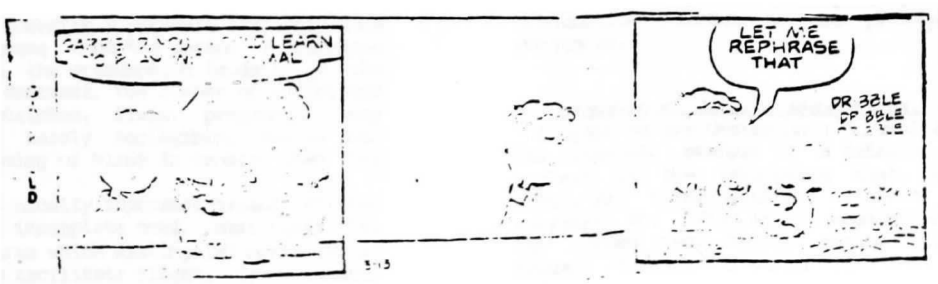


FIGURE 3

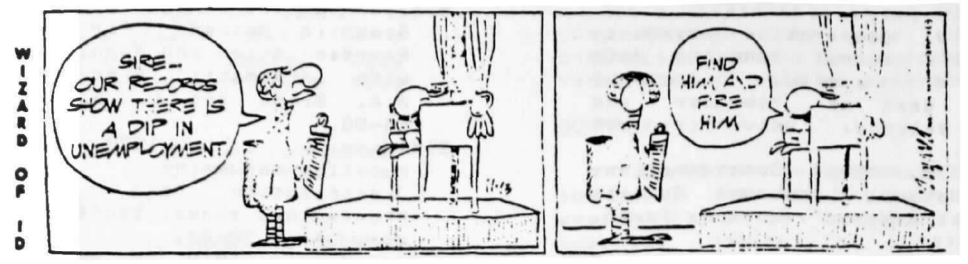


FIGURE 4

FIGURE 5

DEFAULTS REVISITED
or
"Tell me if you're guessing."

Jane Terry Nutter
Computer Science Department
SUNY at Buffalo
Buffalo, New York

ABSTRACT

This paper discusses default reasoning, distinguishing generalizations associated with defaults from both universals and statistical generalizations. I argue that conclusions based on defaults should be reported differently from conclusions which do not involve default reasoning, and that however we represent them, the related inference system must distinguish default claims from other propositions and treat them differently. Two existing analyses of default reasoning are briefly criticized in light of the distinctions presented.

1. Introduction.

A great deal of knowledge seems to take the form of generalizations: neither genuine universals, true of all things in their understood domains, nor simple statistical claims of "more than half", but claims which, although understood sometimes to fail, nevertheless warrant presumptions in the absence of conflicting information. Such generalizations are usually represented by defaults. This paper examines default generalizations, distinguishing them from universals and statistical claims, and pointing out some pitfalls their implementation presents. Especially, it behoves us to realize that answers based on default reasoning represent educated gueses; and however useful they may be, guesses cannot safely or honestly be handed out as facts.

2. Generalizations vs. universal and statistical claims.

In English, "all" rarely means "every single thing without exception", and failing to note this can produce unfortunate results [2]. To use Brachman's example, if we say that all elephants are four-legged gray mammals, and if we treat "all" as indicating genuine universality, then we have no way to talk about Clyde the unfortunate amputee elephant with only three legs. But suppose we always treat "all" as indicating a generalization Clyde the three-legged elephant, but unfortunately we can talk with equal ease about Clyde the non-mammalian elephant, or even about Clyde the non-elephant Indian elephant.

Generalizations cannot be treated like statistical claims either, although the difference here is more subtle. Most people realize that over half the population is female. Yet in the absence of information concerning a person's sex, one does not typically presume that the person in question is female (indeed, the presumption tends to go the other way!). By contrast, the number of flightless birds (emus, ostriches, kiwis, penguins, baby birds, etc.) is hardly negligible. Yet we feel justified in assuming of birds in general that they fly.

Generalizations usually represent causal claims, albeit masked and incomplete ones. Most birds fly, because the features which distinguish something as a bird evolved to facilitate flight. By contrast, statistical claims are evidence for, rather than embody, causal claims. Furthermore, we accept statistical evidence as supporting causal claims only when there is independent reason to suppose that the phenomena involved are relevant to one another.

For example, I recall reading somewhere that for many years, the membership rolls of a baker's union in New York City precisely paralleled the births and deaths in a town in India. Whether this actually happened is not important here; my point is, it could well happen, and if it did, no reasonable person would take it as anything more than a striking (and somewhat humorous) coincidence.

The transitivity of inferences based on generalizations again distinguishes them from statistical claims. Presumability can be inherited through truth-functional inferences; but statistical relationships are far more complex, and statistical inferences follow utterly different rules.

For instance, consider the result of conjoining two statistical claims S and S'. Say the probability of S is x, and that of S' is y. Now what is the probability of S & S'? Well, let's look at some examples.

Suppose the subject is coin tossing. Say S says "Toss 1 will be heads," and S' says "Toss 2 will be tails." Then x = y = .5, and the probability of "Toss 1 will be heads and toss 2 will be tails" we know to be .25, or xy. But is this always the case? Clearly not. Let S be as before, and let S' be "Toss 1 will be tails." Now the probability of S & S' is 0. If S is the same as S', then the probability of the conjunction is the same as the probability of S.

Furthermore, statistical analyses tend to be applied to two fundamentally different sorts of situation. In the first kind, the various events are ex hypothesi independent of one another. We assume that the result of toss 1 does not affect the result of toss 2. In the second, a causal relationship is being sought or presumed. At this point, probabilities become inextricably linked to the theoretical context, and in some sense take on a different meaning. Given one set of results R, the probability of R will differ depending on the hypothesis relative to which it is computed. More importantly, what changes tends to be not the probabilities of individual occurrences, but precisely the probabilities of cooccurrences: that is, the probability of conjunctions changes, without that of the conjuncts changing. So no general rule captures the way the probability of a conjunction relates to the probability of the conjuncts.

3. Examples of default assumptions.

Suppose we are designing a "travel agent" system. The classical example of a default rule in this context is the assumption that, all else being equal, all trips originate wherever the customer currently is. This seems reasonable enough, but the system need hardly assume it, since it can request that information with no great loss of convenience.

But consider the following "rule": within the departure time limits the customer supplies, more direct connections are to be preferred over less direct ones. If someone says, "I'd like a round trip to New York," for the system then to ask, "Where are you leaving from?" seems reasonable; for it to ask, "Would you rather get there in one hour or nineteen and a half?" does not.

Furthermore, imagine a system which mindlessly produced every set of connections from Buffalo to New York — direct, via Albany, via Houston, via Seattle, via London, via Buenos Aires.... While the list might not go on forever, it will surely go on long enough to prove inconvenient. Some presumption must be made to order the alternatives so that reasonable ones get listed early.

Yet we cannot simply add a universal rule that direct routes are to be preferred over indirect ones, because it isn't always true. For example, some people refuse to use certain airlines or airports under any circumstances. Others will want to stop over for a few hours in some intermediate city.

There is also a more general problem. All else being equal, the cheaper of two routes is usually preferred over the more expensive. While the more direct route is usually also the cheaper, it is not always so. One can currently fly from Buffalo direct to Albany, which is shorter and more direct than changing flights in New York City. But it turns out that flying via New York is cheaper. Whether the customer wants to fly direct or via New York City will now depend on which is more important to the customer, convenient time scheduling or low price.

Hence the system cannot presume absolutely either that the more direct route is preferred, or that the cheaper route is. A guarded answer which presumes either, but with explicit reservations, will prove more useful than either a flat presumption which cannot be overruled (a universal) or a failure to make any presumption at all.

Other examples abound. If a customer asks to travel from New York to Cincinnati via Athens, we want the system to recognize that the customer probably means Athens, Ohio, and not Athens, Greece, or even Athens, Georgia. At the same time, this assumption should somehow be reflected in the system's response, lest travellers who mean to go to Athens, Georgia learn of Athens, Ohio by finding themselves there.

4. Problems defaults raise.

Perhaps the most common kind of default takes the form, "In the absence evidence that ~p, you may infer p" [7,8]. When the system is asked "p?" and finds the default rule, it attempts to derive ~p. If it fails to do so, it returns p as the answer. Hence systems augmented by this kind of rule can take advantage of generalizations of the kind above. So far, so good.

But this procedure only looks reasonable so long as we deal with questions like "Can Roger the bird fly?" Then, saying "Of course, he's a bird," seems unobjectionable — but only because nothing depends on the answer. Notice that if we don't care what the answers to our questions are, there is little reason to implement defaults. After all, if we don't care, we can as well say "I don't know" as either yes or no.

But suppose that we do care what answer we get. For instance, consider a medical diagnostic and treatment-recommending system. Suppose that for a particular set of symptoms, treatment x is generally very beneficial, but that in the exceptional cases treatment x invariably kills. Now if A has the symptoms in question, surely we do not want to recommend treatment x solely on the grounds that we don't yet know that A is exceptional. On the other hand, if the symptoms in question can themselves prove fatal, nor do we want to say we don't know anything about what to do for A.

In this kind of case, we would like the system to say something like, "Treatment x usually helps," or "Presumably treatment x helps." Even better would be an answer which directly tells the user what the counterindications are; but at the very least, a responsible system should warn the user that the information results from a presumption, and not an inference. Once the system has issued the warning, the user can then pursue it in further questions.

A further difficulty with defaults lies in deciding what it means for them to be true or false. Clearly "If Roger is a bird, then presumably Roger can fly" can be true even if Roger is a bird, but Roger can not fly. Indeed, "Presumably Roger can fly" can be true, even though "Roger can fly" is false. That is the whole point of saying "presumably": it protects the speaker from saying something false when the facts go the "wrong" way. That is what it means to give a guarded response.

Hence the truth value of defaults cannot be a simple function of the truth values of their component propositions: default operators are not truth functional. Furthermore, defaults make sense because they reflect causal (and hence non-logical) connections among their constituents. The missing information guarantees that their content cannot be a simple function of the contents of the components. But then we should not expect to be able to give a purely logical account of defaults [4].

5. Problems with two proposed solutions.

Several approaches to defaults have been suggested. Some researchers treat defaults as modalized [6,7,8]. Several problems with this approach have been pointed out already (see e.g. [3]). In addition, this approach interprets "In general, birds fly" as something like "If x is a bird and it is compatible with what we know that x flies, then x flies" [7]. But this is only true if every single bird without exception which we do not know to be flightless does in fact fly. That is, if McDermott's version of the generalization is true, it can never be the case that some bird does not fly and we can not prove that it doesn't. But this is surely not what the generalization means.

The fuzzy logic approach [1,5,9,10] uses a continuum of truth values in the closed range <0,1> instead of simply "true" and "false". Several questions immediately arise. First, every "assertion" in the data base must have an associated truth value; where are we to get these from? Second, how are the truth values of propositions related to those of their components, and how are the truth values of conclusions related to those of the premises of the demonstration in question? Preliminary results [1] boil down to the unsurprising claim that the conclusions are no better than the premises, but also on the whole no worse (where "better" is interpreted as numerical "greater than"). It is significant that this is already non-trivial to establish. Third, how do we deal with the apparent result that different demonstrations of the same proposition "establish" different truth values?

But the largest problem, in my opinion, lies in the irresistible temptation to view these fuzzy truth values as probabilities. This tendency is encouraged by the need to assign what, in context, look much like Bayesian prior probabilities to the

propositions in the data base. Some kind of Bayesian analysis may prove useful in A.I. systems; but there is no "cut-rate" way of doing it. Neither fuzzy logic nor default reasoning adequately analyzes probability. Under the circumstances, it seems best to avoid a system which misleads to this extent.

6. Conclusion.

We would like some way to deal with the "funny" truth status of default rules and of conclusions drawn on the basis of default assumptions; but neither modality nor fuzzy truth values seems to capture the desired effect. Furthermore, there seems good reason to suppose that no purely logical analysis could.

But this does not rule out the possibility that logical restrictions on defaults and their consequences can be found and described, on the basis of which a system of inferences allowing default reasoning can be developed. We are currently developing a semantics for default reasoning which treats defaults as propositional operators and which we hope will provide such a basis. Once this has been done, we can hope to deal with defaults in a reasonable and useful way.

Hence an A.I. system which deals with defaults successfully must also have at least two properties which existing proposals lack. First, it must delineate the logical restrictions on defaults and their consequences without ruling out the existence of genuine exceptions, i.e., recognizing that default reasoning sometimes gives the wrong answer. In doing so, it should be careful to distinguish default generalizations both from genuine universals and from statistical generalizations. And second, when the system gives answers which are based on default reasoning, it should admit this weakness by issuing warnings with them. For without such warnings, default reasoning by any scheme is not only unsound: it is also unsafe.

8. Acknowledgments.

I would like to thank Stuart Shapiro and the members of the SNePS Research Group at SUNY/Buffalo for their many helpful comments and suggestions.

8. References.

[1] Aronson, A.R., Jacobs, B.E., and Minker, J. A note on fuzzy deduction. JACM v. 27 (1980) 599-603.

[2] Brachman, R.J. "I lied about the trees" or defaults and definitions in knowledge representation. Draft (1982).

[3] Davis, M. The mathematics of non-monotonic reasoning. A.I. v. 13 (1980) 73-80.

[4] Israel, D.J. What's wrong with non-monotonic logic? Proc. First Annual National Conference on Artificial Intelligence, American Association for Artificial Intelligence (1980) 99-101.

[5] Lee, R.C.T. Fuzzy logic and the resolution principle. JACM v. 19 (1972) 109-119.

[6] McDermott, D.V. and Doyle, J. Non-monotonic logic I. A.I. v. 13 (1980) 41-72.

[7] McDermott, D. Non-monotonic logic II. JACM v. 29 (1982) 33-57.

[8] Reiter, R. A logic for default reasoning. A.I. v. 13 (1980) 81-132.

[9] Zadeh, L.A. Fuzzy sets. Inf. Control v. 8 (1965) 338-353.

[10] Zadeh, L.A. Fuzzy algorithms. Inf. Control v. 12 (1968) 92-102.

PRAGMATIC FACTORS IN PRONOUN REFERENCE ASSIGNMENT

Valerie C. Abbott and John B. Black
Cognitive Science Program
Yale University, New Haven, CT 06520

Identifying factors that influence pronoun reference assignment is a challenge to anyone attempting to characterize the process of language understanding. Because a pronoun itself carries only a small part of the meaning that the understander is expected to assign to it, he or she must use contextual information to assign the pronoun an unambiguous referent. Characterizing aspects of the context which are used for this purpose is an active area of psychological research.

Many recent studies have considered the role of syntactic context, that is, the effect of structural constraints on pronoun reference in a fragment of text, typically a sentence, without recourse to constraints which might be found in the meaning of the text (Langacker, 1969; Sheldon, 1974). Shwartz (1981) has found evidence for the use of syntactic information in the resolution of anaphoric pronouns in single sentences. However, strategies based only on syntax are not sufficient to determine unambiguously the referent of all pronouns. Consequently, investigators have examined the role of semantic factors within sentences in directing the assignment of referents (Caramazza, Grober, Garvey, & Yates, 1977; Caramazza and Gupta, 1979; Ehrlich, 1980).

The studies reported here will focus on the use of pragmatic constraints in resolving anaphoric pronouns. Hirst and Brill (1980) have found that these constraints influence the time needed to assign a referent even when that referent can be unambiguously determined by syntactic rules alone. This result indicates that pragmatic context can be expected to play a significant role in reference assignment. However, the text fragments used in their study were only two sentences long, and the nature of the pragmatic considerations involved were not specified. It remains to be determined whether there are identifiable cues in longer texts which influence reference assignment of anaphoric pronouns. We will be concerned with characterizing two major sources of contextual information in paragraph-length texts, and evaluating their influence on pronominal reference assignment.

First, the presence of a clear main character may be expected to play a role in reference assignment. Black, Turner, and Bower (1979) have shown that the point of view provided by a main character has an observable effect on story understanding. In the extreme case, there may be only one character in a story. When there is more than one character, it is still likely that the main character is given primary consideration for reference assignment. This was investigated in the current experiment.

Second, Schank and Abelson (1977) have suggested that the goals and social roles of characters in stories may contribute to reference assignment. If an act is appropriate to a particular goal or role and the agent of the act is specified by a pronoun, it is likely that the pronoun will be disambiguated to the character who has the appropriate goal or role.

Since the goals the characters in a story are pursuing, the roles they are filling, and the identity of the main character can be experimentally manipulated, we can test whether these contextual cues influence pronoun reference assignment. In the experiments reported below we first test whether subjects are sensitive to these cues alone and in combination in a task requiring explicit pronoun reference assignment. Second, in a task in which reading times for lines of text containing pronouns were measured, it was determined whether these sources of pragmatic constraint influenced the difficulty of reference assignment as measured by reading time.

Experiment I: Explicit Assignment

Four simple two-character stories were written. Each story contained an anaphoric pronoun in the final sentence. Either character could be made the main character of the story, or each character might be weighted equally. Additionally, each character was given a role or a goal in the story. Preceding the clause in which the critical pronoun appeared was a phrase containing an action appropriate to the role or goal of one character or other, or an action which was equally likely to have been performed by either of the characters. For instance, in "Brushing off a table, she smiled at her friend." the action preceding the pronoun is consistent with the role of a waitress. Note that in sentences of this sort, the subject of the main clause is interpreted as the agent of the action in the preceding phrase.

Combination of these cues yields five presentation conditions.

- The main character and goal or role cue are both present and indicate the same referent.
- The main character and goal or role cue are both present and indicate conflicting referents.
- Only the main character cue is present.
- Only the goal or role cue is present.
- Neither cue is present.

Each subject was presented with two stories of the type described above, one in each of two conditions. Following each story on a separate page was a multiple choice question requiring identification of the character to whom the anaphoric pronoun referred.

The results of this experiment are summarized in Figure 1 below. When main character and role or goal cues led to assigning the same character as referent, pronoun reference was determined in accord with both by 84% of the subjects, a significant difference from chance ($\chi^2 = 10.72$, $p < .01$). This shows that main character and role and goal manipulations are powerful enough to influence pronoun assignment when used together. In the case in which neither main character nor the phrase preceding the pronoun provided a cue concerning pronoun reference, subjects chose both characters almost equally often as the referent of the pronoun, 46% of the subjects choosing one and 54% choosing the other ($\chi^2 = 0.12$, ns). When the phrase preceding the pronoun was neutral with respect to the roles or goals of both characters in the stories, but there was a main character,

this character was adopted as the referent of the pronoun by 82% ($\chi^2 = 9.02$, $p < .01$) of the subjects. This is essentially the same level of performance as was observed with both sources of information avalable to the subjects. However, when both characters were given equal weighting in the story, but the phrase preceding the pronoun was appropriate to the role or goal of one character, the referents chosen were consistent with this character for only 62% ($\chi^2 = 1.07$, ns) of the subjects. This pattern of results seems to indicate that subjects are not making extensive use of information about the relationship between an action the agent of which is specified by a pronoun, and the known goals and roles of characters, in assigning the pronoun a referent.

However, this interpretation is complicated by the results of the condition in which subjects had to make a choice between an assignment to the main character of the passage, or to another character with the role or goal appropriate to the action preceding the pronoun. In this situation, subjects chose the assignment which agreed with the main character 38% of the time, and chose the assignment which agreed with the role or goal context 62% of the time. Although this result is not significantly different from chance ($\chi^2 = 1.07$, ns), a difference in the opposite direction would be expected if only main character cues were influencing the choice. This result indicates that although a character's goal or role is not always sufficient to influence pronoun assignment alone, it is important when seen in combination with other information. The difference between the choice of

	CHOICE	
CONDITION	CONSISTENT WITH CUE(S)	INCONSISTENT WITH CUE(S)
BOTH CUES (CONSISTENT)	84	16
MAIN CHAR CUE ONLY	82	18
GOAL OR ROLE CUE ONLY	62	38
BOTH CUES[a] (CONFLICT)	62	38
NEITHER[b] CUE	54	46

[a] consistent = consistent with goal or role cue
[b] consistency arbitrarily determined

Figure 1: Subjects' choice of pronoun referents in percent.

referent in this condition and in the condition in which main character identity is the only cue available is significant ($\chi^2 = 15.47$, $p < .001$). The utility of main character information thus seems to be dependent on the absence of conflicting information.

The results of the this experiment indicate that the extent to which subjects chose one referent or the other was governed by the contextual cues manipulated. The main character of the story was most effective in influencing reference assignment, with consistency of the pronoun's context with the goal or role of a character effective in nullifying this main character effect.

It is conceivable that in this experiment asking explicitly about the referent of a pronoun altered subjects' responses. Thus, it seemed desirable to obtain another measure of the difficulty of assigning referents to anaphoric pronouns in the same texts.

In the following experiment reading times for the sentences of these texts containing anaphoric pronouns were measured. It was expected that reading times would be fastest for pronouns in the condition in which there was a main character, and the phrase preceding the pronoun was appropriate to the role or goal of that character. Reading times should increase as it becomes increasingly difficult to assign a referent unambiguously to a pronoun.

Experiment II: Reading Time

Materials were the four stories used above and six additional stories of the same type written for this study. Each story could appear in any of the five conditions discussed above. The penultimate line of the story contained the action which was consistent with the role or goal of one character or the other, or with either. The final line of each story was constant over conditions and contained an anaphoric pronoun.

Each subject read the 10 stories, two in each of the five conditions. They were instructed to read the stories for comprehension. Each story was presented one line at a time on a computer terminal, subjects pressing the "Return" key when they had finished reading each line. Reading times for the final line of the story were compared between conditions.

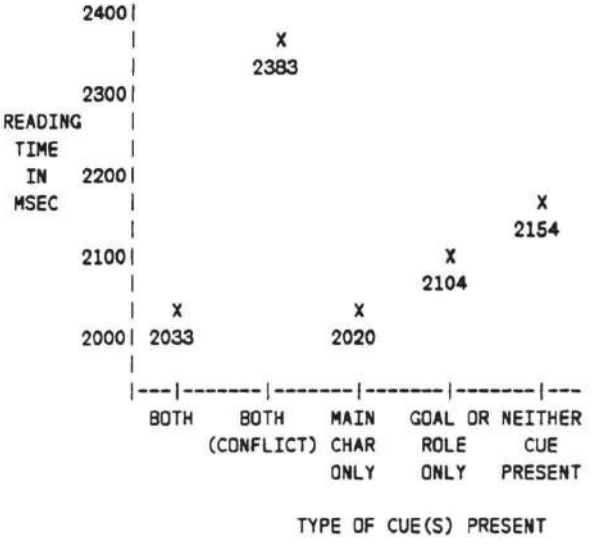


Figure 2: Reading times for a clause containing an anaphoric pronoun

The results for the five conditions are presented in Figure 2. The reading time data is quite consistent the data seen in Experiment I above. A comparison between the condition in which both cues are present and lead to the same choice of referent and that in which both cues are present but lead to conflicting choices shows faster reading times in the former condition ($F = 4.895$ $p = 0.033$). Having only one cue in the form of a main character leads to almost identical reading times as having both cues and results in significantly faster

reading times than the confusing condition ($F = 9.487$ $p = 0.005$). However, although there is a trend, having only the cue of consistency with the goal or role of a character does not lead to significantly faster reading times than the confusing condition ($F = 3.022$ $p = 0.089$). The condition in which neither main character nor consistency with a goal provided a cue as to the reference of the pronoun is a puzzle. Although it is not significantly faster than the confusing condition ($F = 1.325$ $p = 0.258$), it is also not significantly slower than the condition in which both cues are available ($F = 0.527$ $p = 0.480$), the condition in which only the main character is available ($F = 0.608$ $p = 0.448$), or the condition in which only consistency with a goal or role is available as a cue ($F = 0.146$ $p = 0.705$). One possible explanation is that subjects are fairly quick to realize that they have no information with which to make a decision, and proceed in hopes of obtaining the information they need in the remainder of the text. In other words, in the confusing condition, enough information is available, so an attempt is made to find the referent. This proves difficult, leading to increased reading times for such sentences. In the absence of relevant information, the attempt at resolution is deferred.

The results of these two experiments show the influence on pronoun reference assignment of manipulation of pragmatic aspects of the text in which they appear. The main character of the text, in the absence of disconfirming evidence, is quickly and reliably assigned as the reference of these pronouns. They also point out that the influence of some possible pragmatic cues cannot be characterized simply. For example, if the action of an agent represented in the text by a pronoun is consistent with the role or goal of a character, this is not sufficient to lead reliably to assignment of that character to the pronoun. However, the influence of this cue is substantial enough to lead to confusion if there is other evidence indicating another character as the referent. Additionally, it cannot be assumed that the less information available for pronoun reference assignment, the longer it will take subjects to read the sentence in which it appears. From the results of experiment II we can see that subjects proceed rather quickly when they have no information on which to base their choice.

Acknowledgments

We are grateful to Rowell Huesmann for sponsoring this paper, and to Wendy Lehnert and Larry Birnbaum for helpful discussions regarding the research reported here. This research was supported by grants from the Systems Development Foundation and the Sloan Foundation.

References

Black, J. B., Turner, T. J., & Bower, G. H. Point of view in narrative comprehension, memory, and production. *Journal of Verbal Learning and Verbal Behavior*, 1979, **18**, 187-198.

Caramazza, A. & Gupta, S. The roles of topicalization, parallel function and verb semantics in the interpretation of pronouns. *Linguistics*, 1979, **17**, 497-518.

Caramazza, A., Grober, E., Garvey, C. & Yates, J. Comprehension of anaphoric pronouns. *Journal of Verbal Learning and Verbal Behavior*, 1977, **16**, 601-609.

Ehrlich, K. Comprehension of pronouns. *Quarterly Journal of Experimental Psychology*, 1980, **32**, 247-256.

Hirst, W., & Brill, G. A. Contextual aspects of pronoun assignment. *Journal of Verbal Learning and Verbal Behavior*, 1980, **19**, 168-175.

Langacker, R. On pronominalization and the chain of command. In D. Reibel, S. Schane (Ed.), *Modern Studies in English*, Englewood Cliffs: Prentice-Hall, 1969.

Schank, R.C., and Abelson, R.P. *Scripts, Plans, Goals, and Understanding.* Hillsdale, NJ: Lawrence Erlbaum Associates, 1977.

Sheldon, A. The role of parallel function in the acquisition of relative clauses in English. *Journal of Verbal Learning and Verbal Behavior*, 1974, **13**, 272-281.

Shwartz, S. *The search for pronominal referents.* Technical Report 10, Cognitive Science Program, Yale University, 1981.

Topic and Comment in Spoken Sentence Comprehension

Hans Brunner
University of Indiana

Chomsky (1965) has defined the Topic of a sentence as "the leftmost NP immediately dominated by S in the surface structure" and Comment as, quite simply, "the rest of the string". Others have either defined or used these two concepts to denote, among other things, the distinction between (1) "new" information and information that has already been conveyed (e.g., Clark & Haviland, 1977), (2) the notions of "psychological subject" and "psychological predicate" (e.g., Hornby, 1972), or (3) the "current" vs. "presupposed" information of a sentence (e.g., Halliday, 1967). Differing interpretations abound and, in the words of deBeaugrande (1980), "it has remained unclear precisely what phenomenon we are dealing with".

The purpose of this research was to investigate the roles of "topic" and "comment" in different semantic and syntactic contexts. To do this we used the gating paradigm, a procedure in which spoken sentences are repeatedly presented to subjects, the amount of spectral information from each constituent word being gradually increased with each successive repetition. In the first presentation of each sentence, the spectral gate size (i.e., duration from the onset) of each word was only 50 msecs. The remainder of each word was replaced with envelope-shaped noise, a procedure which eliminates the spectral information while preserving prosodic fluctuations in the intensity of the speech. Each target sentence was repeated 10 times, the gate sizes being increased in 50 msec increments across repetitions. Subjects were instructed to simply write down whatever they could understand after each presentation of the sentence. The dependent measure of interest was the amount of spectral information (i.e., the "gate size") necessary for comprehension of each word in the sentence.

This technique was applied to the current issue by transforming the syntax of simple, declarative sentences so as to vary the topicalization of subject and object nouns from one sentence version to the next. Our syntactic transformations, taken from a study by Hornby (1972) are shown below:

(1)The farmer plowed the field.
(2)The field was plowed by the farmer.
(3)It was the farmer who plowed the field.
(4)It was the field that the farmer plowed.
(5)The one who plowed the field was the farmer.
(6)What the farmer plowed was the field.

Hornby (1972) showed that agent of a sentence serves as the topic when presented in syntactic structures with a cleft object (sentence 4), pseudocleft object (6) or in active sentences (1) and as the comment when presented either in passive sentences (2) or in sentences with cleft (3) or pseudocleft (5) agents. The object takes on a complementary role, being part of the comment where the agent is topicalized, and vice versa. The topic of each of of each syntactic form has been underlined, above, according to this criterion. In this study we capitalized on this exchange of roles so that, when comparing the overall effects of topic vs. comment status, we would be comparing each word against different tokens of itself.

Armchair theorists have been asserting for some time now that the topic of a sentence (1) receives less intonational stress (i.e., lower amplitude and F0 and a shorter duration) in production and (2) is somehow prerequisite for correct interpretation of the comment. If this is true, then comprehension of any given word should require less spectral information when it functions as topic than when it is stretched out in time as part of the comment on what has been topicalized. Moreover, if the functionalist approach is correct, then there should be a well-ordered interaction between topicalization and syntax, with agents requiring a smaller minimal gate size in active sentences and sentences with cleft and pseudocleft objects, where they are topicalized, than in the remaining three syntactic forms, where they are part of the comment. And once again, the converse should obtain for the object of each sentence.

Neither of these predictions was supported by the results: The amount of spectral information necessary for word recognition did not decrease as a function of increasing topicalization. Moreover, there was a significant main effect of syntax ($F(5,270)=26.18$), resulting from an increase in the amount of spectral information necessary for word recognition as the syntax of sentences became more complex.

These results should not be construed as evidence against the functionalist approach to sentence comprehension. Our sentences were presented out of context, in the absence of any larger text or dialogue framework. Thus, it is doubtful that the topicalized words in these stimuli really represented anything akin to "given" or "presupposed" information for the subjects. Nonetheless, these results do serve to constrain some of the notions that have been advanced about the nature of topic and comment in the processing and structure of language. They make if quite clear that "topic" and "comment" are textual, rather than syntactic or structuralist concepts. Thus, any effort to define these constructs without reference to intersentential relations simply misses the purpose of topicalization in real-time processing. However, the results also demonstrate that it is important not to lose sight of syntactic effects in text processing. The syntactic constraints of these sentences did much more than just control the focus of attention; they had profound, top-down effects on the overall speed of identification as well.

The current results are only the first in a series of experiments on this issue. In this talk, I will also discuss the effects of similar manipulations on materials presented in various textual frameworks.

ON-LINE PROCESSING OF PRAGMATIC INFERENCES

Colleen M. Seifert, Scott P. Robertson,
and John B. Black

Cognitive Science Program
Yale University

Cognitive science researchers have proposed a wide variety of inferences and inference mechanisms that may be used in comprehending stories. Inferences are concepts, or links between concepts, which are not explicitly stated in a text but which are present in the final memory representation. Many previous psychological experiments on inferences have been unable to distinguish between inferences that are generated during comprehension (on-line) and those that are constructed later (for example, during summarization or question answering). The experiments presented here contrast four types of pragmatic inferences to determine whether they are usually generated on-line.

Pragmatic inferences are a class of inferences that result from the application of world knowledge to information in a text. Knowledge structures typically employed in the production of pragmatic inferences (especially for narratives) are goal structures, planning mechanisms, and scripts (Schank & Abelson, 1977; Wilensky, 1978). A number of psychological experiments have demonstrated the use of individual schematic structures in producing pragmatic inferences (e.g. Bower, Black & Turner, 1979; Graesser, Gordon, & Sawyer, 1979; Smith & Collins, 1981), but have not shown the on-line operation of a combination of knowledge structures involved in pragmatic inference generation. In the two studies discussed here, we will present evidence that 1) knowledge-based inferences about goals, plans, and actions are made during reading and 2) inferences about consequent or associated states of the world are not made during reading. We will also give indirect evidence for on-line forward inferencing of plans from goals.

Knowledge of goals and plans organizes otherwise disconnected text elements, and thus it is important that they be inferred early in the comprehension process (Owens, Bower, & Black, 1979; Smith & Collins, 1981). Lower level inference types, like story actions, are used to fill in information specified by already active schemata (Bower, Black & Turner, 1979). State information, however, while potentially inferable, is not predicted to be generated as part of the comprehension process. There is considerable evidence that physical states that are antecedents or consequences of actions are not a central part of narrative representations (Black, 1980; Graesser, 1981; Kemper, 1982; Lehnert, Robertson, & Black, in press; Robertson, Lehnert, & Black, 1981). For example, when someone sits down in a restaurant, information about the position of tables and chairs is not typically accessed.

To test for on-line inferences of the specified types, we measured subjects' reading times for target sentences which *required* a pragmatic inference for coherence. In the first experiment we wrote sixteen short (17 line) stories each containing a goal, a plan for achieving that goal, a set of connected actions, and associated states. Eight of the stories were *script based* (e.g. going to a restaurant, going to the movies), the other eight were *plan based* (e.g. robbing a store, getting directions). Each story included *inference-statements* which explicitly described the goal, the plan, an act, and a state. Following each of these statements was an eight-syllable *target-statement* which required the preceding information to be inferred if it was not already present in memory. For example, sentence 2 when read alone requires that the goal stated in sentence 1 be inferred; sentence 3 requires an inference of the plan stated in sentence 2; sentence 6 may require an action inference (sentence 4) but not a state inference (sentence 5). (Our stories were not as compact as this example suggests.)

1. John was hungry.
2. John hurried to a restaurant.
3. John ordered the special dinner.
4. The waitress brought the food.
5. John had silverware.
6. John ate his meal in a hurry.

Target-statements (e.g. sentence 3) were presented with their associated inference-statements (e.g. sentence 2) either present or absent. Each subject received stories with goal, plan, act, and state inference-statements absent, but within any one story a subject had only one high level inference type (goal or plan) and one low level inference type (act or state) left out. Subjects read the stories one line at a time from a CRT screen and their reading times for the target-statements were recorded. It was assumed that inference generation would be evident in increased reading times for the target-statements in the inference-statement absent conditions. After the reading task and a short intervening task, the subjects were given a recognition test (1-7 scale) which included the inference-statements. High recognition ratings for absent inference-statements indicates the presence of the inferences in the final story representations.

Table 1 shows the mean reading times for target-statements and mean recognition ratings for inference-statements of the different types in the present and absent conditions. The analysis of reading times showed that goal and action targets took longer to read when their inference-statements were absent, but this was not the case for plans or states. Recognition results showed a specific interaction in which states were not falsely recognized when they are left out of the stories while the other types of inference-statements were. The reading time data and recognition data together support the view that goals and actions are inferred on-line whereas states are not. Plans proved problematic and were investigated further in a second experiment.

	Target RT		Inference Recognition	
Type of Inference	Inference Absent	Present	Inference Absent	Present
Goal	* 1.660	1.559	4.89	5.81
Plan	1.626	1.601	4.95	6.09
State	1.538	1.487	* 3.62	5.82
Act	* 1.595	1.448	4.75	6.06

Table 1. Mean reading times (sec.) and recognition ratings for the different inference types.

Though the reading time difference for plans was not significant in the first experiment, the high recognition rating for absent plans suggests that they were inferred at some point. A closer look at the materials revealed a possible explanation: knowledge of the goals in stories where the plan inference-statements were left out may have allowed subjects to infer the plans before their target-statements were read. For example, knowledge of the goal "John was hungry," may lead to a prototypical plan expectation, i.e. "going to a restaurant." If a prototypical plan is inferred when a goal in read, the presence or absence of the plan inference-statement would not have made any difference.

In a second experiment, prototypical plans in our materials were changed to less typical plans to minimize forward inferencing from the goals. In addition, some story titles were changed to decrease the chances of inferring a goal prior to reading the goal target-statements. Also, action inferences were not included in the second experiment since this effect had already been clearly demonstrated.

The results of the modified experiment are shown in Table 2. The reading time differences for goal and plan inferences increased and plans now became significant. We again failed to find evidence for on-line state inferences. The recognition data remained consistent with these results, showing a high false alarm rate for goals and plans, but not for states.

	Target RT		Inference Recognition	
Type of Inference	Inference Absent	Present	Inference Absent	Present
Goal	* 1.764	1.613	5.28	5.94
Plan	* 1.720	1.626	5.69	6.27
State	1.536	1.490	* 3.97	5.56

Table 2. Mean reading times (sec.) and recognition ratings for the different inference types.

Taken together, these experiments support the view that some pragmatic inferences, specifically goals, plans, and actions, are made during reading while others, specifically low level states, are not. It is especially important to note that high level inferences about goals and plans are made on-line. This result is congruent with models of language comprehension that incorporate strong *top down* uses of pragmatic knowledge during understanding. Active goal and plan schemata serve *during reading* to organize otherwise disconnected concepts in the text. We also obtained indirect evidence for on-line forward inferencing of prototypical plans from goals since we were only able to demonstrate that plans were inferred in a backward manner from plan inference-statements when they were non-prototypical of an active goal.

In terms of low level actions, the results support the view that script and plan completion inferences (remember that we had both script-based and plan-based stories) found in the representation after reading are not reconstructed at test time, but are built during reading. On the other hand, there was no evidence that inferences about states of the world occur during comprehension, even though we know that they are available after comprehension and even during comprehension in response to question probes (Graesser, 1981). Of course, some types of states may be very important and reliably inferred in some texts (Owens, Bower, & Black, 1979); however, the theoretical claim is that low level states *in general* are inferred on-line less often than the other types of inferences studied.

This "fine tuning" of data about the types of inferences made on-line provides important constraints on inference models. Since pragmatic inferences are probable rather than necessary, and since there is so much inferential material available at any given time from world knowledge, direct measures are needed to tell when inferences are made and which types are made. Although most models of language comprehension include an inferencing component, it is important to examine how different classes of knowledge are differentially utilized by the comprehension process.

Acknowledgments

We are grateful to Arthur Graesser for sponsorship and to Brian Reiser for comments on this paper. This research was supported by grants from the Sloan Foundation and Systems Development Foundation.

References

Black, J. B. Memory for state and action information in narratives. Twenty first Annual Meeting of the Psychonomic Society, St. Louis, Missouri, 1980.

Bower, G. H., Black, J. B., & Turner, T. J. Scripts in memory for text. *Cognitive Psychology*, 1979, **11**, 177-220.

Graesser, A. C. *Prose Comprehension Beyond the Word.* New York: Springer-Verlag New York, 1981.

Graesser, A. C., Gordon, S. E., & Sawyer, J. D. Memory for typical and atypical actions in scripted activities: Test of a script pointer + tag hypothesis. *Journal of Verbal Learning and Verbal Behavior*, 1979, **18**, 319-332.

Kemper, S. Filling in the missing links. *Journal of Verbal Learning and Verbal Behavior*, 1982, **21**, 99-107.

Lehnert, W. G., Robertson, S. P., & Black, J. B. Memory interactions during question answering. In H. Mandel, N. L. Stein, & T. Trabasso (Eds.) *Learning and comprehension of text.* Hillsdale, N.J.: Ablex, in press.

Owens, J., Bower, G. H., & Black, J. B. The "soap opera" effect in story recall. *Memory and Cognition*, 1979, **7**, 185-191.

Robertson, S. P., Lehnert, W. G., & Black, J. B. *Alterations in memory for text by leading questions.* Paper presented at the 1982 meeting of the American Educational Research Association, New York.

Schank, R. C., & Abelson, R. P. *Scripts, plans, goals, and understanding.* Hillsdale, N.J.: Erlbaum, 1977.

Wilensky, R. Why John married Mary: Understanding stories involving recurring goals. *Cognitive Science*, 1978, **2**, 235-266.

Generation of Useful Problem Representations in a Semantically Rich Domain: The Example of Physics

Joan I. Heller and F. Reif
University of California, Berkeley

The initial representation of a problem can crucially determine whether the subsequent search for its solution is easy, difficult, or even impossible. However, the processes used to generate initial problem representations, particularly in semantically rich domains, have been studied less extensively than those used for search. Accordingly, the study reported in this paper has aimed to formulate and test a model specifying how human problem solvers can generate effective initial descriptions of problems in a realistically complex scientific domain.

The preceding goal, which is prescriptive, is more general than one concerned with naturalistic studies of actual experts (Chi, Feltovich, & Glaser, 1981; Larkin, McDermott, Simon, & Simon, 1980). In particular, it focuses interest on procedures for generating good problem representations, without necessarily trying to simulate the behavior of experts and without making the assumption that experts behave optimally. From this general point of view, models of good problem description may thus be suggested by purely theoretical analyses as well as by observations of experts. (Indeed, protocol observations of experts reveal relatively little about the processes used to generate initial problem representations since these processes are usually carried out rapidly and almost automatically on the basis of much tacit knowledge.)

A prescriptive point of view, transcending naturalistic studies of expert performance, is also centrally important for attempts to improve human performance or for educational applications. Indeed, in instructional applications, students can not merely be taught to mimic expert performance which often relies heavily on the recognition of patterns acquired as a result of years of experience.

Our prescriptive interest has been specifically focused on human performance in generating effective problem descriptions. From a theoretical point of view, this emphasis allows us to presuppose complex human capabilities (such as natural-language understanding and pattern-recognition skills) while focusing attention on the more sophisticated cognitive skills needed to generate good problem representations. Furthermore, our interest has been in developing experimental approaches which (unlike some forms of computer simulation) allow direct validation of models of good human performance in problem solving tasks.

We chose to study the generation of problem descriptions in the particular domain of physics (especially within the subfield of mechanics) because this is a realistically complex domain representative of other quantitative sciences. On the other hand, this domain is sufficiently simple and well-defined that the generation of problem descriptions can be specified and studied in some detail.

Model of Problem Description

Our aim was to formulate a theoretical model specifying how a human problem solver can generate, for any problem in a particular scientific domain, a useful initial problem description facilitating the subsequent solution of the problem. This model decomposes the description process into two successive stages. The first stage uses mostly domain-independent knowledge to generate a problem description which summarizes and organizes relevant information about the specified situation and problem goal, introduces convenient symbolism, etc. Since the generation of this basic description is relatively straightforward, we shall not discuss it further here.

The next stage of the description procedure is more complex and involves the generation of a "theoretical description" which deliberately redescribes the problem in terms of special concepts provided by the knowledge base for the relevant domain. All the principles in the knowledge base, which are expressed in terms of these special concepts, become thus readily accessible to facilitate the subsequent solution of the problem.

The generation of the theoretical problem description is based on the following considerations. The knowledge base about any domain contains declarative knowledge specifying the particular entities of interest in this domain, the special concepts useful for describing these entities, and principles specifying relationships between these concepts. For example, in the scientific domain of mechanics, the entities of interest are particles or more complex systems consisting of such particles. The special descriptive concepts are special concepts used to describe motion (e.g., "position", "velocity", "acceleration") and special concepts used to describe the interaction between particles (e.g., "force", "potential energy",...). The principles specifying relations between these concepts are "interaction laws" (which specify how the force on one particle by another is related to the properties and positions of these particles) and "motion principles" (which specify how temporal changes of concepts describing motion are related to concepts describing interaction).

The preceding kinds of declarative knowledge in the knowledge base about a particular domain provide the basis for explicit "description rules" that specify procedures for generating a theoretical description of any situation in this domain. In particular, these description rules specify what particular kinds of entities should be described, what special concepts should be used to describe them, what properties of these concepts should be incorporated in the description, and what checks should be made to ensure that the resulting description is consistent with the principles in the knowledge base.

For example, our model for generating a theoretical description in the particular scientific domain of mechanics contains explicit rules specifying that attention is to be focused on particles or certain systems of particles (e.g., strings, solid objects, ...). The motion of each such particle is then to be described by a diagram indicating available information about its position, its velocity, and its acceleration. Similarly, the interaction of each such particle is to be described by a diagram indicating available information about all forces on this particle by other particles (with an explicit algorithm specifying how all these forces are to be identified and enumerated). Finally, the resulting description is to be checked by assessing its consistency with known motion principles (e.g., by checking that the acceleration of any particle has the same direction as the total force on it).

The preceding description procedure, specified by the model, is expected to lead to initial problem descriptions with the following properties:

(1) The resulting descriptions should be considerably more explicit than those commonly generated by actual experts. (2) Strict adherence to the description procedure should avoid most of the errors commonly committed by novices (e.g., omitting forces or introducing non-existent extraneous forces). (3) The description procedure should lead to problem reformulations which are more readily interpretable (e.g., questions about slack strings or touching objects are automatically reinterpreted as questions about forces). (4) The resulting theoretical problem descriptions should substantially facilitate the subsequent solutions of these problems.

Experimental Methods and Results

Our experimental approach for testing a prescriptive theoretical model of human performance has used the following paradigm: Design carefully controlled experimental conditions to induce individual human subjects to act in accordance with the model; then observe whether the resulting performance is effective in the predicted ways.

To implement this paradigm, we have used "external-control experiments" of the following kind. We first design a program of step-by-step directions, and associated knowledge, whereby a human subject can be guided to act in accordance with the model (e.g., directions which implement the steps of the specified description procedure). These directions are problem-independent and at an appropriate level of detail to be reliably interpretable by the subject. In the actual experiments an individual human subject is then induced to carry out a task (e.g., the description and subsequent solution of a problem) by executing the sequentially presented directions of the program implementing the model. In this process the subject is asked to talk out loud about his or her thought processes. The resulting protocol, consisting of the subject's transcribed verbal statements and written work, can then be analyzed in detail.

Figure 1 shows the experimental results obtained by such external-control experiments designed to test the proposed model for generating effective initial descriptions of mechanics problems. Each subject worked on three problems. Figure 1 shows the performance of these subjects in generating good descriptions of motions and of forces, as well as subsequently generating solutions with correct equations and correct answers. The following are the main results obtained in these experiments: (1) The proposed model for generating initial problem descriptions is sufficient to lead subjects to generate explicit descriptions that are complete and entirely correct. In turn, these descriptions greatly facilitate the subsequent problem solutions which are then almost flawless. (2) Although subjects in these experiments possess a good knowledge of basic physics concepts and principles, a knowledge sufficient to implement the individual directions contained in the model, this knowledge is not sufficient to lead to good descriptions. These results are apparent from the much poorer performance of subjects in a comparison group working without external control of the model. (3) The main features of the model are, in fact, necessary for good performance. These results follow from experiments where subjects worked under external control of a modified model that omits certain features of the proposed model (e.g., that provides a direction to enumerate all forces, but does not provide more detailed directions specifying how to enumerate them). (4) The experimental data also verify certain detailed predictions of the model (e.g., the avoidance or occurrence of particular kinds of errors).

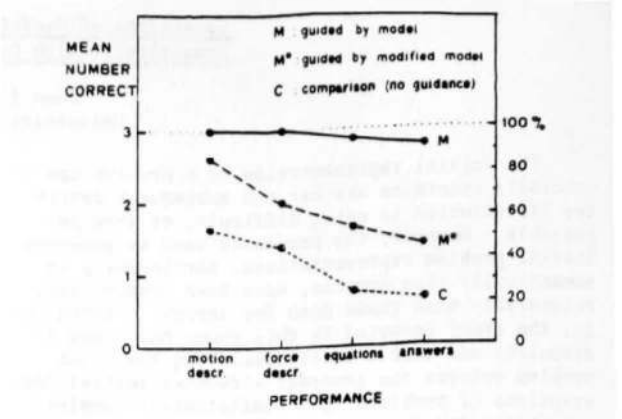


Figure 1. Results of external-control experiments.

Conclusions and Implications

The work briefly outlined in the preceding paragraphs leads to the following main conclusions.

The knowledge base for any scientifc domain implies guidelines specifying how to describe effectively any situation encountered in this domain. These guidelines can be expressed in terms of explicit rules prescribing how to generate a useful initial description of any problem in the domain.

Prescriptive models of effective human performance can be usefully tested by external-control experiments in which individual human subjects are deliberately induced to act in accordance with a model and the resulting performance is then observed in detail.

The work described in the preceding paragraphs was specifically undertaken to formulate a model for generating effective initial descriptions of problems in the particular domain of mechanics. External-control experiments show that this model, when implemented by human subjects, is very successful in leading to good initial problem descriptions that facilitate the subsequent solutions of these problems.

It should be noted that these experiments demonstrate the effectiveness of the specified description rules implemented by human subjects, but were not designed to teach description skills. (Indeed, such teaching would require that control knowledge, explicitly external in these experiments, be internalized by the subjects and made habitual.) However, such a well-validated model for generating effective initial problem descriptions can be used as a basis of explicit instructional methods to teach students effective problem-description skills and thereby enhance their problem-solving abilities.

REFERENCES

Chi, M.T.H., Feltovich, P.J., & Glaser, R., Categorization and representation of physics problems by experts and novices. Cognitive Science, 1981, 5, 121-152.

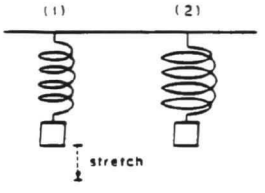
Larkin, J.H., McDermott, J., Simon, D.P., & Simon, H.A., Models of competence in solving physics problems. Cognitive Science, 1980, 4, 317-345.

ANALOGICAL REASONING PATTERNS IN EXPERT PROBLEM SOLVING

John Clement
Physics Department
University of Massachusetts
Amherst, Mass. 01003

Spontaneous analogies have been observed to play a significant role in the problem solutions of scientifically trained subjects [1,2]. In some cases analogies can even lead to the construction of a new mental model for understanding a problem domain. This paper describes a number of different analogical reasoning patterns that have been observed in thinking aloud protocols from expert problem solvers. The purpose of the present study is to identify, classify, and label the critical subprocesses involved in such analogical solutions. In this study each of ten subjects were given a number of problems, including the following one:

FIGURE 1



Spring Coils Problem

> A weight is hung on a spring. The original spring is replaced with a spring made of the same kind of wire, with the same number of coils, but with coils that are twice as wide in diameter. Will the spring stretch from its natural length, more, less, or the same amount under the same weight? (Assume the mass of the spring is negligible compared to the mass of the weight.) Why do you think so?

Subjects were advanced doctoral students and professors in technical fields who had reputations for being creative problem solvers. Seven of the ten subjects generated spontaneous analogies in solving this problem. A spontaneous analogy occurs when the subject, without being prompted, shifts to consider a situation B which differs in a significant way from the original problem situation A, and then tries to apply findings from B to A. In solutions by analogy the two contexts being compared are often perceptually different but they are still seen to be functionally or structurally similar in some way. For example, five subjects attempted to relate the problem to the analogy of a bending rod, as in the transcript excerpt below taken from video tape.

S1: (Draws bending rod in drawing G2-B of fig.2.) My intuition about that [the rod] is that if you.. doubled the length and hung some weight on it, that.. it, would bend considerably further... it would seem that that means that um, in the original problem, the spring in picture 2 [the wider spring] is going to hang farther.

Here S1 generates an analogy by drawing the picture of an analogous problem involving bending rods instead of stretching springs. This analogy has in fact led him to the correct answer, and provides a plausible but only partial justification for it.

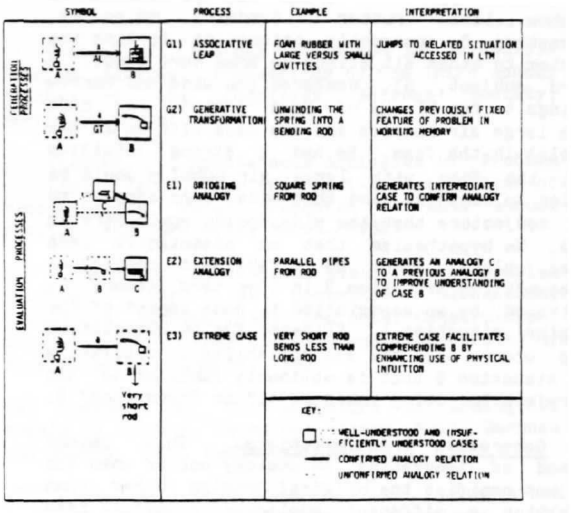


FIG. 2
ANALOGICAL REASONING PATTERNS OBSERVED IN EXPERT PROBLEM SOLVING

Analysis of more complex expert protocols however. makes it apparent that analogical reasoning is not a simple, one-step process, but involves a number of different processes, shown below.

(P1) Generating the Analogy. Given the original conception A of an incompletely understood situation, the analogous conception, B, is generated, or "comes to mind";

(P2) Confirming the Analogy Relation. The analogy relation between A and B must be "confirmed";

(P3) Comprehending the Analogous Case. Conception B must become well understood, or at least predictive;

(P4) Transferring Findings. The subject transfers conclusions or methods from B back to A.

Table 1

The last three processes can occur in any order. Analogies are often proposed tentatively, and processes (P2) and (P3) especially, can be quite time consuming. We have also been somewhat surprised to find that there appear to be not one, but several ways of carrying out each of the above processes. Some of the most important of these sub-processes are shown in fig.2. The figure provides a basic typology of analogical reasoning patterns that have been observed across different subjects. This paper gives an example and brief explanation of each pattern.

ANALOGY GENERATION PROCESSES

Associative leaps. The subject using an associative leap jumps to an analogous situation that differs in many ways from the original problem. A second subject, S2, generated evidence for several associative leaps in the spring problem when he said: "I feel as though I'm reasoning in circles and I think I'll make a deliberate effort to break out of the circle somehow...like rubber bands, molecules, polyesters.." apparently attempting to link the problem to other situations he knew more about. A third subject, S3, compared the wide and narrow springs to two blocks of foam rubber, one made with large air bubbles and one made with small air bubbles in the foam. He had a strong intuition that the foam with large air bubbles would be easier to compress, and this added some support to his conjecture that the wide spring would stretch more. We hypothesize that an associative leap takes place when an established conceptual framework for situation B in long term memory is activated by an association to some aspect of the original situation A. Evidence for an associative leap occurs when the subject shifts to consider a new situation B that is obviously familiar to him or refers to "being reminded of" or "recalling" B.

Generative transformations. This second method of generating an analogy occurs when the subject modifies the original problem rather than recalling a different analogous situation from memory. In other words, the subject transforms the problem by changing an aspect of it which was previously assumed to be fixed. For example, S2 refers to the rod as an "unwound spring". In this case the unwinding of the spring is considered a transformation because the subject is modifying a feature of the spring (its shape) that would ordinarily be held fixed in the problem.

It is hypothesized that a generative transformation occurs when the subject focuses on an internal representation of the existing problem situation A in working memory and changes an aspect of it to create a new but closely related situation B . Thus a generative transformation usually leads to the construction of a new situation B rather than activating an already constructed framework in long term memory.

This subject also generated another analogy via a transformation below while thinking about moving the weight along the spring wire:

S2: Hmmm, what if I imagined moving the weight along the spring? Now what if I recoiled the spring and made the spring twice as long...instead of twice as wide?...uhhh..it seems to me pretty clear that the spring that's twice as long is going to stretch more.

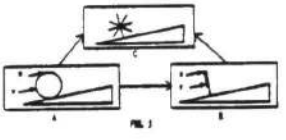
The analogy to the thought experiment of comparing springs of different lengths suggests to him that a wider spring may stretch more than a narrow spring. Notice that the analogy was generated from the rather playful transformation of sliding the weight up and down along the spring wire.

Evidence for a third method, generating an analogy via an abstract principle, has been observed on occasion, but only infrequently [1,2].

ANALOGY EVALUATION PROCESSES

Another finding that has surprised us is the fact that rather than simply generating a single analogy, some subjects generate chains of several analogies. Two types of chains are shown as processes E1 and E2 in fig.2. These are used to evaluate analogies. Processes used to critique and evaluate analogies are at least as important in expert problem solving as processes used to generate them.

Bridging analogies. Determining a match between key relationships in cases A and B is the first and most obvious method for confirming an analogy relation [4,5,6]. However, another interesting process in the form of a "bridging analogy" may also be used. For example, S2 was concerned about the apparent lack of a match between the non-constant slope in the bending rod and the constant slope of a stretched spring. In order to evaluate the bending rod analogy, he constructed the intermediate, bridging example of a spring with square coils as shown in drawing E1-C of fig.2. This allowed him to recognize that restoring forces in the spring come from twisting in the wire as well as bending-- a major breakthrough in his solution which corresponds to the way in which engineering specialists view springs. His discussion of the square spring is evidence for a cognitive bridging analogy, C, which helps him decide whether conceptual frameworks A and B are truly analogous. In this case the square spring analogy eventually aquired the role of a mental model which gave him a new understanding of how springs work.



In a question about whether one can exert a more effective force on a wheel at the top or at the axle (in pushing on the wheel of a covered wagon, for example) several subjects compared the wheel to a lever hinged on the ground (fig.3B). Pushing higher up on the lever would allow it to move a larger weight, they reasoned. Another example of a bridge, which helped one subject to confirm the appropriateness of this analogy is the spoked wheel without a rim shown in fig.3C.

Although physicists usually analyze the wheel problem directly in terms of torques, mathematicians often do not. The reader may be interested in conjecturing about how one mathematician, S4, solved this problem via an analogy to a pulley.

Extension analogies. The diagram for process E2 in fig.2 shows an extension analogy proposed by S1 in the form of two parallel pipes. S1 was hoping to predict whether the radius/stretch relationship in the spring was linear or quadratic or cubic, and his understanding of the bending rod analogy was not sufficient to help him. So he generated a further analogy to the bending rod. In this analogy two pipes are fixed at the left side and held together in such a way that when the weight is applied to the right side, the upper pipe is stretched and the lower pipe is compressed. His analysis of this thought experiment was part of an attempt to model the bending rod in more detail and determine its length/deflection relationship so that this information could in turn be used in analyzing the spring. In such an extension analogy, a second analogous case is used to understand the first analogous case. Thus analogies can be used recursively to understand and evaluate a previous analogous case.

Extreme cases. Aiding in understanding an analogous case is also one of the uses of extreme cases. For example, S2 generated the extreme case

of a very short rod in order to confirm his prior prediction that a short rod would bend less than a long rod (process E3 in fig.2). Other methods of understanding an analogous case are to use a specific fact recalled from memory, a physical intuition, or an analysis in terms of abstract principles [2].

SUMMARY

Fig.2 illustrates several alternative subprocesses for acheiving processes P1, P2, and P3 in Table 1. Together, these subprocesses constitute a collection of intuitive heuristics used by experts in solving problems via analogy. Few of these subprocesses are described by subjects explicitly as they occur (they do not have names for them.) Rather, they must be inferred from patterns in the content of the subject's investigations. Reasoning patterns G1 and G2 in fig.2 are analogy generation patterns. Pattern E1, the bridging analogy, is a method for evaluating an analogy relation. Patterns E2, the extension analogy, and E3, extreme case analysis, are methods for evaluating and improving one's understanding of the analogous case. These reasoning patterns form a set of non-deductive problem solving strategies which: (1) are quite different from traditional problem solving procedures; (2) are associated with imagery reports; and (3) are capable of generating new insights and recognitions of previously undiscovered causal factors in a problem solution [3].

Various "compound solutions" combining two or more of the basic processes shown in fig.2 have also been observed. Our current hypothesis is that most observable chains of reasoning using spontaneous analogies are describable as recursive combinations of these basic patterns.

In the cases of the square spring and the parallel pipes, the novelty of these cases argues that they were at least in part invented by the subject rather than recalled directly from memory. Thus, the analogies observed do not always consist of familiar cases recalled from long term memory; the analogies can also consist of invented cases constructed in working memory. Furthermore, in the square spring and parallel pipes cases, the analogy is used as a mental model which allows the subject to understand the problem situation in a new way. This type of mental model construction appears to be important in the development of creative problem solutions and may play an important role in the development of new explanatory models in science [6].

Research reported in this paper was supported by NSF Award No. SED 8016567.

REFERENCES

[1] Clement, J., Analogy Generation In Scientific Problem Solving, Proceedings of the Third Annual Meeting of the Cognitive Science Society, Berkeley, August, 1981.

[2] Clement, J., Spontaneous Analogies in Problem Solving: Part I- The Progressive Construction of Mental Models. Paper presented at AERA annual meeting, New York City, March, 1982.

[3] Clement, J., Spontaneous Analogies in Problem Solving: Part II- Generation Mechanisms, Simulation, Extreme Cases, and Model Construction, working paper, Physics Department, University of Massachusetts, Amherst, 1982.

[4] Gentner, D., The Structure of Analogical Models in Science, technical report, Bolt, Beranek and Newman, Inc., Cambridge, MA, 1980.

[5] Gick, M. and Holyoak, K.J., Analogical Problem Solving, Cognitive Psychology, 12, 306-355, 1980.

[6] Hesse, M., Models and Analogies in Science. University of Notre Dame Press, Notre Dame, 1966.

[7] Collins, A., Fragments of a Theory of Plausible Reasoning, in Waltz, D., Theoretical Issues in Natural Language Processing-2, Urbana-Champaign: University of Illinois, 1978.

RABBIT: Cognitive Science in Interface Design

by

Michael D. Williams
Frederich N. Tou
Richard E. Fikes
Austin Henderson
Thomas Malone

Cognitive and Instructional Sciences Group
XEROX Palo Alto Research Center
Palo Alto, California

Abstract

A new kind of user interface for information retrieval has been designed and implemented to aid users in formulating a query. The system, called RABBIT, relies upon a new paradigm for *retrieval by reformulation*, based on a psychological theory of human remembering. The paradigm actually evolved from an explicit attempt to design a 'natural' interface which imitated human retreival processes.

To make a query in RABBIT, the user interactively refines partial descriptions of his target item(s) by criticizing successive example (and counterexample) instances that satisfy the current partial description. Instances from the database are presented to the user from a perspective *inferred* from the user's query description and the structure of the knowledge base. Among other things, this constructed perspective reminds users of likely terms to use in their descriptions, enhances their understanding of the meaning of given terms, and prevents them from creating certain classes of semantically improper query descriptions. RABBIT particularily facilitates users who approach a database with only a vague idea of what it is that they want and who thus, need to be guided in the (re)formulation of their queries. RABBIT is also of substantial value to casual users who have limited knowledge of a given database or who must deal with a multitude of databases.

1. Introduction

One way to test a theory is to try to do something useful with it. We have taken a cognitive theory of human remembering together with some artificial intelligence ideas about knowledge representation and used it to design a new paradigm for database retrieval interfaces for casual users. The paradigm is called *retrieval by reformulation*. A small experimental system based on this new paradigm has been implemented in the Smalltalk programming language [Ingalls, 1978] using KloneTalk [Fikes, 1981] on the Xerox Dolphin and Dorado personal computers.

Part of the motivation for designing a new kind of database interface was the unsuitability of existing database interfaces for casual users. Some database interfaces (e.g., SQUARE [Boyce et al, 1975] and SQL [Chamberlin et al, 1976]) require many hours of instruction to learn; others have a syntax which users find difficult to use and understand (e.g., the boolean expressions of DIALOG [Lockheed, 1979]). Interfaces based on the relational data model [Codd, 1970] usually require the user to know in advance which tables and attributes he will be needing, while users of network databases (such as ZOG [Robertson et. al., 1981]) frequently get lost during the course of their search.

To help solve these problems we looked for inspiration to theories of how people retrieve information form their own memory. We believe this approach is promising for two primary reason: (1) To the extent that the interface between a person and his external memory is like the interface between the person and his internal memory the external memory may be easier and more natural to use, and (2) to the extent that human memory systems embody a 'solution' to the problems of retreival from large heterogeneous databases, they may provide useful insights about how to design similar artificial systems.

We began our design process by conjecturing an interface which permitted *descriptive retreival*. The basic tenet of descriptive retrieval is that people retrieve information from (their own) memory by iteratively constructing partial descriptions of the desired target item [Bobrow and Norman, 1975; Norman and Bobrow, 1979; Williams and Hollan, 1981]. The problem was that our conjectured system appeared to give us little more than the traditional boolean expression schemes such as DIALOG. We simply replaced the technical term 'keyword' with the term 'descriptor.' This led us to a re-examination of the problems inherent in boolean expression interfaces.

Upon consideration we conjectured that there were three major sources of difficulty for casual users of interfaces based upon boolean expressions of keywords: (1) the user has incomplete knowledge about the *descriptive terms* needed to create a query (e.g. what car colors does the database know about? red, crimson, rose, mauve?), (2) the user doesn't know what kind of attributes of the item(s) he is seeking the database recognizes (e.g. does the database even has an attribute for car color?), and (3) many users find the syntax of complex boolean expressions difficult to understand.

Yet, if people actually recall information by descriptive retreival then they must face the same problems; they must have some trick to get by those problems. We found such a trick in *retrieval by instantiation*. Retrieval by instantiation postulates that the information retrieved at each iteration of the retrieval process is often in the form of an *instantiation*, i.e., an example item suggested (e.g., analogically or metaphorically) by the partial description [Williams, 1981]. The common consequence of such an instantiation is that one is 'reminded' of something similar to the original item [Schank, 1980; Kolodner, 1980; Bower, Turner, and Black, 1979]. We conjecture that this reminding serves to counter all three of the problems noted for boolean expression schemes. The instantiations provide a *template* for describing the target item, access to the descriptive terms, and can provide the basis for an incremental reconstruction of the target item that avoids much of the complexity inherent in highly structured descriptions.

2. Retrieval by Reformulation

The basic principle underlying RABBIT is a new paradigm, retrieval by reformulation, for information retrieval elaborated from the notion of retrieval by instantiation. The user makes a query by first constructing a *partial description* of the item(s) in the database for which he is searching. RABBIT then provides a description of an *example instance* from the database which matches the user's partial description. Since it is unlikely that the first instance will be exactly what the user is looking for the user can then select any of the attributes shown in the example and incorporate those descriptors, or variations of them, into his partial description, thus, *reformulating* his initial query. At any time the user can request a new example instance, one which matches the latest version of his (partial) description, and then use the descriptors of that new instance to refine his query description still further.

Figure 1 shows RABBIT in the midst of a retrieval interaction. The interface consists of four primary window panes. The 'Description' pane specifies an implicitly defined boolean expression that appears to the user as a partial description of the item(s) he is seeking. The 'Example' pane contains an example item that matches the partial description as of the last user initiated retrieval cycle from the RABBIT defined perspective. More precisely, it contains a description, called the *image*, of an instance from some well-defined *perspective* (e.g., "STAR 8011 computer" can be viewed from the perspectives of "a manufactured product," "a computer," "an electronic device," "a piece of office equipment," and "a piece of stock in a store."). The 'Matching Examples' pane lists instances which satisfy the partial description as of the last retrieval cycle. The 'Previous Description' pane contains the description used on the last retrieval cycle which determines the perspective for presentation of the example and the list of matching examples. The example pane command pop-up menu is also displayed.

The example instance mentioned above is a central element of the interface. It serves several purposes: it functions as a *template*, it permits *access* to additional descriptors, it provides *semantic resolution* of potentially ambiguous terms, and it frequently serves as a *counterexample*.

The example instance is a template in the sense that its presentation provides a pattern for making a query using the descriptors in the instance's image. It permits access to new descriptive terms through the alternatives and describe commands elaborated below.

It also provides semantic resolution in that the context of a term such as the role name 'manufacturer' establishes and refines the term's meaning. The role name 'manufacturer:' could refer to a person or a nation or a corporation. The statement 'manufacturer: Xerox' in the context of a description of a computer product helps resolve a host of potential meanings.

The example instance is also a counterexample to the user's intentions whenever it is not exactly what the user is looking for. Rather than simply permitting the user to express his displeasure with the counterexample and have RABBIT try to guess what is wrong with it, the system tries to encourage the user to articulate what is wrong with the instance presented. The counterexample's simple presence serves to remind the user that his query description is incomplete or wrong and, in addition, point out the particular parts of his description that need correction or modification.

Finally, since the amount of information known about the retrieved instance could be considerable, the information actually presented in the image is limited to be only that information which can be inferred to be relevant based on the query description the user has given so far. (E.g., information concerning the dinner menu or house specialty of a given restaurant would be available from the perspective of "a place which serves food" but not from the perspective of "a business." So if the user had begun his query with the descriptor '**Business**', then the image of the retrieved instance, even if it is a restaurant, would not, initially, include information about its dinner menu.)

The current implementation of RABBIT supports a small set of 5 basic operations for creating a query description given the descriptors provided in the image of the example instance. These operations, shown in figure 1, are require and prohibit (which specify that the given descriptor is or is not to be a descriptor of the retrieved instance, respectively), alternatives (which presents the user with a popup menu of alternative descriptors to the given one), specialize (which shows the specializations of the given descriptor), and describe (which allows the user to examine a description of a given descriptor or to describe recursively what that descriptor should be). The describe command provides the user with the capability to build *embedded descriptions*, an example of which appears in figure 1 with the value of the attribute 'disk' being itself a description. [Tou, 1982] and [Tou, Williams, Fikes, Henderson, and Malone 1982] contain a more complete discussion of the paradigm of retrieval by reformulation and the user interface to RABBIT.

This paradigm of retrieval by reformulation, in effect, defines a form of interaction by which RABBIT can assist casual users in formulating queries. Much of the intelligence of RABBIT comes from control of this interaction by appealing to the conceptual structure of the database.

3. Perspectives

The KL-ONE epistemology for representing knowledge [Brachman, 1979] has had a major influence on the development of RABBIT. One of the main uses of KL-ONE is the implementation of *perspectives*. A perspective is simply a way of describing an event or item from a particular viewpoint [Bobrow and Norman, 1975, Bobrow and Winograd, 1977, Goldstein and Bobrow, 1980, Goldstein, 1980]. In RABBIT, a perspective specifies which descriptors are included in the image of any instance presented to the user. RABBIT perspectives are dynamic in that the perspective from which the user views the instances in the database changes depending on the current partial description and on where he is within the database.

There are two distinct mechanisms RABBIT uses to construct a perspective. First it filters the attributes to be presented to a user by including only attributes implicitly acknowledged by the user. Since the partial description is a representation of the user's intent to the computer, that description is a legitimate basis for determining what information to include in the image of the example instance. In RABBIT the attributes included in the image are exactly those that belong to the instance classes occurring in the partial description. For example, if one were to see the computer descibed in figure 1 retrieved under the partial description '**Product**' (i.e. without the descriptor '**Computer**') then only the attributes 'name', 'manufacturer', and 'cost' would be presented. Once the user refines the partial description to specify that he is seeking a computer, additional attributes (e.g. 'disk:,' 'cpu:,' ...) would appear.

A second mechanism for creating perspectives actually extends the perspective of any given instance beyond attributes directly held by the object. Note in figure 1 that because the user has created an embedded description about the disk of the computer sought, aspects of the disk that the user considers important (e.g. capacity) have been compressed into the image of the computer.

Perspectives serve four main functions in the RABBIT interface:

--controlling the type and amount of information presented
--facilitating the user's understanding of instances
--enforcing certain kinds of semantic consistency
--organizing and managing heterogeneous data.

4. Summary

This paper has briefly described the process of designing a novel type of database interface named RABBIT. RABBIT relies on a new paradigm for information retrieval, *retrieval by reformulation*, derived from a cognitive science theory of human remembering together with some artificial intelligence ideas about knowledge representation. The four main ideas underlying this paradigm are:

1) retrieval by constructed descriptions
2) interactive construction of queries
3) critique of example instances
4) dynamic perspectives.

The first three of these ideas had their origins in human psychology. The development of the fourth idea—dynamic perspectives—was motivated and influenced strongly by the KL-ONE knowledge representation language.

Cognitive Science has played a crucial role in the design of RABBIT. We take the tentative success of the design as an indication of the potential role of cognitive science in the design of human-computer interfaces.

Acknowledgements

A major portion of this work was carried out by the second author under the auspices of the MIT intern program at Xerox PARC. The authors would also like to acknowledge the original stimulus for this work steming from an exciting conference on artificial intelligence and human-computer interfaces sponsored by the Army Research Institute. In particular, we would like Stan Halpern, Janet Kolodner, and Alan Badre to know a part of what came from their efforts in putting that conference together. We would also like to thank John Seely Brown, Tom Moran, Rick Cattell, Laura Gould, and Richard Burton for their patient discussions and guidance. Each contributed crucial pieces of the puzzle many of which we are still putting together. John Seely Brown's questions in particular guided our pursuit of the use of perspectives. Finally, we would like to thank the other members of the Cognitive and Instructional Sciences Group at Xerox PARC for their continuing support and critique throughout the development of RABBIT.

References

Bobrow, D.G., and Norman, D.A. "Some Principles of Memory Schemata," in D.G. Bobrow and A.M. Collins (Eds.), *Representation and Understanding: Studies in Cognitive Science*. New York: Academic Press, 1975.

Bobrow, D.G., and Winograd, T. "An Overview of KRL: A Knowledge Representation Language," *Cognitive Science*, 1, pp. 3-46, 1977.

Bower, G.H., Black, J.B., and Turner, T.J. Scripts in Text Comprehension and Memory, *Cognitive Psychology*, Vol 1, 177-220. 1979.

Boyce, R.F., Chamberlin, D.D., King, W.F., and Hammer, M.M. "Specifying Queries as Relational Expressions: The SQUARE Data Sublanguage," *Communications of the ACM* **18**, 11 (Nov. 1975), pp. 621-628.

Brachman, R.J., Bobrow, R.J., Cohen, P.R., Klovstad, J.W., Webber, B.L., Woods, W.A. "Research in Natural Language Understanding: Annual Report, 1 September 1978 to 31 August 1979," *BBN Report No. 4274*. Cambridge, MA: Bolt Beranek and Newman Inc., August, 1979.

Chamberlin, D.D., Astrahan, M.M., Eswaran, K.P., Griffiths, P.P., Lorie, R.A., Mehl, J.W., Reisner, P., and Wade, B.W. "SEQUEL 2: A Unified Approach to Data Definition, Manipulation, and Control," *IBM Journal of Research and Development* **20** (Nov. 1976), pp. 560-575.

Codd, E.F. "A Relational Model of Data for Large Shared Data Bases," *Communications of the ACM* 13, 6 (June 1970), pp. 377-397.

Fikes, R. "Highlights from KloneTalk: Display-Based Editing and Browsing, Decompositions, Qua Concepts, and Active Role-Value Maps," Proceedings of the 1981 KL-ONE Workshop, Jackson, New Hampshire, October, 1981.

Goldstein, I.P. "PIE: A network-based personal information environment." *Proceedings of the Office Semantics Workshop*, Chatham, Mass., June, 1980.

Goldstein, I.P., & Bobrow, D. Descriptions for a programming environment, *Proceedings of the First Annual National Conference on Artificial Intelligence*, Stanford, CA, August, 1980.

Ingalls, D.H. "The Smalltalk-76 Programming System: Design and Implementation," *Conference Record of the Fifth Annual ACM Symposium on Principles of Programming Languages*, Tucson, AZ: January 1978, pp. 9-16.

Kolodner, J.L. Retrieval and Organization Strategies in Conceptual Memory: A Computer Model. Research Report #187, Department of Computer Sciecne, Yale University, New Haven, CT. 1980.

Lockheed Information Systems. *Guide to DIALOG Searching*, Palo Alto, CA, 1979.

Norman, D.A., and Bobrow, D.G. "Descriptions: An Intermediate Stage in Memory Retrieval," *Cognitive Psychology* **11** (1979), pp. 107-123.

Robertson, G., McCracken, D., and Newell, A. "The ZOG Approach to Man-Machine Communication," *International Journal of Man-Machine Studies* (1981) 14, pp. 461-488.

Schank, R.C. Failure-driven memory. *Cognition and Brain Theory*, Vol. 1, 4, 41-60, 1980.

Tou, F. *RABBIT: A novel approach to information retrieval*, unpublished M.S. thesis, Massachusetts Institute of Technology, Cambridge, Mass., forthcoming.

Tou, F.N., Williams, M.D., Malone, T.W., Fikes, R.E., and Henderson, A. RABBIT: an Intelligent Interface. Xerox Technical Report, forthcoming, 1982.

Williams, M.D. "Instantiation: A Data Base Interface for the Novice User," Xerox Palo Alto Research Center Working Paper, 1981.

Williams, M.D., and Hollan, J.D. "The Process of Retrieval from Very Long Term Memory," *Cognitive Science* **5** (1981), pp. 87-119.

Zloof, M.M. "Query by example," in *Proceedings of the National Computer Conference*, AFIPS Press, Arlington, Va., May 1975, pp. 431-437.

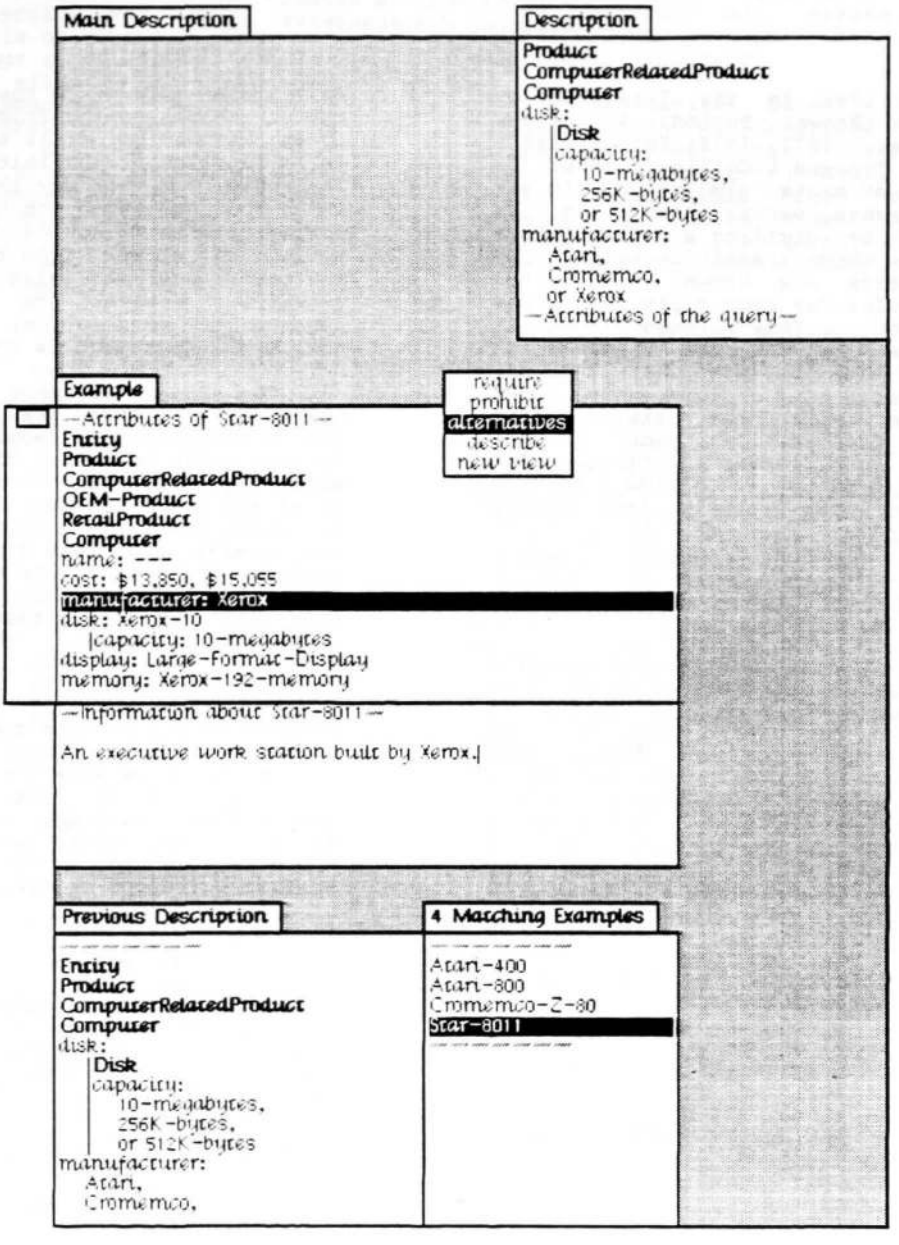


Figure 1. Example of RABBIT display.

CONSTRUCTING RUNNABLE MENTAL MODELS

Allan Collins
Dedre Gentner

Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, Massachusetts 02238

A core idea in the literature on mental models (Brown, Burton, & Zdybel, 1973; deKleer, 1977, 1979; Forbus, 1981; Hayes, 1978; Stevens & Collins, 1980) is the notion of mental simulation. In all these approaches mental simulation is accomplished by dividing a system into a set of states whose transition rules from state to state are known. Given the transition rules for each state, and the topology of connections between states, it is possible to run the system with different inputs to see what happens. This provides a kind of inferential power not possible with the static data structures implied in much of the literature on frames, scripts, and semantic networks (e.g., Collins & Loftus, 1975; Minsky, 1975; Quillian, 1968; Schank & Abelson, 1977).

The Metaphor Hypothesis

In this paper we propose a specific role for metaphor in constructing runnable mental models. It can be stated as follows: Metaphors map the set of transition rules from one domain (the base) into another domain (the target) so that it is possible to construct a mental model to run simulations in the target domain. This is a special case of Gentner's (1980, 1982) more general claim that metaphor is a mapping of structural relations from a base domain to a target domain.

We can illustrate the hypothesis by showing how three metaphors can be used to construct a runnable version of the microscopic model of evaporation discussed by Stevens and Collins (1980). Then, in the next section, we compare the model derived from these metaphors with the model one of our subjects used to reason about evaporation in an experiment where we asked subjects novel questions about evaporation processes.

The first metaphor states that water molecules (or air molecules) are like billiard balls bouncing around in space. The warmer the water is, the more velocity (or greater energy) the average molecule has. The same metaphor applies to the water and air molecules in the air mass above a body of water. This model is incomplete insofar as it includes no notion of the attractive forces between different molecules and the polarity of the electrical charges on different sides of the molecule. But as a first approximation, it is a perfectly good model.

The second metaphor states that a molecule escaping from the water is like a rocket ship escaping from earth. That is to say whether or not it actually escapes is a function of its initial velocity and its angle. In this way the model builds in a rudimentary notion of the attractive forces between molecules, by likening the notion of escape from the attraction of the other water molecules to escape from gravity. However, to understand some aspects of evaporation, this gravity notion of attraction is not enough.

The third metaphor states that the molecules in the air mass over the water can be thought of as people inside a room. As more water molecules collect in the air mass, the room becomes more crowded with water and air molecules. The warmer the air mass, the larger the room. Thus, warm air masses are less dense then cold air masses. The boundary between the air and water is the entry into the room, and if everyone crowds along that border it is hard to get in. This crowded-room metaphor leads to many correct predictions, but is wrong in some fundamental ways. In fact, the space between molecules in a cool air mass never becomes crowded. Cool air masses hold less moisture because the water molecules in them tend to lose energy with each interaction. Then the attractive forces between water molecules tend to attract the molecules back to the water surface or to form raindrops or dew.

Now we want to show how these three metaphors enable a person to construct a runnable model of evaporation processes. We would argue that people usually know certain interaction rules of billiard balls such as those depicted in Figure 1. Velocity of each ball in the interaction is represented by a vector, and the transition rule of the interaction by the large arrow. Rule 1 shows that without collision, speed and direction are maintained. Rule 2 shows a head-on collision with a non-moving ball where momentum is transferred from one ball to the other. Rules 3 and 4 show how momentum is transferred as a moving ball strikes a non-moving ball at different angles. Rules 5 and 6 show typical interactions when both balls are moving. These rules summarize one's local knowledge about how billiard balls interact.

From these local interaction rules, one can derive certain global properties of how a container full of molecules will behave. That is we can construct an aggregate model of molecular interaction (Stevens & Steinberg, 1981) based on the mechanical model of billiard-ball

interaction. The most important properties of this aggregate model are that there is variability of speed and direction of the molecules. This produces randomness of motions of the molecules, with some going toward the surface, some not. There is elasticity of interaction so that energy can be transferred from molecule to molecule, but not lost. Finally, there is no change in direction or velocity without a collision. In our view, people can either imagine molecules moving in this aggregate fashion (like seeing dust particles moving in the sunlight) or by following a single molecule moving around and encountering other molecules according to the local interaction rules shown in Figure 1.

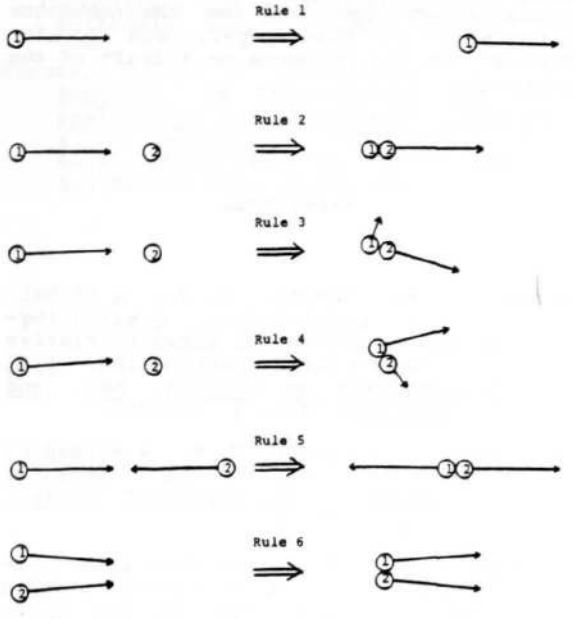


Figure 1. Some interaction rules for perfectly elastic billiard balls.

The rocket-ship metaphor gives a simple three state description of behavior of molecules near the surface. If they have any downward component of velocity, they do not escape. If they are headed straight up, there is some minimum initial velocity they need to escape. If they are headed up at an angle to the surface, the smaller the angle the greater the initial velocity they need to escape (because of the attraction of the surface over a larger part of the trajectory). This three state model summarizes what the rocket-ship metaphor implies about the effects of the water's surface.

The crowded-room metaphor, like the billiard-ball metaphor, leads to construction of an aggregate model at the microscopic level. The model has the following behavior. The warmer the air mass, the larger the room is. As water evaporates into the air mass, it fills up with molecules. Cold air masses take less time to fill up with molecules. When the air mass is filled, then no more water molecules can get in. If the air mass does not mix completely (depending on winds), then water molecules may accumulate in the air along the water's surface and no new molecules can get in, even though the air mass is not filled. If a crowded air mass is cooled, the water molecules may be squeezed out for lack of space. These behavioral properties reflect the way air masses actually behave, even though the model is essentially incorrect.

In an earlier paper (Stevens & Collins, 1980) we described the kind of inferential power that runnable models provide for answering novel questions about the world. In order to see how subjects use models, we conducted an experiment where we asked subjects to reason about such questions.

Experiment on Mental Models

Four subjects were asked eight questions about evaporation. They were asked to explain their reasoning on each question. All were reasonably intelligent, but were novices about evaporation processes. Our analysis will center on one subject, whose model of evaporation processes was very much like the model we constructed from the three metaphors, if not exactly the same model. His view includes notions of the energy needed for molecules to escape from a body of water and the difficulty of water molecules entering a cold air mass because of the higher density. Nowhere does he mention attractive forces between molecules, which suggests that this notion is not part of his model. He seems to share a common misconception that visible clouds (such as one sees coming out of a boiling kettle) are made up of water vapor rather recondensed liquid water. This misconception forced him into several wrong explanations.

We will present the portions of his responses to three of the questions that illustrate his use of the mental model described above.

Q2: On a cold day you can see your breath. Why?

S: I think again this is function of the water content of your breath that you are breathing out. On a colder day it makes what would normally be an invisible gaseous expansion of your breath (whatever), it makes it more dense. The cold temperature causes the water molecules to be more dense and that in turn makes it visible relative to the surrounding gases or relative to what your breath would be on a warmer day, when you don't get that cold effect causing the water content to be more dense . . .

Q4: Which will evaporate faster, a pan of hot water placed in the refrigerator or the same pan left at room temperature? Why?

S: When I first read that question, my initial impression, that putting a pan of hot water in the refrigerator you suddenly have these clouds of vapor in it, threw me off for a second. I was thinking in terms of there is a lot evaporation. Well I guess, as I thought through it more, I was thinking that it was an indication of more evaporation, but it was just (let us say) the same evaporation. Immediately when you put it in anyway, it was more visible. Ahmm, as I think through it now, my belief is that it would evaporate less than the same pan left standing at room temperature and my reasoning there is that the air in the refrigerator is going to be relatively dense relative to the room temperature air, because at a colder temperature again its molecules are closer together (what not), and that in effect leaves less room to allow the molecules from the hot water to join the air. . . .

Q5: Does evaporation affect water temperature? If so, in what way, and why?

S: I guess those water molecules that do leave the surface of the water are those that have the highest amounts of energy. I mean they can actually break free of the rest of the water molecules and go out into the air. Now if they have a, if they are the ones with the most energy, I guess generally heat is what will energize molecules, then that would lead me to believe that maybe, although it may not be measurable, maybe with sophisticated instruments it is, but maybe it would be measurable after your most energetic molecules have left the greater body of water. Those that remain are less energetic and therefore their temperature perhaps less.

The subject's first two answers manifest the crowded room model: The particles in cold air are crowded together, which acts to make one's breath more visible and to make it more difficult for water molecules from a hot pan to get in. The last answer manifests the rocket ship and billiard ball models: The particles move around and those that escape are the high energy particles, leaving the low energy particles behind and hence cooling the water.

These excerpts illustrate the underlying molecular model of evaporation that the subject had, and how he used it to find answers to novel questions. His model is close to, if not the same as, the model we constructed from the metaphors in the previous section. The hypothesis of the paper is that this subject's underlying model was constructed by pasting together his models of how familiar objects behave. While he may not have drawn upon billiard balls, rocket ships, and crowded rooms, he must have drawn upon some such objects in order to create the model he was using. Based on this model, he was able to deal quite successfully with the questions, even though his model was incorrect in several ways.

Acknowledgments

This research was supported by the Personnel and Training Programs, Psychological Sciences Division, Office of Naval Research, under contract number N00014-79-C-0338, Contract Authority Identification Number NR 154-428. We thank Michael Williams for an engaging conversation that led to the metaphor hypothesis of this paper, and to Ken Forbus for his comments on a draft of the paper.

References

Brown, J. S., Burton, R. R., & Zdybel, F. A model-driven question-answering system for mixed-initiative computer-assisted instruction. IEEE Transactions on Systems, Man, and Cybernetics, 1973, 3, 248-257.

Collins, A., & Loftus, E. F. A spreading activation theory of semantic processing. Psychological Review, 1975, 82, 407-428.

de Kleer, J. Multiple representations of knowledge in a mechanics problem solver. Proceedings of the Fifth International Joint Conference on Artificial Intelligence. Cambridge, Mass.: MIT, 1977, 299-304.

de Kleer, J. The origin and resolution of ambiguities in causal arguments. Proceedings of the Sixth International Joint Conference on Artificial Intelligence. Tokyo, Japan: 1979, 197-203.

Forbus, K. D. A study of qualitative and geometric knowledge in reasoning about motion. Cambridge, Mass.: MIT AI Technical Report No. 615, 1981.

Gentner, D. Studies of metaphor and complex analogies: A structure-mapping theory. Paper presented at the A.P.A. Symposium on Metaphor as Process, Montreal, September 1980.

Gentner, D. Are scientific analogies metaphors? In D. S. Miall (Ed.), Metaphor: Problems and perspectives. Brighton, Sussex: Harvester Press, Ltd., 1982.

Hayes, P. J. Naive physics: Ontology for liquids. Unpublished manuscript, 1978.

Minsky, M. A framework for representing knowledge. In P. H. Winston (Ed.), The psychology of computer vision. New York: McGraw-Hill, 1975.

Quillian, M. R. Semantic memory. In M. Minsky (Ed.), Semantic information processing. Cambridge, MA: The MIT Press, 1968.

Schank, R. C., & Abelson, R. P. Scripts, plans, goals and understanding. Hillsdale, N.J.: Erlbaum, 1977.

Stevens, A. L., & Collins, A. Multiple conceptual models of a complex system. In R. E. Snow, P. Federico, & W. E. Montague (Eds.), Aptitude, Learning, and Instruction (Vol. 2). Hillsdale, N.J.: Erlbaum, 1980.

Stevens, A. L., & Steinberg, C. A typology of explanations and its application to intelligent computer aided instruction (BBN Report No. 4626). Cambridge, MA: Bolt Beranek and Newman Inc., March 1981.

Bi-Directional Inference

by

Stuart Shapiro, Joao Martins and Donald McKay*

Department of Computer Science
State University of New York at Buffalo
4226 Ridge Lea Road, Amherst, NY 14226

*(current address: Research & Development Activity, Special Systems Division, Federal and Special Systems Group, PO Box 517, Paoli, PA 19301)

This work was supported in part by the National Science Foundation under Grants MCS878-02274 and MCS80-06314 and by the Instituto Nacional de Investigacao Cientificia (Portugal) under Grant No. 20536.

Abstract

Inference can be viewed as a search through a space of inference rules. Backward and forward inference differ in the direction of the search: backward inference searches from goals to ground assertions; forward inference searches from ground assertions to goals. This paper describes an inference procedure, called bi-directional inference, which limits the number of inference rules searched. Bi-directional inference results from the interaction between forward and backward inference and loosely corresponds to bi-directional search. We show through an example that, when used throughout a session of related tasks, bi-directional inference sets up a conversational context and prunes the search through the space of inference rules by ignoring rules which are not relevant to that context.

1. Introduction

Bi-directional inference (BDI) combines forward inference (FI) and backward inference (BI) to limit the search through a space of inference rules by establishing a context on the basis of an ongoing session. We use the term "bi-directional inference" because the resulting search loosely corresponds to bi-directional search (Kowalski 72, Pohl, 71).

The benefits of BDI become clear during an extended session in which the user asks questions and adds assertions all of which are related. BDI sets up a conversational context and prunes the space of inference rules searched (either during BI or FI) by ignoring rules which are not relevant to the context.

In BDI there are two sets of inference frontiers, one growing from the assertions added in FI and the other growing during BI from the questions asked. Whenever two frontiers meet some answers are produced.

BDI has been implemented in SNIP, the SNePS Inference Package. We present examples of BDI and compare the results obtained using BDI with the results obtained using BI or FI only. Although SNIP has a much richer rule syntax than used in these examples (Shapiro, 79a, 79b) they suffice to illustrate BDI.

2. Basic notions of SNIP

SNIP relies on a declarative representation of inference rules (SNePS semantic network (Shapiro, 79a). Every rule may be used both in FI and BI. When a rule is used, it is activated, remaining that way until explicitly de-activated by the user. The activated rules are assembled into an active connection graph (acg) (McKay and Shapiro, 81), a collection of MULTI processes (McKay and Shapiro, 80) which carry out the inference. The acg also stores all the results generated by the activated rules. If during some deduction SNIP needs some of the rules activated during a previous deduction, it uses their results directly instead of rederiving them. The acg that is built for one query or assertion is not discarded after the query has been answered or the assertion "fully" understood by making all possible inferences from it. Rules of the network remain active, allowing a dynamic context to be constructed. The dynamic context is the collection of rules which have been activated. In addition, the active rules are more prominent: when searching for inference rules to be used, if any previously activated rules are appropriate then only those rules will be considered and no other rules will be activated. Hence rules apparently irrelevant to the current dynamic context are ignored.

3. Backward Inference

We present an example of BI, explaining very briefly how acg's work. A complete explanation can be found in (McKay and Shapiro, 81).

Suppose that SNIP is being used as a database retrieval system for some company interested in recruiting computer science (CS) majors. The recruiting policies of the company are stored as rules in the database (Lines 1-4, Fig. 1). The

```
V(x,y)[ Planning-to-visit(x) & CS-major-at(y,x) -> Good-prospect(y)]
V(x,y)[ Top-school(x) & CS-major-at(y,x) -> Good-prospect(y)]
V(x)[ Good-prospect(x) -> Send-literature-to(x)]
V(x)[ Good-prospect(x) & Graduating(x) -> Invite-for-interview(x)]
Top-school(MIT)
Top-school(CMU)
CS-major-at(Don,SUNY)
CS-major-at(Ted,CMU)
CS-major-at(Anna,MIT)
CS-major-at(John,UCLA)
```

Figure 1
Initial database

company's database also contains a list of top schools and a list of the CS majors at different schools (Lines 5-10, Fig. 1).

Every year the company updates its database with the names of all students graduating in CS and all the schools that the company will visit during that year (Fig. 2). The company then uses SNIP to find out

```
Planning-to-visit(SUNY)
Planning-to-visit(CMU)
Graduating(Don)
Graduating(Ted)
Graduating(John)
```

Figure 2
Information updating the database

which CS majors should be invited for interviews, which ones should be sent the company's literature, etc.

We now consider the acg describing the reasoning of SNIP when it is asked who should be invited for an interview.

An acg is represented as rectangles and circles. Each rectangle represents a rule instance (a deduction rule together with a substitution for the variables in the rule); the antecedents appear to the left of the double line and the consequents to the right. Circles (called goal nodes) represent goals to be proved. Rule instances and goal nodes are connected by directed edges. Substitutions flow through the edges. Rule instances and goal nodes can be viewed as producers of formulas sent out on the edges leaving them and as consumers of formulas coming in on the edges pointing to them. Some edges have switches (represented by square brackets) which have the effect of renaming the variables in the substitutions flowing through them. For ease of reference, rule instances have labels of the form An (where n is an integer). Those labels are used for notational convenience only and have no relation with the way acg's work.

Initially, a request is created which contains the atomic formula being sought. The rule instance labeled A1 in Figure 3 represents the request to

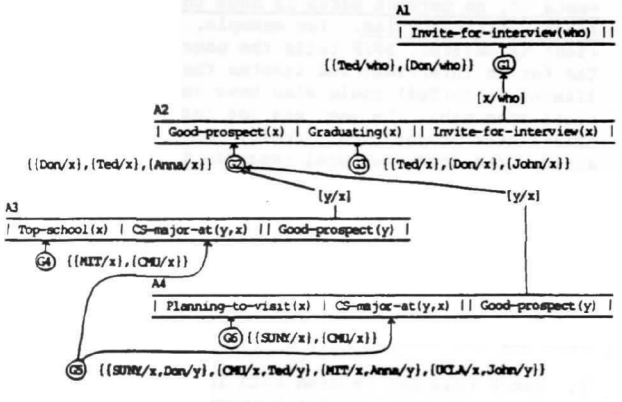


Figure 3
acg for backward inference

deduce all instances of the atomic formula Invite-for-interview(who). The next step is to create a goal node for the atomic formula. A goal node (G1) is added below the instance being sought. One of the jobs of the goal node is to match its atomic formula against the network to find all formulas which unify with it. If there are ground instances the goal node produces them immediately. For every matching formula in consequent position of some rule, a new rule instance is added to the acg. The other job of the goal node is to remember all substitutions it receives (these substitutions are represented enclosed in curly brackets next to the goal node). When a goal node receives a new substitution, it sends it to all rule instances to which it points. In this case, G1 can't find any ground instances of Invite-for-interview(who) but the rule $\forall(x)$[Good-prospect(x) & Graduating(x) → Invite-for-interview(x)] may be used to derive such instances. Rule instance A2 is thus created by G1 (Fig. 3). Notice that the variable 'x' in A2 should be bound to 'who' in A1 when an answer is produced by A2. For this reason a switch ([x/who]) is inserted in the link between A2 and G1 and has the effect of translating between variable contexts. Switches are computed by the network matching function (Shapiro, 77) which was used by G1. For details of how this is done see (McKay and Shapiro, 81).

Goal nodes G2 and G3 are created for the antecedents of A2. G2 finds two rules which can produce instances of Good-prospect(x) and creates the corresponding rule instances (A3 and A4, Fig. 3). G3 finds three ground instances of Graduating(x), namely Graduating(Ted), Graduating(Don) and Graduating(John). The substitutions {Ted/x}, {Don/x} and {John/x} are stored by G3 and sent to its consumer (A2).

Goal nodes are created for the antecedents of A3 and A4. G4 finds two top schools (MIT and CMU), and sends the substitutions to A3. G5 finds the CS majors at different schools, informing both A3 and A4. A3 deduces that both Ted and Anna are good prospects. A4 deduces that both Don and Ted are good prospects after receiving from G6 the information that SUNY and CMU will be visited.

The information about good prospects flows through the acg reaching A2 which deduces that both Ted and Don (good prospects who are graduating) should be invited for interviews and the answer is finally produced by A1.

Notice that BI tries to get each answer in all possible ways, and so the same answer can be produced several times. In this particular case the answer Good-prospect(Ted) was produced twice, by rule instances A3 and A4.

4. Forward Inference

In this section we discuss the results obtained if the company chooses to use FI. We will assume that the information represented in Figure 1 is stored in the database and that FI is done with the information represented in Figure 2.

Doing FI with Planning-to-visit(SUNY) generates the acg of Fig. 4: rule instance A1 is created along with goal nodes for its antecedents (G1 and G2). G1 is immediately satisfied, and G2 finds CS-major-at(Don, SUNY), sending to A1 the substitution {Don/y}. Notice that G2 is performing some amount of BI, reflecting a characteristic of SNIP in which BI and FI are closely interconnected. A1 deduces Good-prospect(Don), creating rule instances A2 and A3 to do further FI. A2 deduces its consequent but A3 doesn't since Graduating(Don) is not in the database yet.

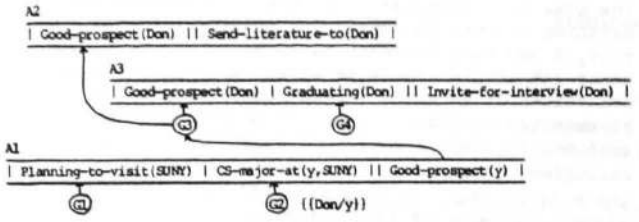


Figure 4
acg for forward inference

After entering all the information of Figure 2, SNIP has deduced (acg not shown) that Don, Ted and Anna should be sent literature and that Don and Ted should be invited for interviews. In other words, all possible inferences were made, even if the user was only interested in some of them. FI does not take the user's interests into account filling the database with assertions which may never be used.

5. Bi-directional Inference

In this section, we introduce BDI and show that it establishes conversational contexts, focusing SNIP's inferences within those contexts and thereby limiting the space of rules searched. BDI results from the interaction of FI and BI and can be obtained either by doing BI following FI or by doing FI following BI. We consider each of these cases in turn.

5.1. Backward Inference Following Forward Inference

Suppose that the user says "I am planning to visit SUNY, who shall I invite for an interview?". In this context, by asking 'Invite-for-interview (who)?' the user wants to consider only the CS majors from SUNY. We show how FI can be used to set up the 'SUNY context' which is then used to answer the user's query. In a pure BI system, finding the CS majors from SUNY who should be invited for an interview requires finding the intersection between all CS majors from SUNY and all persons who should be invited for an interview (or, in some systems, generating all of one and testing each to see if it satisfies the other).

The user begins by doing a small amount of FI with Planning-to-visit(SUNY). The amount of inference can be defined by the number of network pattern matches performed. Let us assume, for the sake of argument, that by "small amount of FI" we mean that FI is only allowed two network matches. The first match finds the rule $\forall(x,y)$[Planning-to-visit(x) & CS-major-at(y,x) → Good-prospect(x)], setting up the rule instance A1 (Fig. 5) and the second match is used by G2 to look for instances of CS-major-at(y,SUNY), finding CS-major-at(Don,SUNY). This is enough to deduce Good-prospect(Don) but nothing can be done with this because finding unactivated rules requires a match. Therefore, the inference stops, leaving behind the active rule instance A1 (Fig. 5).

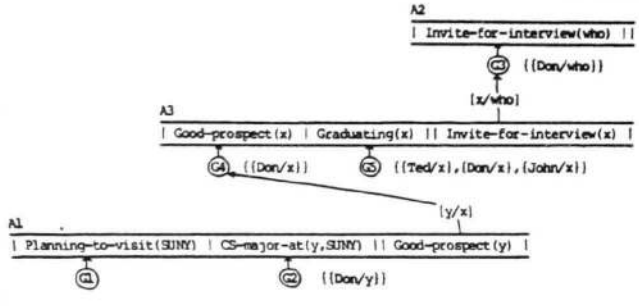


Figure 5
acg for bi-directional inference

If the user now asks the question 'Invite-for-interview(who)?' rule instances (A2 and A3) and goal nodes (G3, G4 and G5) are created (Fig. 5) as discussed in section 3. Here, however, goal node G4 finds that there is an active rule that can produce instances of Good-prospect(x), namely A1. Instead of doing a network pattern match to find additional rules, it uses rule instance A1 immediately. The substitution {Don/y} flows through the acg producing the answer Invite-for-interview(Don). In this case the CS majors from other schools were not even considered since SNIP had set up the "SUNY context".

Suppose that CS-major-at(Don,SUNY) were not in the network and thus rule instance A1 could not produce any answer even though instances of Invite-for-interview(who) could have been derived for CS majors of other schools. Following the query 'Invite-for-interview(who)?', SNIP would return an "I don't know" answer. This, at first glance, seems to be wrong. However, taking into account that the user only wants to consider the CS majors from SUNY this makes perfect sense, showing a feature of BDI in which derivable instances which are irrelevant to the context are effectively ignored by SNIP.

5.2. Forward Inference Following Backward Inference

Suppose that the database contained the information of Figure 1 and the user asked who should be invited for an interview. SNIP builds an acg as shown in Figure 3, except that goal nodes G3 and G6 have no stored data. The acg produces no answers since the information in the database is insufficient. If the user now does FI with any of the propositions of Figure 2, the waiting goal nodes are found. Whenever a new assertion is produced for FI, and a goal node already exists that wants it, no network match is done to find additional relevant rules. For example, if Graduating (Ted) is entered, SNIP tells the user to invite Ted for an interview, and ignores that Send-literature-to(Ted) could also have been derived, since presumably the user was not interested in this latter proposition. Again, BDI takes into account the conversational context, ignoring the rules irrelevant to the active context.

6. Conclusions

We presented an overview of BDI, pointing out the two characteristics required by a system to make the BDI behavior possible:

1. Every rule may be used both in FI and BI.
2. There is a distinction between rules which have been activated and rules which haven't.

Relying on these two characteristics, when SNIP (a system which uses BDI) searches for rules to be used, it looks for activated rules first and just in case of failing to find any activated rule, non-activated rules are considered. In addition, as a matter of efficiency, activated rules remember all the results produced, not solving the same problem twice. The resulting inference loosely

corresponds to a bi-directional search. We say 'loosely corresponds' because not only may there be several bi-directional searches going on in parallel (one for each question asked) which can intersect each other, but also there are two levels of search, the first through the activated rules, and the second, which is tried only after failure of the first, through the non-activated rules.

The example presented, although very small and simplistic, shows that BDI effectively prunes the search through the space of inference rules by focusing the system's attention towards the interests of the user.

In BDI, some of the disadvantages of pure FI and pure BI do not exist. One of the disadvantages of pure FI is that it may fill the database with derived propositions which may never be used. We showed that BDI ignores some derivations which do not interest the user. One of the disadvantages of BI is that all apparently relevant rules are tried, regardless of the actual data. We showed that BDI ignores inactive rules in favor of rules activated by previous (forward or backward) deduction.

7. Acknowledgements

Many thanks to Terry Nutter, Ernesto Morgado, Jeannette Neal and the other members of SNeRG (the SNePS Research Group) for their comments on earlier versions of this paper and for their general discussions while the research was in progress.

8. References

1. Kowalski, R., And-or Graphs, Theorem-proving Graphs and Bi-directional Search, in Machine Intelligence 7, Meltzer and Michie (eds.), Halsted Press, 1972.
2. McKay D. and Shapiro S., "MULTI - a LISP Based Multiprocessing System", Proc. 1980 LISP Conference, pp. 29-37.
3. McKay D. and Shapiro S., "Using Active Connection Graphs for Reasoning with Recursive Rules", Proc. IJCAI-81, pp. 368-374.
4. Pohl I., Bi-directional Search, in Machine Intelligence 6, Meltzer and Michie (eds.), American Elsevier, 1971, pp. 127-140.
5. Shapiro S., "Representing and Locating Deduction Rules in a Semantic Network", in Proc. Workshop on Pattern-Directed Inference System, SIGART Newsletter 63, 1977, pp. 14-18.
6. Shapiro S., "The SNePS Semantic Network Processing System", in Associative Networks, N.V. Findler (ed.), Academic Press, 1979a, pp. 179-203.
7. Shapiro S., "Numerical Quantifiers and their use in Reasoning with Negative Information", Proc. IJCAI-79, 1979b, pp. 791-796.
8. Shapiro S. and McKay D., "Inference with Recursive Rules", Proc. First AAAI Conference, 1980, pp. 151-153.

ACTIVELY LEARNING TO USE A WORD PROCESSOR

John M. Carroll and Robert Mack
IBM Thomas Watson Research Center
Yorktown Heights, New York 10598

Learning to use a word processor provides a study of real complex human learning that is fundamentally "active", driven by the initiatives of the learner. People learn by actively trying things out, by reasoning, and by referring to prior knowledge. Our view is that these are natural -- albeit demanding -- strategies for people to adopt when confronted by a learning task of non-trivial complexity. What is especially noteworthy in the present case is that the learners we have studied are almost entirely innocent with respect to computer technology. In the context of learner innocence, we argue, these "natural" strategies entrain severe and wide ranging learning problems. Analysis of these problems, in turn, suggests research directions for the analysis of real human learning within Cognitive Science and practical directions in which computer word processing systems, and the educational technologies that support their training and use, might evolve.

In this research project, ten office temporaries spent four half-days learning to use one of two possible word processing systems in our laboratory. These people were highly experienced in routine office work, but quite naive with respect to computers in general and word processing in particular. We asked them to imagine a scenario in which a word processing system had recently been introduced to their office and they had been asked to be the first to learn it (to then pass this knowledge on to colleagues). The point was that they were to learn to use the system using the training materials that accompany it as their only resource.

Our method involved prompting learners to "think aloud" as they worked through the training materials. They were to report questions that were raised in their minds, plans and strategies they felt they might be considering or following out, and inferences and knowledge that might have been brought to awareness by on-going experiences. We remained with the learners, to keep them talking and to intervene if at any time it appeared that a problem was so grave that a learner might leave the experiment if we did not help out. Our prompting remained non-directive, and indeed once learners got going we needed to prompt very infrequently. Our analysis consisted first of an enumeration of "critical incidents", constrained by the consensus of the experimenters, which were cataloged and classified in various ways. The chief goal of this was to form a picture of the typical experience of a learner, and it is this induced "prototype" learning experience to which we will refer in what follows.

Learning by doing.

Our learners relentlessly wanted to learn by trying things out rather than by reading about how to do them. Half of our learners tried to sign on to the word processor before reading how to do so. In part this was impatience: they were reluctant to read a lot of explanation or get bogged down following meticulous directions. But it also devolved from mismatched goals: Learners wanted to discover how to do specific things at particular times, and this did not always accord with the sequence in which topics were treated in the manual.

Learning by trying things out according to a personal agenda of needs and goals is not merely a preference. Learners who try to follow out manual instructions are often unable to do so. The instruction sequences are fragile in the sense that it is easy to get side-tracked and there is no provision in them for recovery. One example is a learner who inadvertently paginated (reformatted) a document at the beginning of an exercise on revising documents. This not only rearranged the lines in the file to make right margins even, it also stored the document away. The learner had not yet learned how to retrieve documents and the manual itself provided no recovery information for this (or any other) type of error. Accordingly, she was forced to try to discover how to retrieve the document on her own.

Once the document was restored, she was faced with an equally staggering problem: the pagination operation had rearranged the lines of her file so that the revising instructions did not refer to the same document. An experienced user who understood reformatting could have reinterpreted the instructions and adapted them to this rearranged text. But this learner had no idea what she had done, and thus was puzzled by the fact that the instructions seemed to be wrong. The fragility of instruction sequences, coupled with the propensity of learners to try to recover by initiating exploratory forays, can result in problem tangles: Learners, who may not even fully understand the individual operations, have little basis for appreciating the subtle interdependence of clusters of word processor operations. They find themselves in distorted or even unrecognizable problem situations.

When learners do not, or cannot, follow directions the problems that arise can result in their losing track of what they are trying to do. It is likely, of course, that this loss of task orientation contributes to the overall failure of learning -- as indicated by the trouble all learners had applying their learning experiences to the routine typing "transfer task" after training. None of the learners were able to type, revise, and print a simple one page letter without some trouble with each of these basic skills.

What is more surprising perhaps is that even when learners were able to successfully follow instruction sequences out, they still seemed to experience a loss of task orientation, as evidenced by comments like: "What did we do?", "I know I did something, but I don't know what it is!" or "I'm getting confused because I'm not actually doing anything except following these directions." For these subjects, the overall orientation toward accomplishing meaningful tasks (e.g., type a letter, print something out) has been subverted by a narrower orientation toward following out a sequence of instructions.

Learning by thinking.

Just as learners take the initiative to try things on their own, so also are they active in trying to make sense of their experience with the word processor. Learning passively by rote assimilation of information is atypical. Rather, learners actively try to develop hypotheses about why it operates the way it does. These quests after meaning can be triggered by new and salient facts. They can be forced by discrepancies between what is expected and what actually happens. They can be structured by the learner's personal agenda of goals and queries, referred to as new problems arise. In each case, learners' lack of knowledge about word processing makes it difficult for them to reason out coherent solutions that accurately represent the objective operation of the system.

For example, learners have no basis for recognizing and ruling out irrelevant connections; their interpretations of word processing systems are often influenced by spurious connections between what they think they need and what they perceive. In one case, a learner tried to decide if a "File" command had stored a document file away. It was not stored because the command was entered in a text input mode where all typed strings are interpreted as text, and not executed as commands. But she assumed that the file had been stored, and adduced evidence to confirm this premise. For example, at one point she notices a status message "INPUT MODE 1 FILE" which indicates that she is in the text input mode. However, the word "file" matched her file command, and this was enough to suggest some kind of feedback that her "File" (as in store document) command had worked.

In such cases, reasoning appears to consist in adducing factual support to a premise the learner would like to hold as true. The learner above began with the hypothesis that she had stored the document file away, and sought evidence to confirm that this was the case. Her adduction here was incorrect because she did not know which facts were relevant to verifying the premise. In other cases, reasoning appears to consist in abducing a hypothesis when it, together with other assumptions the learner may already hold, is consistent with some fact or observation. One learner tried to move the cursor in a protected area of the display. When this locked the keyboard, she hypothesized that this fact meant that she was at the right place on the screen to do what she set out to do.

Learners also set goals which they actively pursue by trying to solve problems. They are hampered in this by their innocence of the appropriate problem space, or domain of possible actions and interpretations relevant to accomplishing goals and addressing queries. Accordingly, their strategies are often local and fragmentary; they have difficulty integrating information or other experiences, and in formulating their concerns in ways that map transparently onto system functions. When learners cannot solve problems or answer questions, they add them to a personal agenda of goals and queries as they go along. As new opportunities arise, learners return to these standing queries and try to resolve them.

Learning by knowing.

To this point, we have argued that a new user of a word processing system relies on active exploration and ad hoc reasoning as learning strategies. However, not all possibilities are explored and not all hypotheses that could be reached are reached. What constrains these strategies is a sense of what could be appropriate -- and this devolves from prior knowledge on the part of the learner: knowledge about devices "like" word processors (e.g., typewriters), knowledge about office routine and work in general, even knowledge culled from interacting with the word processor up to that point in time.

Our learners were unable to resist referring to their prior knowledge about typewriters as a basis for interpreting and predicting experience with word processors. One came to a halt as she read an instruction in the manual which said "Backspace to erase." It seemed that she could not interpret this instruction for, as she pointed out, BACKSPACE does not erase anything. She had irresistibly availed herself of her knowledge of how backspacing works on a typewriter, unable to even consider that this knowledge might be inappropriate for the present case. Other learners tried to use SPACE and RETURN keys to move the cursor -- which insert spaces and blank lines -- but merely move the typing point on a typewriter.

Our learners were experienced with conventional office work: typing letters, filing, etc. Their knowledge about how these routine tasks are organized in the office creates expectations in them about how analogous tasks ought to be performed in the "office of the future" (as represented by the word processor in our laboratory). Thus, one response to revising a letter task is to retype. This is striking since it is the capability of the word processor to store and retrieve documents -- for revision, among other things -- that is its fundamental advance over previous office technologies.

As a learning experience progresses, the learner is acquiring and organizing new bits of knowledge. The ultimate goal -- and the final measure of success in the learning situation -- is that of assembling these pieces into a coherent fabric, an understanding of the word processor. Along the way, any prior bit of knowledge is available for use as a basis for expectations concerning successive interactions with the system. One system we studied seemed to flaunt inconsistency in similar operations. Thus, to delete a word, one positions the cursor under the word's initial character and keypresses WORD DELETE. However, to underscore a word, one positions the cursor under the final character of a word and keypresses WORD UND. This inconsistency caused one learner to misexecute one and then the other of these two operations in a dismal cycle of negative transfer.

Summary.

Perhaps the most apt discussion of the world of the new user of a word processing system is that often quoted phrase of William James: "a bloomin' buzzin' confusion" People in this situation see many things going on, but they do not know which of these are relevant to their current concerns. Indeed, they do not know if their current concerns are the appropriate concerns for them to have. The learner reads something in the manual; sees something on the display; and must try to connect the two, to integrate, to interpret. It would be unsurprising to find that people in such a situation suffer conceptual -- or even physical -- paralysis. They have so little basis on which to act.

And yet people do act. Indeed, perhaps the most pervasive tendency we have observed is that people simply strike out into the unknown. If the rich and diverse sources of available information cannot be interpreted, then some of these will be ignored. If something can be interpreted (no matter how specious the basis for this interpretation), then it will be interpreted. Ad hoc theories are hastily assembled out these odds and ends of partially relevant and partially extraneous generalization. And these "theories" are used for further prediction. Whatever initial confusions get into such a process, it is easy to see that they are at the mercy of an at least partially negative feedback loop: things quite often get worse before they get better.

What's wrong? We would argue that the learning practices people adopt here are typical, and in many situations adaptive. The problem in this particular learning situation is that new learners of word processors are innocent in the extreme. "Word processor", so far as we know, is not a natural concept. People who do not know about word processors have little, possibly nothing, to refer to in trying to actively learn to use such things. Innocence turns reasonable learning strategies into learning problems.

EXAMPLES IN THE LEGAL DOMAIN: HYPOTHETICALS IN CONTRACT LAW

Edwina L. Rissland*
Department of Computer and Information Science
University of Massachusetts
Amherst, MA 01003

Abstract

In this paper, we discuss the use of examples in the law, in particular "hypotheticals" in contract law. We present a framework for representing examples, show how this can be used to generate new hypotheticals, and discuss their role in the dialectic of refining or learning legal doctrine.

1. Introduction

Examples are important in many disciplines like mathematics, law and linguistics. They are central to reasoning and learning processes such as induction, concept formation, rule refinement and theory formation [Hawkins 1980; Kuhn 1970; Lakatos 1976; Lenat 1977; Polya 1965; 1968; Rissland 1978, 1982; Soloway 1978; Winston 1975].

In the law, where much reasoning is done by example [Levi 1949] and analogy [Berman 1968], examples -- i.e., cases -- are indispensable. Examples force one to consider possibilities and nuances. In teaching a legal doctrine, they are used to point out its "gaps, conflicts and ambiguities" [Kennedy 1980]. They are used in restatements of the law, which are compendia of legal doctrine in the form of principles, examples and references, e.g., Restatement, Second, Contracts [1981]. They are critical to the "realist number" which shows both that the law is much more than a set of clearcut concepts and rules [Llewellyn 1931], as the formalists of this century and before had hoped.

2. Epistemological Considerations

The examples in the law that we consider are of two types: (1) "real" cases, i.e., cases actually litigated; and (2) "hypothetical" cases ("hypotheticals" or "hypos"). Both types can be represented by a frame-like data structure [Minsky 1975] and the frames can be linked together by various types of relations. In describing frames for cases, we are laying out a conceptual framework to represent the knowledge used by students and teachers of the law.

The frame for a real case includes the following slots: Title, Citation, Date, Fact Situation, Process History/Outcomes, Arguments, Opinions, Links to other cases, Links to legal doctrine/rules/statutes. A slot can have a simple filler, as in the Title, Citation or Date slots, or a complex one as in the Opinions which can be structured into main, concurring, and dissenting opinions. Links to other cases include "procedural history" links, like affirmed, reversed, amended, and "substance" links, like criticised, distinguished, explained, harmonized, etc., which describe how the courts through their opinions related the cases.

Hypotheticals can also be represented by a frame. The most important features of a hypo are the Fact Situation, the Arguments that interpret the fact situation with respect to particular legal doctrines, and the links to other hypos and real cases. Thus the frame for a hypo is like that for a real case. The links between a hypo and a real case include "abstracted from", "particularized from", "generalized from".

One can also make a taxonomy of cases in the law, much as in mathematics [Rissland 1978]. Such a taxonomy is not explored here, but the categories might include:

1. standard cases (typically found in the casebooks);
2. landmark cases that have far reaching effects;
3. first impression cases that bring up an issue for the first time;
4. counter cases that show the limits of or the invalidity of a rule or doctrine;
5. anomalous cases that don't seem to fit in.

While we have used some of the link types used in LEXIS [Sprowl 1976] and legal digests and case citators like Shepard's Citations, the framework and taxonomy we have described could be used to design a legal data base that reflects more of the structure of the law than those currently in use.

3. Hypotheticals in Contract Law

In contract law, one master question is "Which promises should the law enforce?", where enforcement means either making the promisor fulfill his promise to the promisee (i.e., "specific performance") or make the promisor pay "damages" to the promisee for his breach [Fuller and Eisenberg 1981; Knapp 1976].

There are several ways of dealing with this question. The "gift-consideration" distinction tries to relate enforceability with the "consideration" given by the promisee in return for the promise [Section 17, Restatement, Second, Contracts]:

*Supported in part by the National Science Foundation under grant IST-80-17343. Opinions expressed in this report are those of the author and do not necessarily reflect views of the U.S. Government.

"...the formation of a contract requires a bargain in which there is...consideration..."

Another approach is that of "reliance" in which the (typically injurious) reliance of the promisee upon the promise is highlighted [Section 90, Restatement, Second, Contracts]. A third is the use of "formalities" like the legal seal [Section 96]. Each of these ways of looking at the master question emphasizes different aspects of a promise and each has its own stengths, weaknesses, inconsistencies and ambiguities.

The following is a set of hypos (actually just the fact situations) typical of those used in law school to: (1) point out the gift-consideration distinction; (2) show doctrinal weaknesses and ambiguities; and (3) show possible conflicts between doctrines such as consideration and reliance. The hypos are really caricatures of the real case of Dougherty v. Salt, decided by the N. Y. Court of Appeals in 1919, which is a standard case in first year Contract Law (e.g., see [Fuller and Eisenberg 1981]).

In each of the hypos, one is to ask, "Is this promise enforceable?" In other words, if the promisor breaches, ought the promisee be awarded damages or performance?

Hypo1:
Facts: Aunt Tillie says, "Charlie, you are such a nice boy; I promise to give you $10,000."

Hypo2:
Facts: Same as Hypo1 with the addition that Charlie says, "Dear Aunt Tillie, I can't take something for nothing, let me give you my third grade painting."

Hypo3:
Facts: Same as Hypo2 except Charlie offers to mow Tillie's lawn.

Hypo4:
Facts: Same as Hypo2 except that Charlie's last name is Picasso.

Hypo5:
Facts: Same as Hypo1 with the addition Aunt Tillie's assets are in ruin and that keeping her promise to Nephew Charlie means her own children starve.

Hypo6:
Facts: Same as Hypo1 with the addition that Charlie makes an unreturnable deposit on a new car.

If one argues from the standpoint of consideration doctrine, Hypo1 is a paradigmatic example of a pure gift, "a gratuitous promise". which would not be enforceable. Hypo2 is an attempt to make Hypo1 look enforceable under consideration doctrine. Hypo3 is another attempt to alter Hypo1 into an enforceable promise. Hypo4 is used to point out that one is making value judgements on the consideration per contra the doctrine that one should not inquire into the adequacy of the consideration. Hypo5 introduces an emotional "heart rendering" aspect to show there are limits and exceptions to consideration doctrine, such as duress. Hypo6 introduces an element of reliance which leads to conflicting outcomes from reliance and consideration argumentation.

4. A Frame for Promise Hypos

In applying the framework of Section 2 for the domain of contract law, we used the following facets in the sub-frame for the fact situation of a "promise" case:

1. the status of the PROMISOR
2. the subject matter of the PROMISE
3. the status of the PROMISEE
4. the RETURN ACTION by the promisee
5. the RELATION between the promisor and promisee

The full frame of the case would also include:

1. ARGUMENTS for various outcomes of the hypo according to various doctrines;
2. further NOTES/DISCUSSION of the hypo, such as historical significance;
3. RELATIONS to other cases (real and hypothetical).

Each of these major sub-blocks has facets; those for the PROMISOR and PROMISEE are similar; those for PROMISE and RETURN somewhat so. The PROMISOR and PROMISEE can be further described by such attributes as PERSONAL STATUS, INTENTIONS and BARGAINING POWER, which can be further broken down. For instance, PERSONAL STATUS includes SEX, AGE, MARITAL STATUS (these are for largely traditional, historical and common law reasons related to the once unequal status of women under the law).

The description of the PROMISE includes the subject matter of the promise and conditions on it. The RETURN action of the promisee can be: (1) no action; (2) forbearance (i.e., refraining from doing an act, like suing); (3) an action. An action itself has aspects like: (1) the action benefits or does not benefit the promisor; (2) the action leaves the promisor worse off/better off/the same.

One can also structure the RELATION facet of the promise situation for instance according to whether it is familial (e.g., father-daughter) or non-familial (e.g., debtor-creditor, friends or neighbors).

The following is a fact situation sub-frame instantiated for the first Aunt Tillie - Nephew Charlie hypo:

```
PROMISOR: Aunt Tillie
   PERSONAL STATUS: female, elderly, widow
   PERSONAL ATTRIBUTES: kind, rich
   INTENTIONS: the best
PROMISE: $10,000
   CONDITIONS: none
PROMISEE: Charlie
   STATUS: male, young
RETURN: none
RELATION:
   FAMILIAL: Aunt-Nephew
```

5. Generating Hypotheticals

It is apparent that one can generate new hypotheticals -- that is their frames -- by changing slot fillers in a hypothetical frame. Since the possible fillers for a slot can often be arranged in hierarchies, many modifications can be described in terms of super, sub and sibling node substitution and thus lead to modifications affecting generality and specifity. For instance, generalizing Tillie and Charlie to abstract individuals A and B results in the following:

Hypo1': A promises B $10,000.

Making another change gives:

Hypo1": "JR" promises B $10,000.

In the last, knowing that "JR" (as in Ewing) often has bad intentions creates a hypo very different in "feeling" from the "Aunt Tille - Nephew Charlie" or "A promises B" hypos; the "JR" hypo introduces questions of "good/bad faith".

Elaborating the description of any of the elements of the fact situation is another way a creating a new hypo. For instance, elaborating "Aunt Tille" to "old, senile Aunt Tillie" and "Charlie" to "manipulative, black-sheep-of-the-family Charlie" gives a very different character to the hypo.

6. Computer-generated hypos

We are currently investigating the generation of hypotheticals using the CEG (Constrained Example Generation) method of "retrieval plus modification". in which a new example is generated by retrieving a known example (that comes close to what is wanted) and then modifying it to meet the current requirements [Rissland and Soloway 1980, Rissland 1982]. So far, we have been dealing only with constraints such as "more/less general/specific" "different but of the same class" (e.g., familial). Higher level constraints are "heart rendering", "more/less surprising" (e.g., against one's default assumptions).

We are experimenting with ways to generate three or four sentence long hypos similar to those found as exercises in casebooks and as illustrations in the Restatements. To produce the English text from the frame, we are currently using sterotypical precanned text templates and then filling in the templates with information from the hypo frame. An example of such a template filled in the most general way is:

" A promises B X in return for Y ."

More sophisticated -- longer and subtler -- hypos will need more sophisticated text generation such as McDonald's MUMBLE [McDonald 1981].

7. Summary and Conclusions

We have been studying the structure of legal knowledge, specifically real and hypothetical cases, using a structural approach of frames and relations and how one generates hypotheticals; we have actually experimented with our ideas in the domain of Contract Law, we feel that these methods are easily transferable to other domains such as Property and Torts.

We feel our work contributes to: (1) a better understanding of the use, structure and generation of examples in general and legal hypotheticals in particular; (2) epistemological analysis of legal domains; (3) legal data base design; (4) hypothetical generation for teaching and ICAI (Intelligent Computer Assisted Instruction) systems.

8. References

Berman, H. J., "Legal Reasoning". In International Encyclopedia of the Social Sciences.

Fuller, L. L., and M. A. Eisenberg, Basic Contract Law. West Publishing Co., Minn., 1981.

Hawkins, D., "The View from Below". For the Learning of Mathematics. Volume 1, No. 2, FLM Publishing Association, Quebec, Canada, November 1980.

Kennedy, D., "Utopian Proposal". Draft memo, Harvard Law School, 1980.

Knapp, C. L., Problems in Contract Law. Little, Brown and Co., 1976.

Kuhn, T. S., The Structure of Scientific Revolutions. Second Edition. University of Chicago Press, 1970.

Lakatos, I., Proofs and Refutations. Cambridge University Press, London, 1976.

Lenat, D. B., "Automatic Theory Formation in Mathematics". Proc. IJCAI-77.

Levi, E. H., An Introduction to Legal Reasoning. University of Chicago Press, 1949.

McDonald, D. D., "Language Production: The source of the dictionary." In The Nineteenth Annual Meeting of the Association for Computational Lingistics, Stanford University, 1981.

Minsky, M. L., "A Framework for Representing Knowledge". In The Pyschology of Computer Vision, Winston (ed), McGraw-Hill, 1975.

Polya, G., Mathematical Discovery. Volume II. Wiley, New York, 1965.

Polya, G., Mathematics and Plausible Reasoning, Volumes I and II. Princeton University Press, 1968.

Restatement, Second, Contracts. American Legal Institute, Philadelphia, 1981.

Rissland, E. L., "Constrained Example Generation". Submitted for publication, 1982.

Rissland, E. L., "Understanding Understanding Mathematics". Cognitive Science, Vol. 2, No. 4, 1978.

Rissland, E. L., and E. M. Soloway, "Overview of an Example Generation System". In Proc. First National Conference on Artificial Intelligence. Stanford, August 1980.

Soloway, E. M., "Learning = Interpretation + Generalization: A Case Study in Knowledge-Directed Learning". COINS Technical Report 78-13, University of Massachusetts, 1978.

Sprowl, J. A., A Manual for Computer-Assisted Legal Research. American Bar Foundation, Chicago, 1976.

Winston, P. H., "Learning Structural Descriptions from Examples" in The Psychology of Computer Vision, Winston (ed), McGraw-Hill, 1975.

Learning Recursive Procedures by Middleschool Children[1]

Yuichiro Anzai
Carnegie-Mellon University & Keio University

and

Yuzuru Uesato
Keio University

Introduction

Recursion is a recurrent theme in human thinking. It has been around for a long time in some fields related to cognitive science: for instance, it has taken place in information-processing models of cognition, in the theory of computation, in cognitive and developmental psychology, or in teaching computer programming to novices.

Intuitively, recursive formulation may lead to understanding of potentially infinite phenomena in compact, finite terms. On the other hand, since recursive definition involves top-down, tightly connected organization of knowledge, it may not be easy to learn, or to be applied to formulation of complex problems. These expectations, however, are less well examined experimentally. Besides, there are some other points such as memory load for executing recursive procedures, the firmly established character of recursive functions in the theory of mathematics, or practical application to teaching computer programming, which make recursion an interesting theme for cognitive science. As one topic related to recursion, this paper discusses the question of whether recursive procedures are cognitively difficult to learn, based on a rule induction experiment conducted on middleschool children. It concludes that recursive procedures may be acquired based on learning of the corresponding iterative procedures.

Learning Recursive Procedures

A recursive function treated here is simply a function whose definition includes the function itself. As a simple but representative example, we use exclusively in this paper the factorial function "*fact*" defined on *N*, the set of positive integers, as follows:

$$fact(n) = fact(n\text{-}1) \times n \text{ for any } n \in N,\ n>1, \quad \text{and } fact(1) = 1.$$

The above definition is recursive, but of course *fact* can be defined iteratively:

$$fact(n) = 1 \times 2 \times \ldots \times n \quad \text{for any } n \in N.$$

The above two kinds of definitions are functionally equivalent, but have many cognitively different points. Let us consider below only the point relevant here: how people acquire the recursive procedure for computing factorials, based on example data. First, suppose that a student is given an iterative sequence of data for factorials:

$$fact(1) = 1 \quad fact(2) = 1 \times 2 \quad fact(3) = 1 \times 2 \times 3.$$

It may be easy for him to generalize the above simple patterned sequence, and to obtain the general iterative definition, $fact(n) = 1 \times 2 \times \ldots \times n$ ($n \in N$). Note that the induced definition itself can easily be interpreted to provide procedures (multiplications) for actual computation.

On the other hand, suppose that the student tries to induce the factorial function based on the following recursively generated data:

$$fact(3) = fact(2) \times 3 \quad fact(2) = fact(1) \times 2 \quad fact(1) = 1.$$

In this case, although the data, if regarded declarative, can be generalized formally to generate $fact(n) = fact(n\text{-}1) \times n$ ($n \in N$), the student needs to consider all the subformulas, $fact(k) = fact(k\text{-}1) \times k$ ($k = 2,\ldots,n\text{-}1$), to actually compute $fact(n)$: the data allow direct generalization by converting, for example, 3 to n and 2 to n-1, but he is necessary to organize the given segments of data to acquire the recursive computational procedure. It may be much more difficult than in the iterative case.

However, we can advance our speculation one more step. The student, while he is engaged in the task of inducing the factorial from the iterative data, might notice the regularity of embedded pattern in the data. The left column of Fig. 1 illustrates it for an iterative data set. If this kind of structural embedding was discovered, acquisition of the iterative definition of the factorial may result in learning the nested procedural structure of the factorial. Then, if the nested structure as shown in the left of Fig. 1 resides in memory, and if recursive data are presented, the data may match the nested structure fairly easily as shown in Fig. 1. Thus, the recursive procedure may be learned by the successive presentation of the iterative and recursive data sets in this order.

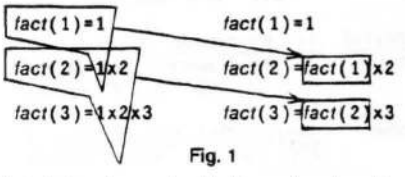


Fig. 1

Nested structure embedded in an iterative data set and its relation to the corresponding recursive data set

The preceding simple discussion gives us the hypothesis that, if a student knew none of the factorial function, or the concept of recursion, he finds it easier to learn the iterative procedure for the factorial rather than the recursive one, but after he learned it, he must already be ready to assimilate the recursive procedure. In the following rule induction experiment, we examine this hypothesis by using middleschool children.

Experiment

Subjects and procedure

88 middleschool children (age about 14) participated in the experiment. The rule to be induced was the numerical function for computing factorials of positive integers. Two kinds of formats for example data were considered. One was the iterative format, and the corresponding to-be-induced function was called WHITE in the experiment. The other was the recursive format, and the corresponding function was named BLACK.

For each format, a sequence of three data sets was prepared. The first data sets for WHITE and BLACK were given as follows:

First data set for WHITE

Let us think about the following computation for a given number. The answer to the computation is called "WHITE" For example, "WHITE of 2" is computed as follows:

(1) Start with 1.

(2) Multiply 1 by 2. The result is 2.

[1]Thanks are due to John Anderson, Robin Jeffries and Herbert Simon for their comments on this work. Please address correspondence to Yuichiro Anzai, Faculty of Science and Technology, Keio University, 3-14-1, Hiyoshi, Kohoku, Yokohama, Japan.

"WHITE of 2" is 2.

Now, compute "WHITE of 4". (Write the computation and the answer.)

First data set for BLACK

Let us think about the following computation for a given number. The answer to the computation is called "BLACK". For example, "BLACK of 2" is computed as follows:

(1) "BLACK of 2" is "BLACK of 1" multiplied by 2.

(2) "BLACK of 1" is 1.

"BLACK of 2" is 2.

Now, compute "BLACK of 4". (Write the computation and the answer.)

In each of the above data sets, two segments of information, *(1)* and (2), for the factorial of 2, and the value of it were supplied. There was provided a problem at the last line, which was to compute the factorial of 4. If a subject gave the correct answer to the problem, then he was considered to have acquired a factorial-computing procedure, iterative or recursive, depending on which data set, WHITE or BLACK, was presented to him.

The second data sets for WHITE and BLACK included three segments of information, and were designed as shown below:

Second data set for WHITE

Let us think about the following computation for a given number. The answer to the computation is called "WHITE". For example, "WHITE of 3" is computed as follows:

(1) Start with 1.

(2) Multiply 1 by 2. The result is 2.

(3) Multiply 2 by 3. The result is 6.

"WHITE of 3" is 6.

Now, compute "WHITE of 5". (Write the computation and the answer.)

Second data set for BLACK

Let us think about the following computation for a given number. The answer to the computation is called "BLACK". For example, "BLACK of 3" is computed as follows:

(1) "BLACK of 3" is "BLACK of 2" multiplied by 3.

(2) "BLACK of 2" is "BLACK of 1" multiplied by 2.

(3) "BLACK of 1" is 1.

"BLACK of 3" is 6.

Now, compute "BLACK of 5" (Write the computation and the answer.)

The third data sets, each of which contained four segments of information, were defined in a similar manner.

The subjects were divided into two groups called G1 ($n = 45$) and G2 ($n = 43$). The group G1 was given the data in the order of W-1, W-2, W-3, B-1, B-2 and B-3, where W-*i* and B-*i* denote the *i*-th data set for WHITE and BLACK respectively. On the other hand, G2 was given the data in the order of B-1, B-2, B-3, W-1, W-2 and W-3. Both groups were given five minutes for each data set, which were ample enough for middleschool children. The data sheets were collected from the subjects for each data set, and no direct feedback of answers was given.

Results and discussion

The results are tabulated in Table 1. The more data sets presented, the greater number of subjects who answered correctly, both for WHITE and BLACK. The percent correct was larger for G1's WHITE (60% for the third set) than for G2's BLACK (33% for the third set), but even the latter gave fairly good performance. Also, if the data for BLACK were presented after WHITE as for the group G1, the performance was better than its opposite: G1 for BLACK gave 16%, 29% and 64% of percent correct for the data sets with two, three and four segments of information, but G2 for BLACK provided 0%, 14% and 33%, which were relatively smaller. On the other hand, the performace for WHITE was similar for the two groups, regardless of the order of presentation.

The result is thus generally in agree with our expectations. It was easier for the children to have worked on the iteratively generated data sets, but acquisition of the recursive procedure was facilitated by learning the iterative one.

Also, note that the WHITE data for G1 and G2 show a similar tendency, and the BLACK data for the two groups provide a *different sort of* similar tendency: the rate of increase of the percent correct decreased for the WHITE data with respect to the number of presented data sets, while it increased for the BLACK data. This particular trend may have reflected the subjects' relative difficulty in discovering regularity in a small number of information segments in a recursive data set.

Table 1

Percent correct for the induction experiment

	G1		G2	
Data set no.	WHITE	BLACK	BLACK	WHITE
1	11(%)	16	0	9
2	42	29	14	30
3	60	64	33	47
No. of subjects	45		43	

(For almost all the subjects, if a subject gave the correct answer for the *j*-th data set, he was also correct for all the *i*-th sets, where $1 \leq i < j$.)

Thus, we think that recursive computation may be apparently difficult for children to learn, but also that it may be acquired by inducing the nested structure, and interpreting it as a procedure, based on the recursive data. Let us provide one possible mechanism that generates the gross characteristics of the experimental results, which is essentially similar to the one briefly described in the previous section. Suppose that the third data sets for WHITE and BLACK given in the experiment were represented as follows:

WHITE	BLACK
(equal (times 1 2) 2)	(equal (black 4) (times (black 3) 4))
(equal (times 2 3) 6)	(equal (black 3) (times (black 2) 3))
(equal (times 6 4) 24)	(equal (black 2) (times (black 1) 2))
(equal (white 4) 24)	(equal (black 1) 1).

Assume that, successively embedding the segments in the WHITE data set, we obtained the nested formula:

(equal (white 4) (times (times (times 1 2) 3) 4)).

Note that, if we identify (times (times 1 2) 3) with (black 3), and also identify "white" with "black", then the formula matches the first segment in the above BLACK set:

(equal (black 4) (times (black 3) 4)).

This kind of correspondence holds also for the first and second data sets. Generalization at this point, which yields the correspondence between (times (times (... (times (times 1 2) 3) ...) *n*-1) *n*) and (black *n*), provides the procedural basis for the recursive definition of the factorial function, which is based on nested arithmetic calculation.

Discussion

The relation between conceptual and procedural understanding in problem solving has raised many issues complex but central for cognitive science. At some deeper level of understanding, a person can both handle with knowledge procedurally, and appreciate it declaratively. Recursion provides a simple example for this matter: since it is usually formulated in a compact form, its declarative representation may be simpler than the corresponding iterative form. But such declarative representation must be accompanied by procedural knowledge for actual computation, and this knowledge might be cognitively complex. The argument presented in this paper suggests that such knowledge can be acquired not directly, but by working on iterative data.

An example of the process of learning a recursive strategy by discovering a nested structure in knowledge of results obtained by weaker, nonrecursive strategies was presented in Anzai & Simon (1979). The strategy acquisition process reported there is essentially similar to the recursion learning process discussed in this paper: the thesis shared by the two studies is that complex recursive procedures for solving a problem may be acquired by working on the problem, using already available, nonrecursive knowledge.

Which way of learning, by discovery or by instruction, is better has long been a controversial problem in instructional psychology. Learning by doing, which is along the line discussed here and in Anzai & Simon, is basically a process of learning by discovery. In this regard, as suggested in this paper, recursive procedures may be learned by discovery. Recursive computation may be intrinsically more difficult than iterative one, since execution of recursive procedures may require more memory resources. But it does not mean that they can not be acquired by discovery.

However, of course we do not deny the possibility of learning recursive procedures by top-down instruction. The two ways of learning are actually complementary in the real world, and both ways may play important and intertwined roles. Also, we should be cautious when we try to extend the consideration to more complex domains such as computer programming. It is because a complex task necessarily involves many different cognitive subprocesses, and it is not always easy to extract from them only the part played by recursion.

References

Anzai, Y. & Simon, H. A. 1979 The theory of learning by doing. *Psychological Review*, **86**, 124-140.

Prior Knowledge Occupies Cognitive Capacity in Chess Problem Solving, Reading, and Thinking

By Bruce K. Britton and Abraham Tesser

Abstract

Prior knowledge was varied in problem solving, thinking, and reading tasks in three experiments. The hypothesis was that the prior knowledge used in a cognitive task uses capacity in the same limited capacity active processing system that is used to process the ongoing task. In a reading experiment, prior knowledge about a target page was manipulated by controlling the preceding pages. In an experiment dealing with problem solving in the context of a chess game, prior knowledge was controlled by comparing experts with novices. In a third study subjects thought about personality descriptions of persons and groups, and about women's fashions and football plays; it was assumed that persons have more prior knowledge concerning the personality of persons than the personality of groups, that women have more prior knowledge about women's fashions, and that men have more prior knowledge about football. In all experiments, use of cognitive capacity in task performance was observed with a secondary task technique.

The results of all three experiments were consistent with the hypothesis that prior knowledge uses capacity in the active processing system. The prior knowledge hypothesis is consistent with some aspects of current cognitive theory but not consistent with others. The results also suggest a fundamental and unexpected limit on the cognitive processing of experts.

Information processing theories of cognitive processing often assume that memories of prior experience are stored over the long term in a relatively inactive state. They also assume that the cognitive task that is undergoing processing at a particular time is being processed in an active processing system, which some models identify as a working memory or short term memory store. When stored prior knowledge is to be used in the performance of a particular cognitive task, the prior knowledge is brought from the inactive state into an active state. In this active state the prior knowledge can be effectively used in performing the ongoing cognitive task.

In the standard model (e.g., Atkinson & Shiffrin, 1968) this change of state of prior knowledge is usually represented in a flow chart as an arrow leading from a long term memory store (the inactive memory) to a short term or working memory (the active processing system). Other models of cognitive processing include a similar assumption; although the metaphor of a spatial transfer of information is not always used, some change in the state of activation of the prior knowledge is expressed with other metaphors.

The active processing system is widely believed to be limited in capacity (Broadbent, 1958, 1971; Navon & Gopher, 1979; Norman & Bobrow, 1975; Posner, 1978). If the active system is limited in capacity, then it is plausible to deduce that any prior knowledge that is active in it will use some of the limited capacity. This paper reports three tests of the hypothesis that the prior knowledge used in an ongoing task uses cognitive capacity in the same active processing system that is used to perform the ongoing task. This will be referred to as the prior knowledge hypothesis.

The prior knowledge hypothesis has not been included conventionally among the explicit assumptions of cognitive processing models. Perhaps this is because the standard model and related models have traditionally assumed a small limit on the capacity of short term memory, with estimates ranging from 2 chunks up to 20 (Lachman, Lachman & Butterfield, 1978). It appears that with even a 20 unit limit, a body of prior knowledge of a size or complexity that approached that limit -- for example, the chess knowledge of an expert chess player -- if transferred to a short term store, would occupy so much of it that little or no capacity would be left over for performing the ongoing cognitive task. The result would be error, delay or failure on the task. Cognitive psychologists may have believed that this outcome did not seem likely to occur, and so the prior knowledge hypothesis may not have seemed easily compatible with models that include a small limit on the capacity of the active processing system. Other cognitive models are less explicit about the capacity of the active processing system, so evidence that large bodies of activated prior knowledge use capacity would be less critical for them.

Because the hypothesis that prior knowledge uses capacity in the active processing system has not been prominent in cognitive theory, the consequences of it have not been thoroughly worked out, and some of them turn out to be interesting. One set of consequences is related to the use of cognitive capacity by persons who do or do not have prior knowledge about a particular cognitive task, i.e., experts and novices. The cognitive programs of experts and novices have been investigated by protocol analysis techniques (e.g., Ericsson & Simon, 1980), but these techniques do not provide data on capacity usage. In the present experiments the secondary task technique was used. This technique was designed to provide data on capacity usage. The prediction of the prior knowledge hypothesis is that experts will use more capacity than novices when they are performing cognitive tasks for which the experts have activated large amounts of prior knowledge. Apparently this prediction has not been tested previously. To test this prediction of the prior knowledge hypothesis, in two of the experiments reported here, 'experts' on chess, and on football, women's fashions and implicit personality theory were observed as they processed problems in their special topics and in topics in which they were not experts. Use of cognitive capacity was measured with a secondary task technique. In a third experiment, differences in prior knowledge about a text topic were induced in readers and the use of capacity was observed in reading later parts of the text.

Another interesting consequence of the prior knowledge hypothesis is that it suggests the existence of a potential limitation on the cognitive processing of experts. If an expert has an extremely large amount of activated prior knowledge for a particular task, the knowledge will presumably use a correspondingly large amount of capacity. If the prior knowledge uses enough capacity, the capacity available for the ongoing cognitive task will be reduced: this follows from the assumption of a limited capacity. A straightforward prediction is that the ongoing task will be performed more slowly by such an expert with a very large amount of prior knowledge than by a person with less prior knowledge (assuming the prior knowledge is adequate to perform the task). In extreme cases of prior knowledge, so much active capacity may be occupied that the expert may not be able to complete the cognitive task at all. Such an hypothesis could be used to account for: (1) the long periods of time taken by extremely know-

ledgeable experts to solve problems that are solvable in less time by somewhat less knowledgeable experts, (2) the decreases in scholarly productivity that are sometimes reported anecdotally when scholars reach extremely high levels of expert knowledge about their special subject, (3) the incubation effect in problem solving, in which problem solvers who take time off from a thoroughly studied problem, presumably allowing some prior knowledge to be deactivated, report that when they return to the problem, they have an increased chance of solution, (4) the reduction of usable cognitive capacity that may be associated with aging individuals, who presumably have large amounts of prior knowledge. A possible qualification of this extension of the prior knowledge hypothesis is that experts seem likely to be able to chunk their knowledge more efficiently than novices, and chunks would presumably occupy less capacity. But in a very high level chunk, the usable information may not be visible on the surface. In order to reach a level of information that actually can be used in the performance of the ongoing task, the chunk may have to be unpacked to the point where usable information is revealed (Estes, 1972; Johnson, 1972). The unpacking process may use additional capacity that the less expert can avoid. It should be noted that such extreme cases of prior knowledge were not included in the present studies. The levels of prior knowledge used in the present studies may be regarded as intermediate in size between the levels of novices and those of high level experts, and decreases in performance of the ongoing task were not expected.

It is well to state at the outset what conclusions can be drawn from the various possible outcomes of the tests proposed here. If the prior knowledge is not shown to use capacity, that is consistent with the hypothesis that the cognitive task is performed in one active system, and the prior knowledge is active in a quite different system that does not share capacity with the first. If prior knowledge is shown to use capacity, that is consistent with the hypothesis that both the cognitive task and the prior knowledge are using capacity in the same active processing system.

The results of all three experiments were that subjects took longer to react to secondary task probes in the high prior knowledge conditions. Thus, the results of these experiments were all consistent with the hypothesis that the prior knowledge that is used in an ongoing cognitive task occupies capacity in the same limited capacity system that is used to perform the cognitive task. There are several aspects of the cognitive handling of prior knowledge that may make use of capacity. First, the retrieval of the bodies of knowledge from inactive memory may use capacity. The retrieval process presumably includes both search and decision components. Such a retrieval process may only occur once, at the beginning of the involvement of prior knowledge in the ongoing task, or it may be going on more or less continuously during performance of the task. Multiple retrievals would use capacity over a longer span of time than would a single retrieval episode.

Second, once a particular body of knowledge has been confidently located, its change of state from an inactive to an active status may use capacity. Third, once that activation has occurred, the maintenance of the activated state may be necessary, at least if the active state has rapid decay properties like those of conventional short term stores. The maintenance may be continuous, it may be periodic, as if the activation is regularly 'refreshed,' or it may be intermittent and dependent on the time course of use of the knowledge in the task. Fourth, the elements of the activated body of knowledge themselves are likely to occupy capacity, and the more extensive the knowledge is, the more elements it has, and the more capacity it can be expected to occupy.

Finally, the use of prior knowledge in the performance of the cognitive task may require additional cognitive operations that use capacity. These may involve the unpacking of chunks, searches through them, and decision processes associated with their use in the ongoing task. Or the prior knowledge may be in the form of programs of cognitive operations that are to be carried out as part of the cognitive task. Such programs enable additional operations, and these may use capacity.

The results reported here clarify the interpretation of some previous research on the use of cognitive capacity in reading. In a series of investigations of the influence of text characteristics on the use of cognitive capacity in reading, it was found that easy passages used more capacity than difficult ones (Britton, Westbrook, & Holdredge, 1978), where ease and difficulty were defined by cloze tests and ratings. This finding has been replicated (Britton, 1980; Britton, Zeigler. & Westbrook, 1980). It has been pointed out by Anderson and Armbuster (in press) that the easy passages used in those studies were about topics for which readers are "more apt to have available schemata or perspectives . . . than are those from the difficult passages." (p. 15). This interpretation is similar to the notion, based on the present results, that the readers had prior knowledge about the easy passages. The results of Britton, Graesser, Glynn, Hamilton, and Penland (in press) on genre differences can be interpreted along the same lines, as can the results of Britton, Westbrook, Holdredge and Curry (1979) that passages with more discourse level meaning (but identical to passages with less discourse level meaning) used more capacity.

Some limitations of these conclusions should be noted. First, they may only apply to complex bodies of prior knowledge, and probably not to isolated individual units. For such units, the retrieval, activation, maintenance and use of the knowledge may require so few cognitive operations that no observable capacity is used. Also, if the use of the prior knowledge is very highly practiced it may use less capacity (Shiffrin & Schneider, 1977; Schneider & Shiffrin, 1977).

Second, there appears to be a special case of combinations of prior knowledge and cognitive task for which prior knowledge will probably reduce use of capacity. These are tasks for which the completed solution of the task is already stored in memory and is easily accessible. For example, if the subject is asked to multiply 37 x 8, many mental operations will be carried out to arrive at the correct answer of 296. But if the subject is immediately asked again to multiply 37 x 8, the prior knowledge of the answer will be retrieved from memory, and the effect will be to reduce the number of mental operations and so the use of capacity.

Dynamic Construction of Finite Automata From Examples Using Hill-Climbing

Masaru Tomita

Computer Science Department
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

Abstract

The problem addressed in this paper is heuristically-guided learning of finite automata from examples. Given positive sample strings and negative sample strings, a finite automaton is generated and incrementally refined to accept all positive samples but no negative samples. This paper describes some experiments in applying hill-climbing to modify finite automata to accept a desired regular language. We show that many problems can be solved by this simple method.

1. Introduction

Consider the following problem:

Describe the property that all strings in the right-list have but no string in the wrong-list has. Does a string (1 1 0 1) have this property? You may answer the question by using any of the following: English, a regular expression, or a finite automaton.[1]

right-list	wrong-list
()	(1 0)
(1)	(1 0 1)
(0)	(0 1 0)
(0 1)	(1 0 1 0)
(1 1)	(1 1 1 0)
(0 0)	(1 0 1 1)
(1 0 0)	(1 0 0 0 1)
(1 1 0)	(1 1 1 0 1 0)
(1 1 1)	(1 0 0 1 0 0 0)
(0 0 0)	(1 1 1 1 1 0 0 0)
(1 0 0 1 0 0)	(0 1 1 1 0 0 1 1 0 1)
(1 1 0 0 0 0 0 1 1 1 0 0 0 0 1)	(1 1 0 1 1 1 0 0 1 1 0)
(1 1 1 1 0 1 1 0 0 0 1 0 0 1 1 1 0 0)	

It might be possible to construct the machine by a "typical" schema-filling method (i.e., finding rough property in the samples first, comparing these strings carefully). However, in this paper,

[1]The answer is strings over (1 + 0)* without odd number of consecutive 0's AFTER odd number of consecutive 1's. Therefore (1 1 0 1) has the property.

we try to construct the machine directly by searching in the problem space (i.e., a set of all finite automata) using hill-climbing, rather than by analyzing the samples carefully.

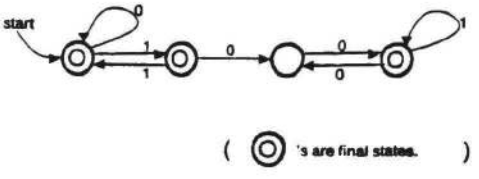
One of the biggest advantages of hill-climbing is its simplicity, that is, we do not have to know our problem space well, while a "typical" schema-filling method requires us to provide all possible schemas, and therefore to know everything about our problem space.

We shall see that hill-climbing works much better than expected in our problem space, and in fact solved most of the problems.

1.1. The finite automata used in this paper

We restrict our problem domain to be only over {1,0}*. Furthermore, since every non-deterministic finite automaton has an equivalent deterministic finite automaton (see [7]), we deal only with deterministic finite automata, that is, there is at most one 1-arrow and one 0-arrow from each state. Thus, in this paper, the terms "finite automaton", "automaton" or "machine" all mean "deterministic finite automaton". Given a string s, if there is a transition from the initial state to any of the final states, then s is accepted by the machine, otherwise s is rejected. For example, the machine of the sample problem is shown in figure 1.

Figure 1: The machine of the sample problem



Each machine with n states is denoted by the following form:

$((A_1, B_1, F_1)(A_2, B_2, F_2) \ldots (A_n, B_n, F_n))$.

Each (A_i, B_i, F_i) corresponds to the state i, and A_i and B_i indicate the destination states of the 0-arrow and the 1-arrow from the state i, respectively. If A_i or B_i is zero, then there is no 0-arrow or 1-arrow from the state i, respectively. F_i indicates whether state i is one of the final states or not. If F_i is equal to 1, the state i is one of the final states. The initial state is always state 1. For instance, figure 1 is represented as follows:

((1 2 1)(3 1 1)(4 0 0)(3 4 1)).

1.2 The problem

We now are ready to describe the problem precisely. Given a right-list (a set of positive sample strings) and a wrong-list (a set of negative sample strings), we can think of the following three tasks:

1. To find a machine that accepts all strings in the right-list but none in the wrong-list.
2. To find a machine with n states that accepts all strings in the right-list but none in the wrong-list.
3. To find the machine with fewest states (simplest machine) that accepts all strings in the right-list but none in the wrong-list.

The first task is trivial because one can easily construct a trivial machine that accepts exactly all strings in the right-list but nothing else. We

call the second task construction of finite automata, and the third task simplification of finite automata.

1.3. Sample Problems

Throughout this paper, we consider the particular seven problems shown in figure 2.

Figure 2: Sample Problems

Problem 1

right-list	wrong-list
( )	(0)
(1)	(1 0)
(1 1)	(0 1)
(1 1 1)	(0 0)
(1 1 1 1)	(0 1 1)
(1 1 1 1 1)	(1 1 0)
(1 1 1 1 1 1)	(1 1 1 1 1 1 1 0)
(1 1 1 1 1 1 1)	(1 0 1 1 1 1 1 1)
(1 1 1 1 1 1 1 1)	

Problem 2

right-list	wrong-list
( )	(1)
(1 0)	(0)
(1 0 1 0)	(1 1)
(1 0 1 0 1 0)	(0 0)
(1 0 1 0 1 0 1 0)	(0 1)
(1 0 1 0 1 0 1 0 1 0)	(1 0 1)
(1 0 1 0 1 0 1 0 1 0 1 0)	(1 0 0)
	(1 0 0 1 0 1 0)
	(1 0 1 1 0)
	(1 1 0 1 0 1 0 1 0)

Problem 3

[The problem we introduced at the beginning]

Problem 4

right-list	wrong-list
( )	(0 0 0)
(1)	(1 1 0 0 0)
(0)	(0 0 0 1)
(1 0)	(0 0 0 0 0 0 0 0 0)
(0 1)	(1 1 1 1 1 0 0 0 0 1 1)
(0 0)	(1 1 0 1 0 1 0 0 0 0 0 0 1 0 1 1 1)
(1 0 0 1 0 0)	(1 0 1 0 0 1 0 0 0 1)
(0 0 1 1 1 1 1 1 0 1 0 0)	(0 0 0 0)
(0 1 0 0 1 0 0 1 0 0)	(0 0 0 0 0)
(1 1 1 0 0)	
(0 1 0)	

Problem 5

right-list	wrong-list
( )	(1)
(1 1)	(0)
(0 0)	(1 1 1)
(1 0 0 1)	(0 1 0)
(0 1 0 1)	(0 0 0 0 0 0 0 0 0)
(1 0 1 0)	(1 0 0 0)
(1 0 0 0 1 1 1 1 1 0 1)	(0 1)
(1 0 0 1 1 0 0 0 0 1 1 1 1 0 1 0)	(1 0)
(1 1 1 1 1 1)	(1 1 1 0 0 1 0 1 0 0)
(0 0 0 0)	(0 1 0 1 1 1 1 1 1 1 1 0)
	(0 0 0 1)
	(0 1 1)

Problem 6

right-list	wrong-list
( )	(1)
(1 0)	(0)
(0 1)	(1 1)
(1 1 0 0)	(0 0)
(1 0 1 0 1 0)	(1 0 1)
(1 1 1 1)	(0 1 1)
(0 0 0 0 0 0 0 0)	(1 1 0 0 1)
(1 0 1 1 1 1)	(1 1 1 1 1)
(0 1 1 1 1 0 1 1 1 1)	(0 0 0 0 0 0 0 0 0)
(1 0 0 1 0 0 1 0 0)	(0 1 0 1 1 1)
	(1 0 1 1 1 1 0 1 1 1 1)
	(1 0 0 1 0 0 1 0 0 1)

Problem 7

right-list	wrong-list
( )	(1 0 1 0)
(1)	(0 0 1 1 0 0 1 1 0 0 0)
(0)	(0 1 0 1 0 1 0 1 0 1 0 1)
(1 0)	(1 0 1 1 1 0 1 0)
(0 1)	(1 0 1 0 1)
(1 1 1 1 1)	(0 1 0 1 0 0)
(0 0 0)	(1 0 1 0 0 1)
(0 0 1 1 0 0 1 1)	(1 0 0 1 0 0 1 1 0 1 0 1)
(0 1 0 1)	
(0 0 0 0 1 0 0 0 0 1 1 1 1)	
(0 0 1 0 0)	
(0 1 1 1 1 1 0 1 1 1 1 1)	
(0 0)	

The solution of these problems are:

1. 1*
2. (1 0)*
3. any string without an odd number of consecutive 0's AFTER an odd number of consecutive 1's.
4. any string without more than 2 consecutive 0's.
5. any string of even length which, making pairs, has an odd number of (0 1) or (1 0)'s.
6. any string such that the difference between the numbers of 1's and 0's is 3n.
7. 0*1*0*1*.

We also consider the inverse problem of those in figure 2. The inverse problems are created by exchanging the right-list and the wrong-list. We use these 14 problems in our experiments and refer to the inverse problem of problem i as i-.

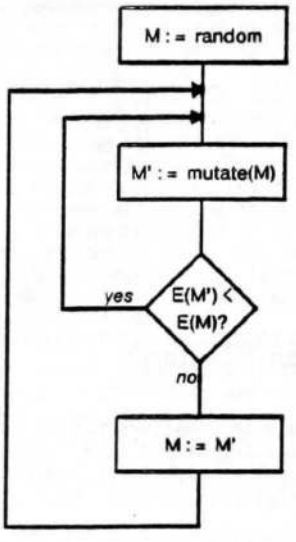
2. Construction of Finite Automata

In this section, we describe an experiment in constructing a finite automaton with n states from a given right-list and a wrong-list, using the hill-climbing. In particular, we let n equal 8. We shall see that each of the 14 problems can be solved in at most a few thousands steps.

2.1. Algorithm

The hill-climbing algorithm of this experiment is shown in figure 3.

Figure 3: Flowchart of the Hill-Climbing



We first construct a random machine with 8 states. We next make a copy of this machine, where the copy is slightly altered from the original by an operator mutate. We compare the new machine with the original by an evaluation function E. The better machine is called current generation and we make a copy of this machine, and so forth. The worse machine is simply discarded. The operator mutate and the evaluation function E are defined more precisely in the following.

Operator mutate: Taking a machine $((A_1,B_1,F_1) \ldots (A_8,B_8,F_8))$ as its argument, the operator mutate chooses one digit randomly, and replaces it by another digit.[2] That is, the mutation in our algorithm is randomly one of the following: delete an arrow, insert an arrow, change the destination of an arrow to another destination, make a non-final state a final state, and make a final state into a non-final state.

Evaluation Function E: The evaluation function E takes a machine as its argument and returns r - w, where r is the number of strings in the right-list accepted by the machine, and w is the number of strings in the wrong-list accepted by the machine. If r - w < 0 then it returns 0.

2.2. Result

We show in this section the result of our experiments. We first show in figure 4 the trace of the experiment of problem 3, to see how our algorithm gradually refines a random machine into the desired machine.

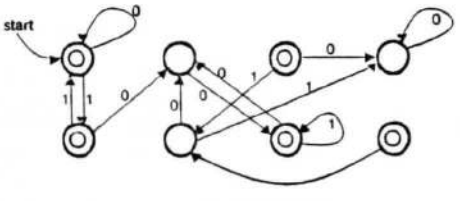
$$^{2}0 = < A_i = < 8; 0 = < B_i = < 8; \text{ and } 0 = < F_i = < 1.$$

Figure 4: Sample Trace of Problem 3

```
                                                                                E G
-------------------------------------------------------------------------------------
(((1 4 0)(6 2 1)(3 3 1)(1 0 0)(0 0 1)(4 8 0)(4 1 1)(3 2 1)) 0 1)
(((1 4 0)(6 2 1)(3 3 1)(1 0 0)(0 0 1)(4 8 1)(4 1 1)(3 2 1)) 0 2)
(((1 4 0)(6 2 1)(4 3 1)(1 0 0)(0 0 1)(4 8 1)(4 1 1)(3 2 1)) 0 3)
(((1 4 0)(6 2 1)(4 3 1)(1 0 0)(0 0 1)(4 8 1)(4 1 0)(3 2 1)) 0 4)
(((1 4 0)(6 2 1)(4 3 1)(1 0 0)(0 0 1)(4 8 1)(4 1 0)(3 4 1)) 0 5)
(((1 4 0)(6 3 1)(4 3 1)(1 0 0)(0 0 1)(4 8 1)(4 1 0)(3 4 1)) 0 6)
(((1 4 0)(6 3 1)(4 3 1)(1 0 0)(0 0 1)(4 8 1)(4 1 0)(3 2 1)) 0 7)
(((1 4 0)(6 3 1)(4 3 1)(1 0 0)(0 0 1)(4 0 1)(4 1 0)(3 2 1)) 0 8)
(((1 4 0)(6 3 1)(4 3 1)(1 0 0)(0 0 1)(0 0 1)(4 1 0)(3 2 1)) 0 9)
(((1 4 0)(5 3 1)(4 3 1)(1 0 0)(0 0 1)(0 0 1)(4 0 0)(3 2 1)) 0 10)
(((1 4 1)(6 3 1)(4 3 1)(1 0 0)(0 0 1)(0 0 1)(4 0 0)(3 2 1)) 2 11)
(((1 4 1)(6 3 1)(4 4 1)(1 0 0)(0 0 1)(0 0 1)(4 0 0)(3 2 1)) 2 12)
(((1 4 1)(6 3 1)(4 4 1)(1 0 0)(0 0 1)(0 1 1)(4 0 0)(3 2 1)) 2 14)
(((1 4 1)(6 3 1)(4 4 1)(1 0 0)(0 0 1)(0 1 1)(4 5 0)(3 2 1)) 2 15)
(((1 4 1)(6 3 1)(0 4 1)(1 0 0)(0 0 1)(0 1 1)(4 5 0)(3 2 1)) 2 16)
(((1 4 1)(6 2 1)(0 4 1)(0 0 0)(0 0 1)(0 1 1)(4 5 0)(3 2 1)) 4 17)
(((1 4 1)(6 3 1)(0 8 1)(0 0 0)(0 0 1)(0 1 1)(4 5 0)(3 2 1)) 4 18)
(((1 4 1)(0 3 1)(0 8 1)(0 0 0)(0 0 1)(0 1 1)(4 5 0)(3 2 1)) 4 19)
(((1 4 1)(0 3 1)(0 8 1)(0 0 0)(0 0 0)(0 1 1)(4 5 0)(3 2 1)) 4 20)
(((1 4 1)(0 3 1)(0 8 1)(0 0 0)(0 3 0)(0 1 1)(4 5 0)(3 2 1)) 4 21)
(((1 4 1)(0 3 1)(0 8 1)(0 0 0)(0 3 0)(0 6 1)(4 5 0)(3 2 1)) 4 22)
(((1 4 1)(1 3 1)(0 8 1)(0 0 0)(0 3 0)(0 6 1)(4 5 0)(3 2 1)) 4 24)
(((1 4 1)(1 3 1)(0 8 1)(0 0 0)(0 5 0)(0 6 1)(4 5 0)(3 2 1)) 4 25)
(((1 4 1)(1 3 1)(0 8 1)(0 0 0)(0 5 0)(0 6 1)(0 5 0)(3 2 1)) 4 26)
(((1 4 1)(1 3 1)(0 8 1)(0 0 0)(0 5 0)(0 8 1)(7 5 0)(3 2 1)) 4 27)
(((1 4 1)(1 3 1)(0 8 1)(0 0 0)(0 5 0)(0 6 1)(7 5 0)(2 2 1)) 4 28)
(((1 4 1)(1 3 1)(0 8 1)(0 6 0)(0 5 0)(0 6 1)(7 6 0)(2 2 1)) 6 29)
(((1 4 1)(1 3 1)(0 8 1)(0 2 0)(0 5 0)(0 6 1)(7 6 0)(2 2 1)) 6 30)
                                    .
                                    .
                                    .
(((1 5 1)(5 0 0)(4 6 1)(0 0 0)(2 1 1)(2 4 0)(8 7 1)(5 0 1)) 12 2046)
(((1 5 1)(5 0 0)(4 6 1)(0 0 0)(2 1 1)(2 4 0)(8 7 1)(5 0 1)) 12 2049)
(((1 5 1)(5 0 0)(4 6 1)(4 0 0)(2 1 1)(2 4 0)(8 7 1)(5 0 1)) 12 2050)
(((1 5 1)(5 0 0)(4 6 1)(4 0 0)(2 1 1)(2 4 0)(2 7 1)(5 0 1)) 12 2051)
(((1 5 1)(7 0 0)(4 6 1)(4 0 0)(2 1 1)(2 4 0)(2 7 1)(5 0 1)) 13 2062)
-------------------------------------------------------------------------------------
total runtime 129.038000 sec
```

Each line corresponds to the current generation M. The column E indicates E(M), and G indicates the cumulative number of steps. The final machine of this trace accepts all strings in the right-list but none in the wrong-list of problem 3 (figure 5).

Figure 5: The final machine of problem 3



We show the result of the 13 other problems in figure 6, only by their final machines.

Figure 6: The Result of Construction

```
Problem                                                                          Steps
-------------------------------------------------------------------------------------
P1   ((0 1 1)(2 8 1)(7 4 1)(8 0 0)(7 2 1)(1 8 0)(7 7 0)(8 8 0)) 98
P2   ((0 5 1)(4 0 0)(2 8 0)(8 0 1)(1 5 0)(2 2 0)(1 4 1)(5 8 1)) 134
P4   ((4 1 1)(3 3 1)(1 2 0)(7 1 1)(6 0 1)(4 0 0)(0 3 1)(1 2 1)) 442
P5   ((6 3 0)(6 6 1)(8 1 0)(1 6 0)(0 1 0)(1 7 0)(5 7 0)(3 5 0)) 1768
P6   ((5 3 1)(6 8 1)(1 8 0)(7 8 0)(4 7 0)(5 7 1)(8 3 1)(4 1 0)) 277
P7   ((1 6 1)(0 0 1)(2 4 1)(3 7 0)(0 5 1)(7 6 1)(7 5 1)(4 2 1)) 206
P1-  ((3 6 0)(0 3 0)(3 7 1)(3 4 1)(0 3 1)(7 1 0)(0 4 1)(0 0 0)) 300
P2-  ((8 6 0)(2 3 1)(2 0 0)(2 8 1)(1 5 1)(1 4 1)(0 4 0)(4 2 1)) 89
P3-  ((3 6 0)(4 5 1)(1 4 0)(2 4 0)(5 7 1)(2 3 0)(0 6 1)(0 0 0)) 1939
P4-  ((4 7 0)(2 7 1)(2 6 0)(8 1 0)(7 5 0)(7 4 1)(1 3 0)(2 0 0)) 246
P5-  ((8 3 0)(6 6 0)(6 2 1)(1 7 1)(3 1 1)(5 8 1)(5 4 0)(1 5 1)) 1844
P6-  ((2 7 0)(7 1 1)(3 2 0)(5 1 1)(0 3 0)(3 7 0)(1 8 1)(3 1 1)) 886
P7-  ((7 6 0)(4 2 0)(4 2 1)(4 3 1)(6 2 0)(2 0 0)(7 5 0)(6 0 1)) 3725
-------------------------------------------------------------------------------------
```

2.3. Discussion

To see how effectively our hill-climbing algorithm has performed, we compare our method with an exhaustive search. There are $(9 \times 9 \times 2)^8$ = about 5×10^{17} machines in our problem space. We now want to know the number of the desired machines in our problem space, so that we can calculate the expected number of steps until the exhaustive algorithm finds the first desired machine. This can be done by the following "sampling" method: take one machine in the problem space randomly, and test if this machine is the desired machine; repeat this procedure 100000 times.

We show the expected number of steps using the exhaustive search calculated by this procedure in figure 7. Although the exhaustive search works better on "easy" problems, it is obvious in general that our hill-climbing works much better than the exhaustive search.

Figure 7: The number of Steps to get the desired machine

Problem	Hill-Climbing	Exhaustive-Search
P1	98	33
P2	134	315
P3	2052	> 50000
P4	442	12500
P5	1768	> 50000
P6	277	50000
P7	206	50000
P1-	300	167
P2-	89	1852
P3-	1939	> 50000
P4-	246	> 50000
P5-	1844	> 50000
P6-	886	> 50000
P7-	3725	> 50000

3. Simplification of Finite Automata

In the previous section, we saw that our hill-climbing method successfully produced a machine that accepts all positive sample strings but no negative sample strings. However, the final machine of the result of problem 2, for example, does not accept our desired regular set (1 0)*. For instance, it does accept a string (1 1 0 0), which is not in (1 0)*. We therefore want the machine to be "generalized" so that it accepts exactly (1 0)*. In fact, the final machines of all problems except problem 1, 3 and 7, need to be generalized.

We define the generality of a machine in terms of its simplicity. The simplicity of a machine is determined by the number of states the machine has, and if two machines have the same number of states, a machine with fewer arrows and final states is simpler.

Our task is to simplify the machines we have obtained in the previous section, so that the machines become the simplest or the most general.

We call this task simplification of finite automata, and it can be also done by using the hill-climbing method as in the previous section.

3.1. Algorithm

The algorithm of the simplification is essentially the same as the algorithm described in the previous section. The major differences are as follows: the evaluation function E(M) returns a higher value if the machine M is simpler; if M does not accept some strings in the right-list, or does accept some strings in the wrong-list, E(M) returns minus infinity; the algorithm starts with the result of the previous experiment instead of a random machine.

3.2. Result

The final machines of these experiments are shown in figure 8.

Figure 8: The Result of Simplification

```
[P2]  ((0 2 1)(1 0 0)) 7
[P4]  ((2 1 1)(3 1 1)(0 1 1)) 88
[P5]  ((4 3 1)(3 4 0)(2 1 0)(1 2 0)) 42
[P6]  ((3 2 1)(1 3 0)(2 1 0)) 174
[P1-] ((2 1 0)(2 2 1)) 145
[P2-] ((2 3 0)(2 2 1)(1 2 1)) 971
[P3-] ((1 5 0)(3 4 1)(2 3 0)(2 4 1)(2 1 0)) 363
[P4-] ((3 5 0)(2 2 1)(4 1 0)(2 0 0)(1 1 0)) <NOT-SIMPLEST>
[P5-] ((4 3 0)(6 6 0)(6 2 1)(1 5 1)(3 1 1)(5 4 1)) <NOT-SIMPLEST>
[P6-] ((2 3 0)(3 1 1)(1 2 1)) 44
[P7-] ((1 5 0)(4 6 0)(4 2 1)(4 3 1)(5 2 0)(4 0 0)) <NOT-SIMPLEST>
```

3.3. Discussion

We compare our method with an exhaustive search. The exhaustive search generates all machines in the order of simplicity, and the first machine that accepts all strings in the right-list but none in the wrong-list is considered the simplest machine. Thus we can calculate the expected number of steps until the exhaustive search finds the desired machine[3]. The result is shown in figure 9.[4] The symbol "---" indicates that the algorithm fails to find the simplest machine. This can happen when the hill-climbing algorithm climbs a "local hill".

Figure 9: The Number of Steps to obtain the simplest machine

Problem	Hill-Climbing	Exhaustive-Search
P1	98	4
P2	141	170
P3	2052	563933
P4	510	8524
P5	1810	553933
P6	451	8524
P7	206	553933
P1-	445	170
P2-	1060	8524
P3-	2302	46593884
P4-	---	553933
P5-	---	553933
P6-	930	8524
P7-	---	46593884

4. Concluding Remark

Our new approach to construction of finite automata from given examples has been shown to work successfully, although it could not find the simplest machines for some problems. To avoid climbing a "local hill", it might be possible to apply adaptive search ([6], [2]) instead of our simple hill-climbing.

[3] Let n be the number of states of the desired simplest machine. Then the expected number of the steps S_n is:

$$S_n = [\Sigma_{i=1 \text{ to } n-1} U_i] + [U_n / (2 \times (n-1)!)],$$

where U_j is the number of all possible machines with j states, that is,

$$U_j = (j+1)^{2n} \times 2^n$$

[4] The number of steps using hill-climbing in this figure is the sum of the number of steps to construct the 8 state machine and the number of steps to simplify it into the simplest machine.

Although our problem domain has been regular languages, we might be able to extend it to context-free languages by constructing Push-Down automata (finite automata with stack, see [7]) using a similar method.

Acknowledgements

I would like to thank Herbert A. Simon and Jaime Carbonell for supervising this work; Masakazu Nakanishi, Yuichiro Anzai, Pat Langley and Takeo Kanade for thoughtful comments on an earlier version of this work; and Cynthia Hibbard for helping to produce this document.

References

[1] Biermann, A.W. and Feldman, J.A.
On the Synthesis of Finite-State Acceptors.
AI Memo 114, Stanford University, April, 1970.

[2] Cavicchio, D.J.
Adaptive Search Using Simulated Evolution.
PhD thesis, University of Michigan, 1970.

[3] Feldman, J.A.
First Thoughts on Grammatical Inference.
AI Memo 55, Stanford University, August, 1967.

[4] Feldman, J.A.; Gips, J.; Horning, J.J.; and Reder, S.
Grammatical Complexity and Inference.
AI Memo CS125, Stanford University, June, 1969.

[5] Fogel, L.J.; Owens, A.J.; and Walsh, M.J.
Artificial Intelligence Through Simulated Evolution.
Wiley, New York, 1966.

[6] Holland, J.H.
Adaptation in Natural and Artificial Systems.
The University of Michigan Press, 1975.

[7] Hopcroft, J.E. and Ullman, J.D.
Introduction to Automata Theory, Languages, and Computation.
Addison-Wesley, 1979.

[8] Lindsay, R.K.
Artificial Evolution of Intelligence.
Contemporary Psychology 13(3), March, 1968.

RETRIEVING MEMORIES OF PERSONAL EXPERIENCES

Brian J. Reiser
John B. Black
Robert P. Abelson

Cognitive Science Program
Yale University

An important aspect of both comprehension and learning is the utilization of one's own past experiences to understand a current situation. In fact, being reminded of an experience often occurs in the process of retrieving generalizations from memory, suggesting that memories of personal experiences should be encoded in terms of the generic knowledge structures that are utilized in comprehension. Retrieval of these memories should therefore reflect the organization of generic knowledge (Schank, 1982). This paper explores the use of one such knowledge structure in the recall of past experiences.

Schank (1982) proposed that *Memory Organization Packets* (MOPs) represent knowledge about common activities. A MOP is represented as a sequence of *generalized scenes*, each of which consists of actions to accomplish a subgoal of the activity. For example, the RESTAURANT MOP would contain the scenes *Being-seated*, *Ordering*, *Eating*, and *Paying*. Generalized scenes can be referenced by more than one MOP. The generalized *Paying* scene contains the information that is true of paying in general, regardless of context. Each MOP consists of the generalized scenes that occur in that context, augmented by *context-specific* knowledge, a specification of how those scenes are modified (*colored*) for the particular situation. Each of the MOPs that refer to the *Paying* scene (e.g., MOVIE, GROCERY-STORE, RESTAURANT) must contain the information necessary to construct a specific colored version of that scene.

An experience typically contains many differences from the generalizations stored in generic knowledge structures. Schank (1982) argued that these deviations connect the contextualizing knowledge structure and memory for the individual experience. The connection serves as a *retrieval index* for the experience (Kolodner, 1980; Schank, 1982).

We propose that retrieval of an experience involves two types of processing: *(1) Establishing the context:* The context necessary for retrieval will be provided by the specific knowledge structures that were utilized to guide behavior in the experience. *(2) Finding an index:* A retrieval index describing the deviation from the generic structure provides a link to an individual experience. For example, the concept *restaurant* plus the index *I ate too much lasagna and felt sick* might retrieve a particular restaurant experience.

The importance of a search context has been suggested by previous researchers (Norman & Bobrow, 1979; Williams & Hollan, 1981), but is necessary to examine whether there are any functional differences between classes of knowledge structures in memory retrieval (Reiser & Black, 1982). Our hypothesis is that establishment of a MOP as the context will figure more importantly in the search process than other types of structures, such as generalized scenes. The unique aspects of adults' experiences are more likely to be deviations from context-specific knowledge (specified by a MOP), than from the more abstract knowledge represented in generalized scenes. Furthermore, retrieval of even those experiences which are stored as scene-deviations will require the utilization of a MOP to reconstruct the context-specific aspects of the experience. For example, one might remember not being able to find the right credit card while paying at a cash register, but initially fail to recall where the incident occurred, what was being paid for, etc. If a context such as DEPARTMENT-STORE or RESTAURANT could be retrieved, it would provide cues for reconstructing other aspects of the experience. Our view may be contrasted with the position that experiences are stored as arbitrary associations between concepts in networks, with *no* functional differences between different types of concepts in memory retrieval.

We examined the roles of MOPs and generalized scenes in memory retrieval in two autobiographical memory experiments. If it is generally necessary to retrieve a MOP structure to access a memory, then retrieval cues which do not specify a MOP should be inferior. If one is asked to remember a *restaurant-paying* experience, retrieval would be more efficient if the processing begins with the RESTAURANT MOP, rather than the generalized *Paying* scene. In addition, specification of the MOP containing a scene should lead to faster retrieval than specification solely of the scene.

Experiment 1

Subjects saw a pair of phrases separated by a 5 second delay, then recalled a personal experience that fit the two phrases. One of the phrases named a MOP, and the other phrase referred to a scene; the order of presentation of the phrases was varied. The MOP cue named a common activity (*took a ride on a train, went out drinking*). The scene cue described an action sequence that could occur in a number of different contexts. Two types of Scene phrases were used. Regular Scene cues described actions that are a normative component of an activity (*picked out what you wanted, paid at the cash register*), while Failure Scene cues described the failure of some goal of a scene (*didn't get what you asked for, couldn't find a seat*). All scene cues were carefully worded so as not to reveal any particular context.

Forty MOP and scene combinations were constructed

from twenty MOP, ten Failure Scene, and ten Regular Scene phrases. Each MOP was paired with both a Regular Scene and a Failure Scene cue; and each scene was paired with two MOPs:

1a. MOP + Failure Scene:	went out drinking; didn't get what you asked for
1b. MOP + Regular Scene:	went out drinking; paid at the cash register
2a. MOP + Failure Scene:	had your hair cut; didn't get what you asked for
2b. MOP + Regular Scene:	had your hair cut; paid at the cash register

Each subject received ten combinations involving each type of scene cue, so that the MOP phrase was presented first for half of the trials for each type of combination. Each MOP and scene were used only once for a given subject. (For example, a subject received items 1a and 2b, or items 1b and 2a.)

Subjects were instructed to recall an experience that fit the combination of the two phrases presented on each trial, and indicate whether they could remember such an experience by pressing either the *Yes* or *No* key. We emphasized that the memory be a *specific* experience, but that it was not necessary to recall all of the details of the experience before responding. After each *Yes* response, subjects wrote a brief description of the experience. Retrieval times were measured from the presentation of the second phrase until the button press.

Table 1 presents the mean retrieval times for the *Yes* responses for 32 Yale undergraduates. Subjects recalled experiences more quickly when the MOP cue appeared first [$min\ F'(1,44) = 7.98$, $p < .01$]. Secondly, Regular Scene trials yielded faster retrieval times than Failure Scene trials [$min\ F'(1,45) = 6.48$, $p < .05$]. The order of presentation equally affected the two scene types [interaction $F < 1$].

	MOP First	Scene First	Mean
MOP + Regular Scene	4.203	6.492	5.348
MOP + Failure Scene	5.986	8.394	7.120
Mean	5.094	7.443	6.269

Table 1: Retrieval Times (in seconds) for Exp. 1

The faster retrieval times when the MOP cue was presented first confirm the prediction that a MOP structure provides the context necessary to retrieve an experience. When the scene cue appears first, extra processing is required to reconstruct a MOP context, slowing retrieval. An alternative explanation is that when the scene cue is first, an episode is retrieved, but it may not match the MOP that is presented later. In contrast, when the MOP is first and a memory is retrieved, it is much more likely to match the scene cue. Hence, the scene first trials would be slower, because sometimes the retrieved episodes must be discarded and memory search resumed. However, this alternative explanation fails to account for the Failure Scene results. It assumes that memories retrieved with MOPs are likely to fit the scenes, while memories retrieved with scenes may not fit the MOPs. This is true for the Regular Scenes, since restaurant experiences typically contain a *Paying* scene, but paying is experienced in contexts other than restaurants. However, this is not true for Failure Scenes, since an episode retrieved from a MOP cue would *not* be particularly likely to fit the given Failure Scene description. Thus, the results are better explained by a model in which retrieval of the MOP is an essential stage in remembering an individual experience.

Since the MOP provides the context for retrieval, the scene cue provides a constraint on the use of the experiences that are stored with the MOP. Each MOP contains a pool of available indices that specify very salient experiences in that context. Subjects search that pool of indices to discover whether any of those experiences could fit the scene cue. For the Regular Scene trials, the subject is relatively free in drawing from this pool of indices — one must be sure only that the experience that is retrieved can be reconstructed to include the necessary scene. However, when a Failure Scene is presented, the use of available indices is severely constrained, since an index must be found that retrieves an experience containing the particular type of goal-failure that is described in the scene cue. This requires careful consideration of the pool of indices, and perhaps some inferencing about the reasons that such a goal failure would arise, thus adding extra processing to the memory retrieval. Therefore, subjects are slower to remember an experience for those trials involving Failure Scene cues.

Experiment 2

If constraining the target experience to a particular MOP context facilitates retrieval of an experience, then subjects should find it easier to remember an experience when given both a MOP and a scene (presented simultaneously) than when presented with a scene alone. However, if activation of a context is a simple matter of retrieving associations of a scene, then there should be little difference between presentation of a MOP and scene combination and the scene in isolation.

The facilitative nature of the MOP was tested in a second experiment by comparing retrieval times for three types of cues: (1) Scene alone, (2) MOP alone, (3) MOP + Scene combination. All MOP + Scene combinations from Experiment 1 were used; in addition, each MOP and each scene phrase was presented alone. Each subject received 10 trials of each cue type. (These trials were blocked by condition, to guard against the MOP of one trial facilitating the scene of the next trial.) The instructions differed slightly from Experiment 1. Subjects were told to recall an experience that fit the presented description consisting of one or two phrases. Since the materials in the three conditions necessarily differed in length, both reading and response times were collected for each trial. Subjects first indicated when they had read the cue, and then responded to indicate whether they remembered an experience that fit the cue. Retrieval times were measured from the subject's reading time button press until the memory retrieval response.

Table 2 presents the mean retrieval times for *Yes* responses in the three conditions for 36 Yale undergraduates. As predicted, subjects were able to

retrieve an experience more quickly when both a MOP and scene were presented, than when the scene was presented alone [*min* $F'(1,42) = 3.53$, $p < .10$; $F(1,35) = 8.43$, $p < .01$ for subjects; $F(1,18) = 6.08$, $p < .05$ for items]. Subjects were faster to respond to Regular than Failure Scenes, but this difference was only marginally significant [$F(1,35) = 3.08$, $p < .10$ for subjects; *ns* for items].

	Scene Alone	MOP + Scene	MOP Alone
Scene Type			
Regular Scene	5.296	3.383	
Failure Scene	5.292	4.307	
Mean	5.294	3.845	2.154

Table 2: Retrieval Times (in seconds) for Exp. 2

Since the MOP provides a better search context than the generalized scene, the combination is a better retrieval cue than the scene alone. Subjects are slower to respond to the combinations than to the MOPs alone, because the scene cue provides an extra constraint on the use of the indices that are stored with the MOP. The subject must be sure that the recalled experience includes the specified scene of the MOP when given a MOP + Scene combination, but any of the indices may be used when given the MOP alone.

Conclusions

The different structures we have discussed may be considered in terms of the amount of *constraint* they place on the search space — i.e., the set of experiences potentially satisfying the cue. A MOP constrains the set more than a generalized scene, since the scene can occur in multiple contexts. A MOP is somewhat less constraining than a MOP + Scene combination, since the combination specifies a particular segment of the event sequence. In addition, Failure Scenes are more constraining than Regular Scenes, since they specify a particular type of occurrence within a given scene.

Our results suggest that a MOP constitutes the optimal level of specificity for a memory cue. Generalized scenes are not constrained enough, since they become better cues when combined with a MOP, and the scene slows retrieval when presented before the MOP. Once a MOP has been accessed, constraints on the use of indices may *increase* retrieval time, since the most accessible indices may not retrieve experiences that satisfy the given cue. Thus, subjects are slower to remember an experience that satisfies a Failure Scene cue than a Regular Scene cue, and are slower to recall an experience that satisfies both a MOP and a scene cue than one that satisfies only the MOP cue.

In summary, we have argued that knowledge structures may be functionally distinguished by their effectiveness in providing a search context. Accessing a MOP is an essential part of retrieving a past experience from memory, since it provides an optimal search context, and can generate context-specific indices to retrieve memories stored with a scene. Specifying the activity type by naming a MOP is facilitative, but constraining the type of experience that occurred in that context may require extra processing to generate appropriate indices. We suggest that research on the use of memory in naturalistic tasks should focus on considerations of how the content of a generic memory structure is utilized to find and reconstruct a memory for a specific experience.

References

Kolodner, J. L. *Retrieval and organizational strategies in conceptual memory: a computer model.* Technical Report 187, Department of Computer Science, Yale University, 1980.

Norman, D. A., & Bobrow, D. G. Descriptions: An intermediate stage in memory retrieval. *Cognitive Psychology*, 1979, **11**, 107-123.

Reiser, B. J., & Black, J. B. Processing and structural models of comprehension. *Text*, 1982, in press.

Schank, R. C. *Dynamic memory: A theory of reminding and learning in computers and people.* Cambridge, MA: Cambridge University Press, 1982, in press.

Williams, M. D., & Hollan, J. D. The process of retrieval from very-long term memory. *Cognitive Science*, 1981, *5*, 87-119.

Personal Memory, Generic Memory, and Skill: A Re-Analysis of the Episodic-Semantic Distinction

William F. Brewer

Department of Psychology
University of Illinois
603 E. Daniel Street
Champaign, Illinois 61820

The purpose of this paper is to propose that human memory must be analyzed into three basic types: personal memory, generic memory, and skills. This analysis will only deal with productive memory systems and so will not cover recognition memory. After the classification is presented, it will be used as a framework to examine the initial work of Ebbinghaus (1885) and the episodic-semantic distinction proposed by Tulving (1972).

In order to make the distinction between the three types of memory clear, consider the following example: An undergraduate goes to the psychology building for a psychology experiment. He finds his way to the correct room, hesitates a minute, knocks on the door, and goes inside. He sees the experimenter and a memory drum in a small bare room. After some preliminary instructions, he is given a number of trials on a long paired-associate list. One of the items on the list is the pair DAX--FRIGID. After the experiment is over he breathes a sigh of relief and leaves the experimental room. This one event can be used to illustrate the three types of memory:

Personal memory. If, the next day, the undergraduate were asked, "Do you remember the psychology experiment you were in yesterday?" he might say something like: "Sure, I remember walking down to the room from the elevator. I remember feeling nervous as I stood there in front of the door. I remember opening the door and seeing the experimenter standing behind the table. I remember being surprised she was a woman. She had a white laboratory coat on, etc." If he were asked, "Was anything going through your mind while you were telling me all this?" the undergraduate might say something like "Yes, I was seeing in my mind's eye much of what I told you. I could see the door, the expression on the experimenter's face when I opened the door, etc." It is this type of memory that will be called personal memory in this paper.

Generic memory. If, some months later, the undergraduate were asked, "Do you remember that you were in a verbal-learning experiment several months ago?" he might say, "Yes." If asked, "Was anything going through your mind while you were giving me this answer?" he might say, "No, I just knew that I had been in the experiment. There were four experiments required for the course--two were filling out social psychology questionnaires, one was a perception experiment, and the other one was the verbal-learning experiment." This is an example of the type of memory that will be called generic memory.

Skill. If, some days later, the undergraduate were asked, "When I give you a nonsense syllable you tell me what word followed. DAX?" , he will probably say "FRIGID." If asked, "Was anything going through your mind when you gave the answer?" he might say, "No, I had practiced the list so many times I just knew what the response was." This is an example of rote memory, one type of skill.

This example was intended to provide an intuitive understanding of the distinction between the three types of memory. The next section attempts to give a general description of each type. This approach to human memory is an attempt to give a psychological version of the relevant philosophical works on memory in the last 70 years (Bergson, 1911; Russell, 1921; Furlong, 1951; von Leyden, 1961; Malcolm, 1963; Locke, 1971).

Personal memory. A personal memory is a recollection of a particular episode in the past of an individual. Personal memory is (always?) experienced in terms of some type of mental imagery--predominantly visual. It usually also includes non-imaginal information. The image is experienced as the representation of a particular time and location. The personal memory episode is accompanied by a propositional attitude that 'this occurred in the past' and is accompanied by a belief that the remembered episode was personally experienced by the individual. A personal memory is also frequently accompanied by a belief that it is a veridical record of the past episode. Personal memory statements frequently fit the linguistic frame: "I remember X." Thus, in the above example: "I remember the expression on the experimenter's face."

Generic memory. A generic memory is the recall of some item of general knowledge. Generic memory is not experienced as having occurred at a particular time and location and is not accompanied by a belief that the information was personally experienced by the individual. Generic memory statements frequently fit the linguistic frame: "I remember that X." Thus, in the earlier example: "I remember that I was in a verbal learning experiment." Semantic memory is the subclass of generic memory which involves the memory for abstract propositional information--for example: 'good is the opposite of bad' or 'the speed of light is a constant.' The operation of semantic memory does not typically carry along with it an experience of mental imagery. Thus when asked, "What is the opposite of good?" the correct answer is given without report of any mental imagery. Perceptual memory is the subclass of generic memory which involves the memory for perceptual information--for example: a map of the United States or the Statue of Liberty. The operation of generic perceptual memory does typically involve mental imagery. Thus, if asked, "Is Oklahoma to the south of Kansas?" or "Which hand of the Statue of Liberty holds the torch?", most individuals will report a "generic" mental image. These generic images are not typically experienced as involving a particular time and location. The similarities and differences between a generic perceptual memory and a personal memory can be examined by the following exercise. Recall the center of your university campus (i.e., form a mental map); now recall your most recent walk across that campus. The first is a generic perceptual memory; the second is a personal memory.

Skill. A skill is the ability to perform a given sequence of motor or cognitive actions. A practiced skill is typically not accompanied by mental imagery. There are a number of subtypes of skill that need to be distinguished. Motor skills refer to the ability to carry out a sequence of mo-

tor actions. This type of memory underlies the ability to ride a bike or hit a tennis ball. Rote skills refer to the ability to repeat a sequence of linguistic objects. This type of memory underlies the ability to repeat the alphabet or give one's social security number. Cognitive skills refer to the ability to carry out some sequence of cognitive operations. This type of memory underlies the ability to take the square root of a number or to make the verb agree in number with the subject in a spoken sentence. Many statements involving skills fit the linguistic frame: "I remember how to do X." Thus, "I remember how to ride a bike, how to say the alphabet, how to take a square root." In the next section of the paper the framework developed above is used.

Ebbinghaus. Ebbinghaus' 1885 monograph showed that it was possible to carry out experiments on human memory. However, in addition to this powerful achievement his work also served to limit the experimental investigation of memory to a particular subclass of memory--that of skill. In the initial pages of the 1885 monograph Ebbinghaus contrasts personal memory with skills. He apparently chose to focus on skill memory for methodological reasons (i.e., no need to use introspective data). In fact, within the area of rote skills, he chose the savings method over the recall procedure because he felt there might still be an important phenomenal component to recall tasks, whereas with the savings method he would just be comparing (behavioral) performance measures. This initial methodological decision by Ebbinghaus had an enormous impact on psychology--for 85 years in psychology the study of memory was the study of rote skills.

Tulving. In the late 1960's a few psychologists were able to break out of the Ebbinghaus focus on skills and began to carry out experiments on semantic memory (e.g., Collins & Quillian, 1969). In a seminal paper Tulving (1972) pointed out the fundamental difference in this type of experiment and formulated the distinction between semantic memory and episodic memory. The definition of semantic memory outlined above essentially follows Tulving's usage. However, Tulving's restriction of this type of memory to linguistic knowledge seemed too narrow, so I adopted the term generic memory for the larger class and the term semantic memory for the propositional subclass (see Hintzman, 1978, and Schonfield & Stones, 1979, for similar arguments).

The construct of episodic memory, as used by Tulving, is harder to deal with. When it is defined in abstract terms, it seems close to personal memory as outlined above. Thus, Tulving states that episodic memory "stores information about temporally dated episodes or events and temporal-spatial relations among these events" (p. 385) and proposes that statements from episodic memory refer to "a personal experience that is remembered in its temporal-spatial relation to other such experiences" (p. 387). However, the examples given by Tulving suggest that things are not that simple. Thus, one of the 4 examples of episodic memory was the statement, "Last year, while on my summer vacation, I met a retired sea captain who knew more jokes than any other person I have ever met " (p. 386). Taken at face value this appears to be an example of generic memory as the term has been used in this paper. A clear example of a personal memory would have been a statement such as, "I remember sitting on the stool at the bar, drinking a hot toddy while he told the traveling sailor joke, etc." One of the other examples suggests a more fundamental difficulty. "I know the word that was paired with DAX in this list was FRIGID " (p. 387). In terms of the classification suggested above this is either an example of generic memory ("I remember that DAX was the word paired with FRIGID") or an example of a rote skill (given DAX the subject says "FRIGID"). The latter interpretation is supported by Tulving's statement that the typical memory experiment in psychology is an episodic memory task (p. 390). Thus, the term episodic memory as used by Tulving apparently includes personal memory, plus semantic memories about autobiographical information, plus skills. In sum, the analysis presented here suggests that the distinction between semantic and episodic memory be replaced by the more analytic distinction between personal memory, generic memory, and skill.

Research on personal memory. The classification of memory into three basic types has powerful implications for empirical research. It is clear that the important topic of personal memory has been little studied by experimental psychologists (probably because of the residual restrictions left by Behaviorism). At Illinois we are currently trying to ask some of the relevant questions: What are the basic parameters of personal memory? (Brewer, in preparation) Are personal memories veridical? reconstructed? (Brewer, in preparation) How are generic memories derived from personal memories? (Brewer & Dupree, in preparation) What are the phenomenal properties associated with the different types of memory? (Brewer & Pani, in progress).

References

Bergson, H. Matter and memory. London: Allen & Unwin, 1911.

Brewer, W.F. Autobiographical memory. In preparation.

Brewer, W.F., & Dupree, D.A. Memory for episodic and generic information. In preparation.

Brewer, W.F., & Pani, J.R. Phenomenal reports in tasks involving personal memory, generic memory and skills. In progress.

Collins, A.M., & Quillian, M.R. Retrieval time from semantic memory. Journal of Verbal Learning and Verbal Behavior, 1969, 8, 240-247.

Ebbinghaus, H. Memory. New York: Dover, 1964 [original German edition 1885].

Furlong, E.J. A study in memory. London: Thomas Nelson, 1951.

Hintzman, D.L. The psychology of learning and memory. San Francisco: Freeman, 1978.

von Leyden, W. Remembering. New York: Philosophical Library, 1961.

Locke, D. Memory. Garden City, NY: Anchor Books, 1971.

Malcolm, N. Knowledge and certainty. Englewood Cliffs, NJ: Prentice-Hall, 1963.

Russell, B. The analysis of mind. London: Allen & Unwin, 1921.

Schonfield, D., & Stones, M.J. Remembering and aging. In J.F. Kihlstrom and J.E. Frederick (Eds.), Functional disorders of memory. Hillsdale, NJ: Erlbaum, 1979.

Tulving, E. Episodic and semantic memory. In E. Tulving and W. Donaldson (Eds.), Organization of memory. New York: Academic Press, 1972.

Temporal Judgments about Natural Events

Norman R. Brown
Lance J. Rips
and
Steven K. Shevell

University of Chicago

The information one remembers about the time of an event is rarely as precise as one would like. For a few consequential events, exact dates can sometimes be recalled; for example, one might remember that John Kennedy's assassination took place on November 22, 1963 or that Pearl Harbor was attacked on December 7, 1941. But aside from these blockbuster events and from recurrent events like birthdays and holidays, exact and explicit dates are usually unavailable. Even fairly important events, such as Spiro Agnew's resignation or the DC-10 crash in Chicago, which could hardly have escaped our notice at the time of their occurrence, now are difficult to date accurately. Things could be otherwise. Events could be logged in memory in the way they are recorded in almanacs, and in this case determining when an event occurred would amount to simple table lookup. But since access to specific remembered dates is uncommon for ordinary events, it is of interest to examine the more indirect means that people use in reckoning how long ago such events happened.

With a few brave exceptions (e.g., Linton, 1975), previous research on temporal memory has been limited to the study of short intervals (on the order of minutes or hours) and to brief events (usually words or syllables) presented to the subject in the laboratory. Examples are the "time perception" experiments of Fraisse (1963) and Ornstein (1969), and the literature on recency judgments in list learning (e.g., Hacker, 1980). Our investigation focuses on people's accuracy in dating natural events over longer intervals. Like the earlier research, however, we employ experimental methods to test individuals' memory for such facts. In this respect, our studies parallel many current investigations of spatial knowledge and cognitive maps.

The Accessibility Principle

Consider an event such as the Chicago DC-10 crash, for which no exact date is retrievable. How could one go about estimating its relative time of occurrence? One possibility is based on the obvious fact that, generally speaking, the longer an event is retained in memory, the less one can remember about it. Thus, given events that are equivalent in other respects, the event about which one remembers most is likely to be the one that happened most recently. We call this rule the "Accessibility Principle," since it asserts that the more accessible the information about an event, the more recent that event will seem. Of course, this principle is hardly foolproof. Factors like the initial salience of an event or its similarity to other events can influence the amount of information retained about it, beyond any effect of sheer passage of time. There is even evidence that, under certain conditions, recallable information can actually increase with delay (Erdelyi & Kleinbard, 1978). Nevertheless, the Accessibility Principle may still be useful as a rough guide to the time of an event, even though subject to error from variables like salience (as we demonstrate below).

We view the Accessibility Principle as a close kin to the Lack of Knowledge Inferences described by Collins (1978) and to the Availability Heuristic of Tversky and Kahneman (1973). The difference is that while Lack of Knowledge and Availability are used to draw conclusions about frequency or probability, the Accessibility Principle yields conclusions about the age of unique events. In the former case, one reasons that since one can't remember the event well, it probably happened infrequently or not at all. In the latter case, one reasons that since one can't remember the event well, it probably happened long ago.

Subjective Age of Paired Events of the 1970's

A straightforward prediction of the Accessibility Principle is that events that are retrospectively vivid and memorable should seem more recent than events that are not (other things being equal). Consider, for example, the DC-10 crash in Chicago and the DC-10 crash in Antarctica of about the same period. Since the DC-10 crash in Chicago is comparatively more memorable than the one in Antarctica, the Chicago crash should be judged more recent, even though, in point of fact, it happened six months earlier (May 25, 1979 vs. November 28, 1979).

We tested this prediction in an experiment using 19 pairs of events like the two DC-10 crashes that were matched as closely as possible for actual time of occurrence and for the content of the events themselves. The pairs included sports and cultural events (e.g., Saul Bellow wins the Nobel Prize vs. Burton Richter wins the Noble Prize) as well as standard news stories, all of which occurred between 1973 and 1980. Within each pair, one of the events was designated as more memorable than the other on the basis of ratings collected from two judges, neither of whom were aware of the hypothesis under investigation. A complete list of the pairs, together with their true dates and memorability status, is given in Table 1. In the experiment proper, the 38 individual events were read to subjects in random order, and the subjects were asked to respond to each with a number that best represented how recently the event happened. The numbers were chosen from a 0-to-9 scale, with high values corresponding to recent events and low values to old ones. We informed subjects before the start of the experiment that all of the events took place after 1970. Since the 15 subjects were of college- or graduate-student age, all of them had lived through the time of the target incidents.

Mean recency ratings from these subjects are also displayed in Table 1. Although on average the true date of the memorable events is slightly earlier than that of the less memorable ones (a difference of .05 years), subjects' ratings place the memorable events later. The overall mean rating for the memorable events is 5.7, whereas the mean for the less memorable events is 5.1. These ratings differed significantly when either subjects or event pairs are considered a random effect [for subjects, $F(1,14) = 20.43$, $p < .01$; for events, $F(1,18) = 4.58$, $p < .05$; however, quasi-$F(1,25) = 4.01$, $.05 < p < .10$]. As an example of this

Accessibility outcome, 80% of the subjects rated the Chicago DC-10 crash as occurring after the Antarctica crash, despite the fact that the opposite order is the correct one. Table 1 also reveals a number of exceptions to the Accessibility predictions, although in most cases these are from pairs in which the difference in memorability is small. As one would expect, the correlation between memorability and recency ratings is significant for these stimulus items [$r(36) = .38$, $p < .05$].

Recall and Perceived Age of Events in 1982

Although our prediction was confirmed that more accessible events seem more recent, measurement of accessibility (the memorability ratings) was fairly indirect for the events of the first study. In a second experiment, we have evaluated accessibility more directly by measuring subjects' recall of events, rather than relying on ratings. We predict that the larger the number of propositions about an incident that a subject can recall, the more recent that incident will seem. In this new experiment, the basic recency judgments and recall protocols were obtained from separate subject groups. Notice, however, that the act of recall may itself make the associated events more accessible. For this reason, it is of interest to compare recency ratings from subjects who have just completed recalling the events and recency ratings from subjects who have not engaged in recall. If recall increases accessibility, then ratings of recency-after-recall should be systematically greater than ratings of recency-without-recall.

The target events in this study were 40 headline-type incidents that were culled from the front pages of the Chicago Tribune and the New York Times between January 4 and January 11, 1982. This collection of events included items such as: Richard Allen resigns as National Security Advisor, the first U.S. test-tube baby leaves the hospital, and the U.S. drops its anti-trust suit against IBM. Since we were interested in tracking the relationship between recency and recall at different intervals after the events took place, we tested several independent groups of subjects: one Recall and one Recency group during the week immediately following the last target event, a second pair of Recall and Recency groups during the week beginning 15 days after the last event, and a third pair 60 days after the last event. To assess our hypothesis that recall increases apparent recency, we also asked subjects in the 60-day Recall group for recency ratings after they had completed their recall protocols. Recency ratings were elicited in a way similar to that of the first experiment (except that the subjects were told that the events happened in the 1980's rather than the 1970's). Recall subjects were given the same event names (e.g., Richard Allen resigns) and were asked to write down all of the facts they could remember directly related to the named events. The recall score for each incident was calculated as the average number of true atomic propositions recalled about it (see Kintsch, 1974). Stricter scoring methods (e.g., counting only directly relevant true propositions) yielded the same pattern of results. Fifteen subjects participated in each of the Recall and Recency groups.

The main results from this second study are given in Table 2 in the form of Spearman correlations between recency ratings and recall scores. Also shown in Table 2 are the correlations between recency and the events' true dates. Two facts about these data stand out. First, as the Accessibility Principle predicts, recall and recency are significantly correlated at each of the three intervals. Data from the first interval are especially interesting since they are least likely to be influenced by media retellings and follow-up reports. Second, and somewhat surprisingly, the number of propositions recalled is a better predictor of recency than the actual date of occurrence at all three intervals. In addition, a trend in the rating data followed the prediction that subjective recency would increase following recall. The average recency rating after recall was 5.7 for subjects in the 60-day Recall group; however, the average rating from the 60-day Recency group was 5.3. But although this trend was significant when tested over events [$F(1,39) = 13.07$, $p < .01$], it was nonsignificant when tested against subjects [$F(1,28) = 1.28$, $p > .10$].

Implications

According to the Accessibility Principle, the apparent age of an event depends upon the amount of information about it that one can bring to mind. This principle gained credence from the results of our first study, in which more memorable events were rated as taking place more recently than similar events of approximately equal objective age. The second experiment strengthened the case for Accessibility by demonstrating that the number of facts recalled about an event is a powerful predictor of its subjective time of occurrence. We have little doubt that other cognitive processes can also affect temporal judgments for natural events like these. As we have acknowledged, certain influential or recurrent events may be tagged with dates; the time of lesser events may be estimated through their causal connections to these influential ones. Still, a glance at the items in Table 1 suggests that causal links to datable events may not always be present, and in these circumstances, the Accessiblity Principle may be the dominant method for temporal judgments.

The Accessibility hypothesis bears an analogy to classical strength theories of time perception, which predict that the strength of the memory trace at the time of test determines the apparent age of the associated event (see the references cited by James, 1890, Pp. 632-633, and more recently, Hinrichs, 1970, and Morton, 1968). Pure strength theories, however, have not fared especially well in tests involving multiple list learning (Hintzman & Block, 1971; Flexser & Bower, 1974). By implication, these earlier results suggest that the mechanism responsible for our accessibility effects is not as simple as a unidimensional quantity connected to one's memory for an event. Our experiments leave the exact nature of the underlying mechanism as an open question. Nevertheless, the similarity mentioned above between the Accessibility Principle, the Availability Heuristic, and Lack of Knowledge Inferences may indicate that we are tapping part of a very general and complex inductive procedure.

Acknowledgments

We thank Martin Ringle and David Zager for their advice and assistance. We also acknowledge the Sloan Foundation for its support of this research.

References

Collins, A. Fragments of a theory of plausible reasoning. In D. L. Waltz (Ed.), Theoretical issues in natural language pro-

cessing-2. New York: Association for Computing Machinery, 1978.

Erdelyi, M. H., & Kleinbard, J. Has Ebbinghaus decayed with time?: The growth of recall (hypermnesia) over days. Journal of Experimental Psychology: Human Learning and Memory, 1978, 4, 275-289.

Flexser, A. J., & Bower, G. H. How frequency affects recency judgments: A model for recency discrimination. Journal of Experimental Psychology, 1974, 103, 706-716.

Fraisse, P. The psychology of time. New York: Harper & Row, 1963.

Hacker, M. J. Speed and accuracy of recency judgments for events in short-term memory. Journal of Experimental Psychology: Human Learning and Memory, 1980, 6, 651-675.

Hinrichs, J. V. A two-process memory strength theory for judgments of recency. Psychological Review, 1970, 77, 223-233.

Hintzman, D. L., & Block, R. A. Repetition and memory: Evidence for a multiple trace hypothesis. Journal of Experimental Psychology, 1971, 88, 297-306.

James, W. The principles of psychology. Vol 1. New York: Holt, 1890.

Kintsch, W. The representation of meaning in memory. Hillsdale, N.J.: Erlbaum, 1974.

Linton, M. Memory for real-world events. In D. A. Norman and D. E. Rumelhart, Explorations in cognition. San Francisco: Freeman, 1975.

Morton, J. Repeated items and decay in memory. Psychonomic Science, 1968, 10, 219-220.

Ornstein, R. E. On the experience of time. Baltimore: Penguin, 1969.

Tversky, A., & Kahneman, D. Availability: A heuristic for judging frequency and probability. Cognitive Psychology, 1973, 5, 207-232.

TABLE 1
Stimulus Events, True Dates, and Mean Recency Ratings, Experiment 1

Event Pairs	Date	Recency Rating
1. Reagan and Bush nominated by the Republican convention.	7/80	8.2
Carter and Mondale nominated for a second term by the Democratic convention.	8/80	7.5
2. Dustin Hoffman won an Academy Award for Kramer vs. Kramer.	4/80	7.8
Sally Field won an Academy Award for Norma Rae.	4/80	7.2
3. A DC-10 crashed in Chicago.	5/79	7.1
A DC-10 crashed in Antarctica.	11/79	5.5
4. Lord Mountbatten assassinated in Ireland.	8/79	5.9
U.S. Ambassador Adolph Dubs assassinated in Afghanistan.	2/79	6.7
5. The Supreme Court affirmed a lower court decision ordering California Medical School to admit Allan Bakke.	6/78	6.5
The Supreme Court ruled that labor unions could distribute material of a political nature at an employment site.	6/78	5.1
6. David Berkowitz was arrested on a murder charge.	8/77	6.7
Gene Leroy Hunt was arrested on a murder charge.	4/78	5.3
7. West German terrorists hijacked a Lufthansa airliner.	10/77	6.0
An alleged bank robber, Thomas Hannan, hijacked an airplane in Nebraska.	10/77	4.1
8. Roots won an Emmy Award.	9/77	6.5
Eleanor and Franklin won an Emmy Award.	9/77	6.1
9. Annie opened on Broadway.	4/77	6.1
The Gin Game opened on Broadway.	10/77	3.5
10. Saul Bellow won a Nobel Prize in literature.	10/76	5.4
Burton Richter won a Nobel Prize in physics.	10/76	4.3

TABLE 1 (cont.)

11. Bruce Jenner won an Olympic Gold Medal in the decathlon.	7/76	5.1
Evelin Schlaak won an Olympic Gold Medal in the discus throw.	7/76	3.9
12. Mao Tse-tung died.	9/76	4.8
Chou En-lai died.	1/76	5.4
13. Muhammad Ali KOs Joe Frazier.	10/75	4.3
Muhammad Ali KOs Jean-Pierre Coopman.	2/76	4.6
14. E. L. Doctorow's Ragtime published.	7/75	4.7
Irving Stone's The Greek Treasure published.	9/75	5.0
15. Linda Ronstadt's Heart Like a Wheel won a Gold Record.	1/75	6.3
John Denver's An Evening with John Denver won a Gold Record.	2/75	4.6
16. Aristotle Onassis died.	3/75	4.8
H. L. Hunt died.	11/74	5.1
17. Steve Garvey wins baseball's Most Valuable Player award.	11/74	5.3
Jeff Burroughs wins baseball's Most Valuable Player award.	11/74	4.1
18. Patty Hearst kidnapped.	2/74	4.1
J. Reginald Murphy, editor of the Atlanta Constitution, kidnapped.	2/74	5.1
19. Spiro Agnew resigned as Vice Pres.	10/73	3.0
Nelson Rockefeller resigned as Governor of New York.	12/73	4.5

Note. The first member of each of the pairs was rated as the more memorable. The standard error of the above means is .46.

TABLE 2

Spearman Correlations between Recency Estimates, True Dates, and Number of Recalled Propositions, Experiment 2

	Number of Propositions Recalled	True Date
Recency Rating		
+0 Days	.80***	.18
+15 Days	.69***	.41**
+60 Days	.68***	.34*

*p < .05
**p < .01
***p < .001

Psychological Issues Raised by an AI Model of Reconstructive Memory

Janet L. Kolodner
Department of Computer Science
Georgia Institute of Technology
Atlanta, Georgia 30332

Lawrence W. Barsalou
Department of Psychology
Emory University
Atlanta, Georgia 30332

1. Introduction

This paper presents some psychological implications of an AI model of reconstructive memory. Psychologists have characterized human memory as reconstructive for years (e.g., [1], [6]). AI simulation of reconstuction goes further since computer implementation requires explicit specification of processes and representations. The particular AI model we consider here is Kolodner's [5] E-MOP based model, implemented in a computer program CYRUS. The model has three interrelated components: a retrieval process, an underlying memory organization, and processes for developing memory organization with the encoding of new events. The retrieval process was designed to imitate reconstructive retrieval strategies observed in people. The memory organization both supports and causes reconstructive retrieval. Processes for developing memory organization build new knowledge structures (i.e., learn) as new events are encoded. These new knowledge structures enable subsequent reconstructive retrieval of the new events.

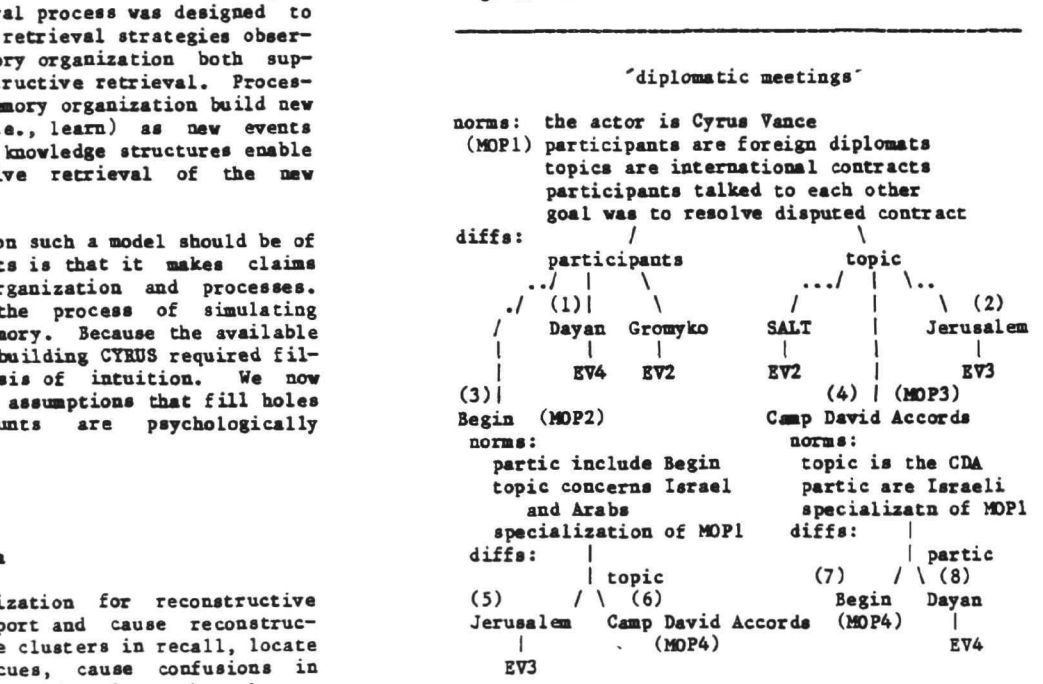
One important reason such a model should be of interest to psychologists is that it makes claims about human memory organization and processes. These claims stem from the process of simulating human reconstructive memory. Because the available model was incomplete, building CYRUS required filling it in in on the basis of intuition. We now ask whether the added assumptions that fill holes in psychological accounts are psychologically valid.

2. The E-MOP model

2.1 Memory Organization

A memory organization for reconstructive retrieval must both support and cause reconstruction. It must generate clusters in recall, locate and develop retrieval cues, cause confusions in recall and recognition, and emulate other characteristics of human remembering. Kolodner's memory organization uses conceptual categories called Episodic Memory Organization Packets, or E-MOPs (similar to Schank's MOPs [7]) that organize episodes in memory. A central assumption is that there is one E-MOP for each type of activity a person may be involved in, where type is defined as events that achieve a similar goal. Diplomats are involved in "diplomatic meetings", "diplomatic trips", "negotiations", and "state dinners". Each individual event is stored in the E-MOP(s) it fits into.

E-MOPs incorporate both episodic and generic memory (commonly, but incorrectly called semantic memory). The generic component consists of generalizations describing most of its members (i.e., some members may exhibit violations of these "norms"). Most "diplomatic meetings" discuss an international contract, for example, but a particular meeting might be called to plan an international event. An E-MOP's second component is its organization of member episodes. Episodes are organized based on how they differ from the E-MOP's norms. An episode is indexed and retrieved from an E-MOP by its relevant differences. When more than one episode has the same difference, a new sub-MOP is formed based on their similarities and differences. The figure below illustrates this organization:



The norms are features characteristic of diplomatic meetings. The episodes are indexed according to their similarities and differences in topic and participants. Meetings with the same topic or participants form E-MOPs whose norms are composed of their similarities. Below these norms, a set of similar instances are subsequently differentiated by their differences (i.e., mapped into indices). CYRUS organizes E-MOPs in three ways: (1) hierarchically, as just described; (2) by causal, temporal, and containment relationships between normative features; (3) by these same relations between indices.

2.2 Maintaining memory organization over time

Encoding new episodes requires both retaining the old organization to some extent and accomodat-

The work of the first author was partially supported by NSF under grant No. IST-8116892.

ing it to the new input. When a new episode is indexed identically to an old one, a new E-MOP must be formed to subsume them. That is, E-MOP formation is triggered by "reminding" [7], which occurs when the new episode retrieves the other similar one. The new E-MOP's norms are the similarities between the two items, and its indices are their differences. Because generalizations about a kind of event based on only two items may be inaccurate, subsequent episodes encoded with this E-MOP are used to refine these norms. If a feature not a norm for the first two episodes turns out to be normative for most others, what was initially an index can become a norm. Similarly, if a false generalization were made, norms for the first two instances can be relegated to indices.

2.3 Retrieval

Retrieval cues are abstracted from requests to remember an event. Such requests can be partial or complete specifications of the event to be retrieved. A request specifies an E-MOP to be searched and which indices within the E-MOP are to be traversed to find the event. An important assumption is that an E-MOP index cannot be traversed unless it is specified. In this way, retrieval is directed by the information in the request and further information that can be derived from it.

Since this process can fail in several ways, reconstructive strategies are proposed to deal with various types of failure. First, if the information in a request does not specify an E-MOP to be searched, then one or a small set of E-MOPs must be chosen. This process sees if any of the features stated in the request have E-MOPs associated with them (i.e., schema triggering).

A second type of failure stems from E-MOPs being untraversable unless their indices have been specified. A retrieval cue may specify features that don't correspond to E-MOP indices. Or, a retrieval cue may be so general that it doesn't specify enough features to direct traversal processes to a unique item. In that case, plausible features corresponding to E-MOP indices must be inferred from the given retrieval cues. A "meeting with Menachim Begin" might plausibly have taken place in Jerusalem. The strategies which make these inferences capitalize on an E-MOP's norms and knowledge about plausible relationships between different event features. Once such information has been specified, the corresponding indices are traversed. Interestingly, both types of strategies mentioned so far can lead to retrieval confusions and false starts.

A third type of strategy derives from the relationships between events in memory. Individual events refer to other events they are related to. If an event related to the requested event can be better specified, the related event can be used to further specify the requsted one. To recall a particular museum visit, for example, one might attempt to recall the trip it was part of.

3. Psychological Issues

This model stems from observations of how people remember, and what they forget. Although the processes and organization used to construct a complete model of reconstruction seem to work, are they really psychologically valid? One aspect of the model that has received empirical investigation to a large extent is reconstructive retrieval strategies [8]. People appear to elaborate upon requests to remember in many of the ways CYRUS does. Nevertheless, many issues remain untouched or at least require further attention. How does the organization of a set of events constrain the manner in which people elaborate on retrieval cues? That is, to what extent are such strategies content-dependent? Do people use the elaboration strategies used by CYRUS? Do they use others? What strategies are used most often, in what order, and for what reason? Similar to the content-dependence issue is the context-dependence issue. To what extent is elaboration affected by immediately previous searches for other events? for the same event? Given retrieval failure while searching an organized set of events, how do people select new parts of the organization for search? How does a retrieval access change organization? How sensitive is retrieval to incorrectly specified cues? Is the model too dependent on correctness?

Perhaps the most central issues the model raises are: How are events organized in memory? And how does this organization change over time? CYRUS assumes that events are the fundamental organizing units in memory. Is this true of human memory? If not, then what are the fundamental units? There may be several types of organizing principles. Others to be considered are: location (e.g., a local restaurant or bar); time (e.g., Christmas, summer); participants (e.g., Nixon, a spouse, a close friend)? If there are several ways events are organized, what determines which will apply to a given set of events? The content of the events? The goal the organization will serve? Perhaps several organizations simultaneously exist over a set of events.

A related issue concerns knowing what feature(s) should be used to discriminate two episodes sorted to the same E-MOP. There may be numerous features that distinguish two events, but only those that will be useful in the later evolution of generic knowledge should be chosen for indexing. How can such features be chosen? Another related issue is how many indices are grown each time reminding occurs. Another central issue concerns E-MOP construction. Is a new E-MOP constructed every time someone is reminded of an old event by a current one? To what extent is generic structure automatically acquired from and imposed on events? Or is conscious attention necessary to abstract normative information from previous events, organize it into E-MOPs, and apply it to new events? Human data may be informative on these points.

In the proposed memory organization, MOPs and their sub-MOPs form hierarchies in which common properties are stored once at the highest possible point in the hierarchy. This economy of storage parallels what psychologists call "cognitive economy" [2]. To date, it appears that the organization of semantic memory (i.e., lexical meaning) violates cognitive economy [3]. But to what extent is this violation true of other types of generic knowledge? Does human organization of events reflect cognitive economy? Or do people have much looser, less integrated and non-inclusive organizations for events?

An important aspect of E-MOPs is that they combine "episodic" and "generic" memories. This implies that episodic and generic memory are not separate entities but are intimately connected. If this is so, what exactly is the connection? When does episodic information (e.g., E-MOP indices) become generic (e.g., E-MOP norms or frame information)? When and how does generic information become confused with episodic information to

generate confusions? In E-MOPs, both happen as generalizations are refined and corrected.

There are a number or topics not covered in the original model which are nonetheless important to a theory of human memory organization and retrieval. One such issue concerns the roles of automatic versus conscious processes that encode information into memory. Temporal, spatial, and frequency information appear to be automatically acquired — even without knowing they are doing it, people encode these fundamental aspects of events [4]. In contrast, the acquisition of content information often seems to depend more on the use of conscious attention. When such information doesn't receive attention, the information is not acquired. How do these two types of processes interact to store events? Conscious attention may be responsible for the construction, organization, and reorganization of generic structure, since it usually contains content information. Automatic processes may be responsible for the strengthening of generic knowledge and the integration of spatial and temporal information into it. Finding algorithms for these latter phenomena and interfacing them with content-oriented processes appears to be an interesting problem.

A related issue is the role of similarity among events. This factor can facilitate people's memory performance on some occasions and interfere with it on others. Observing such phenomena in people's memory for events may further constrain the way in which we view generic knowledge of events and its use during retrieval. In E-MOPs, when a property doesn't correlate with other events, an index is set up differentiating the event with the deviant property from other events in the E-MOP. Correlation and differentiation play the role of keeping events suitably accessible. An event which conforms to the norms of an E-MOP will not be easily accessible because it won't have many indices differentiating it. On the other hand, events which have differentiating features will be accessible if those features are specified in or derived from a retrieval cue. What is the role of similarity and differentiation in people's memories? What is the actual effect they have on memory's organization?

Analogy is another area not covered in the original model. CYRUS does not address the migration of generic information from old E-MOPs to new ones. Generic knowledge associated with a particular E-MOP might be useful, however, in creating a new related E-MOP or in understanding something in a similar E-MOP. To what extent does "generic" structure generalize from one set of events to another? Must a completely new structure be built for each new set or does transfer occur? What procedures transfer the structure of an old E-MOP to a new one? How can knowledge in one E-MOP (e.g., for Vance) be used to understand something about a related referent (e.g., Haig)?

4. Future Directions

We are currently designing experiments that we hope will help answer the questions above. The experiments, no doubt, will raise additional questions. As a joint Artificial Intelligence and Psychology project, we will address these questions in the same way we have found it profitable to consider their ancestors --by building computer programs and by collecting human data.

5. References

[1] Bartlett, R. (1932). Remembering: A Study in Experimental and Social Psychology. Cambridge University Press, London.

[2] Collins, A.M., & Quillian, M.R. (1969). Retrieval time from semantic memory. JVLVB (8), 240-247.

[3] Conrad, C. (1972). Cognitive Economy in semantic memory. JEP (92), 149-154.

[4] Hasher, L. & Zacks, R.T. (1979). Automatic and effortful processes in memory. JEP:G (108), 356-388.

[5] Kolodner, J. L. (1980). Retrieval and organizational strategies in conceptual memory: a computer model. Research Report #187. Dept. of Comp. Sci., Yale, New Haven, CT.

[6] Norman, D.A. & Bobrow, D.G. (1979). Descriptions: An intermediate stage in memory retrieval. Cognitive Psychology (11), 107-123.

[7] Schank, R. C. (1980). Language and memory. Cognitive Science (4), 243-284.

[8] Williams, M. W. and Hollan, J. D. (1981). The Process of Retrieval From Very Long-Term Memory. Cognitive Science (5), 87-119.

SOFT CONTROL OF COGNITIVE PROCESSES

Michael R. Fehling

USC/Information Sciences Institute

4676 Admiralty Way

Marina del Rey, CA 90291

Submitted to:

Fourth Annual Conference

of the Cognitive Science Society

Ann Arbor, Michigan August 4-6, 1982

Topic:

Problem Solving and Knowledge Structures

Sponsor:

Gary M. Olson

1. INTRODUCTION

A critical feature of any problem solving system is its control structure. This, of course, refers to a mechanism (and its associated knowledge) used allocate processing resources among the various components of the system as they are needed to carry out some task within some problem domain. It is clear that the control structure of a problem solver is a major determiner of that system's ability to efficiently and effectively carry out any task.

As important as the notion of control is, it is surprising that so little work has been devoted to it by either computer scientists interested in developing expert systems or psychologists interested in modeling human cognition. It has been the feeling in Artificial Intelligence that, if there were enough knowledge available in the construction of an expert system, the problem of selecting an appropriate control structure would be a minor one (Feigenbaum, 1977). And, as we shall discuss below, although some recent psychological models have addressed issues that are closely related to the control problem, little or no research has directly addressed the general question of control of cognitive processes.

In this paper we report some work we are doing on the control problem. The ultimate goal of this research is to design and implement an expert system that controls other expert systems. That is, we are developing a problem solving system that is specialized to select and maintain a control regime for components of another "embedded" expert system. Our Expert System Controller (ESC) is able to reason about control. It uses both general knowledge and domain specific knowledge of the embedded system to create and maintain control plans for scheduling the use of the embedded system's component processes.

It is our belief that the issue of reasoning about control is one that must be addressed by anyone interested in developing more powerful problem solving systems, whether those systems are intended as expert systems or as models of human cognition. Moreover, it is a central premise of our research that such systems require soft control. By this term we mean the following:

> The ability to apply problem solving techniques to the problem of control itself (i.e., to reason about control)
>
> The ability to select from alternative control plans the one that is most appropriate in a particular task environment
>
> The ability to apply general (albeit less powerful) knowledge when specific domain knowledge is unavailable
>
> The ability to opportunistically deviate from a selected control strategy as a response to new information.

Soft control yields a flexibility of interaction among the various components of domain and control knowledge that allow for opportunistically allocating resources to activities most likely to make efficient progress in completing the task at hand.

2. META-COGNITION and CONTROL

First, let us discuss control in terms of human cognition by consider the vast amount of psychological research on the use of strategies to guide processing. A brief examination of research on this topic shows that, in any given task context, some particular processing strategy may be proposed as the organizer and controller for a more basic set of cognitive skills. Strategy guided models

have been offered as a description of many types of cognitive skills. Examples include models for text processing (e.g., Clark, 1978), logical inference (Revlin & Leirer, 1978), memory retrieval (Brown, 1978), perception (Kolers, 1972), and so forth.

Perhaps a generalization and expansion of the idea of processing strategy is Flavell's (1976) concept of meta-cognition. Meta-cognition refers to cognitive processing involving knowledge about other cognitive processes or the results of other cognitive processes. One place where this concept has been used extensively is the research on the topic of learning strategies (e.g., O'Neil, 1978; O'Neil & Speilberger, 1979).

This research demonstrates the ubiquity of task specific strategies. Each strategy appears to be a kind of specialized "control plan" that organizes the cognitive processes underlying performance in a particular task domain. This, in turn, suggests that there exists some general mechanism to produce these specialized control plans and to monitor their use. Although little work has been done to determine the characteristics of the meta-cognitive mechanism, we note the following important features.

> First the diversity of strategies that arise in different contexts indicates that these meta-cognitive structures are typically highly "tuned" to the specific problem domain Thus, both creation and selection for use of such control plans is a function of specific domain knowledge.
>
> Second, the use of strategies is opportunistic in that use of one strategy may be interrupted or even abandoned in favor of another known strategy as a response to some special circumstance that is noticed during task performance.
>
> Third, control can revert to more general knowledge and problem solving techniques when situation specific knowledge is insufficient.

These observations together indicate that meta-cognition is probably best modeled as what we refered to above as a mechanism for soft control.

Now let us consider the need for soft control in the context of expert systems in Artificial Intelligence research. We wish to show that there is a need for soft control in expert system just as that required for models of human cognition.

Recall Feigenbaum's argument, mentioned earlier, that the control problem for expert systems is secondary to the problem of representing sufficient knowledge about the problem domain. The knowledge in an expert system embodies primarily expert "rules of thumb" and descriptions for when such knowledge is applicable. Any such rule of thumb is typically a large chunk of domain specific knowledge that has compiled into it the control that would have been necessary to take the several smaller steps that are equivalent to it. Reasoning with such large chunks produces shorter inference chains which, therefore, greatly reduce the magnitude of the control problem for managing these inferences. In this sense most expert systems simply finesse the control problem by relying upon a very powerful set of domain specific principles that embody both domain knowledge and control assumptions for use of that knowledge.

Unfortunately, the exclusive use of expert rules can have severe limitations. The powerful domain principles of the expert system are usually only plausible rules of inference which do not embody logically necessary relationships. Hence, expert systems of this sort can fail precipitously at the limit of their knowledge, that is, when the system encounters new situations for which the special rules do not apply. When such rules fail the system is unable to retreat to weaker but more general methods of inference to determine such things as why the rule failed in this case, how to modify it to fit, or at least, how to start from smaller and less efficient but more universal principles to derive a response to the new situation. That is, control is too rigid to allow the system's performance to gracefully degrade as the limits of its expert knowledge are reached.

A general solution, which we have adopted, is to provide the expert system with an ability to revert to the more basic form of problem solving when the expert rules do not apply. However, control methods for using the expert rules will probably be useless for the more complex inferences required when using general principles. So the expert system must be able to select (or construct) a new control plan that is appropriate for the type of knowledge being used at each point in the task. Moreover, the system must detect when and how to make a graceful shift from one mode of inference to another. In general, the system must be able to develop or select from a stock of control plans that allow the system to use a variety of types of knowledge during task performance.

Therefore, to build a more flexible expert system or a more general cognitive model, one must design a system that has the ability to reason about control. Furthermore, the system must be able to select an appropriate control regime for a specific task context. It must have the ability to apply expert "rules of thumb", or, when such rules are not available, it must be able to engage in novel reasoning using finer grained and less specific logical rules. And it must be able to decide when to do which. That is, either a cognitive model or an expert system needs a means to provide soft control.

3. THE EXPERT SYSTEM CONTROLLER (ESC)

Next, we briefly describe some features of an expert system we are developing that realize the concept of soft control. As stated earlier, the primary application of this system is as an expert system to control other expert systems. However, in creating a problem solving architecture in which both domain and control plan reasoning are supported, we are developing a type of model that may also be valuable as a framework for developing cognitive models in which meta-cognitive processes, as well as the processes and knowledge they control, can be explicitly described.

In order that our Expert System Controller (ESC) have the capability to provide soft control, it must have the following features.

> An architecture which supports problem solving about selection, modification, and use of control plans as well as problems within a substantive problem domain.
>
> A representation language for expressing control relations (e.g., sequencing, tests, parallelism, etc.)
>
> The ability to opportunistically modify or

abandon a control strategy in response to new information.

ESC is an extension of the Hearsay-III problem solving system (Erman et al., 1981). Hearsay-III is a "blackboard model" in which knowledge is represented by a collection of individual processing components called "knowledge sources" (KSs). KSs embody the knowledge associated with a particular part of a problem solving task and are activated by the occurence of patterns on a "communications blackboard". KSs can interact during problem solving by leaving new "triggering" patterns on the blackboard that activate other KSs. Since more than one KS can be activated at a time, a "scheduler" is provided that makes decisions as to the firing order of the activated KSs. (For the reader unfamiliar with the architecture of blackboard models, see Rummelhart, 1977, pp. 103-116.)

Hearsay-III provides blackboard structures for both domain and scheduling purposes and provides for the implementation and use of knowledge sources for scheduling as well as domain knowledge sources. Thus, reasoning about scheduling can be accomplished by methods that are consistent with those used for problem domain reasoning. In order to extend the problem solving capabilities of this model to the full domain of control concepts we are adding an explicit control representation and a mechanism to react to that representation. The control notions that can be represented include

Programmatic control relations (e.g., sequential, parallel, or conditional)

Non-programmatic control relations (e.g., co-operative subprocesses [all of which combine to contribute to some goal] versus competitive subprocesses [each of which provides an alternative way to achieve a subgoal])

Descriptors of problem structures, goals, and knowledge sources.

Descriptors of hierarchical plans as well as descriptors of conditions under which control "jumps" out of such a plan in a non-hierarchical way.

Methods based on some work we have been doing using the Dempster-Shafer calculus (Shafer, 1976), to express preference relations among plans and activities (cf. also, Barnett, 1981).

Besides developing an explicit representation for reasoning about control, we are augmenting the architecture of Hearsay-III to fully support the control domain as a problem-solving activity. This extension provides abilities such as the following

Interpretation of control plans in the representation alluded to above.

Filling out of partially specified control plans using domain independent control knowledge to affect the elaboration

Optimization of execution within plans by using any applicable general knowledge.

Construction of control plans using preference relations supplied by scheduling knowledge sources when more specific control constraints are not available.

Communication of plan progress to scheduling knowledge sources, thus allowing the scheduling knowledge sources to modify and improve plans opportunistically.

4. CONCLUSIONS

The explicit control representation and other modifications we are making to the basic Hearsay-III architecture provide a means to achieve soft control in a problem solving system. We believe this model will be useful as a framework for building expert systems that have greater flexibility and power than those currently available.

This framework should also interest cognitive scientists whose concern is models of human cognitive since it provides a framework for a model in which meta-cognitive processes are treated uniformly with all other cognitive processes. Newell (1980) has pointed out that, if we are to get rid of the homunculus that always controls the processes of cognitive models, we must incorporate into those models a representation of the way that strategies arise from general and domain specific knowledge as a response to task conditions. Perhaps the issues that we have raised in developing our notion of soft control will help to evict this homunculus.

Note: This research was supported by Defense Advanced Research Projects Agency contract MDA 903-81-C-0335. Views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official opinion or policy of DARPA, the U.S. Government, or any other person or agency connected with them.

References

Barnett, J. A. Computational Methods for a Mathematical Theory of Evidence. In Proc. 7th Joint Conf. on Artificial Intelligence. IJCAI, Vancouver, B. C., 1981.

Brown, A. L. Knowing When, Where, and How to Remember: a Problem of Meta-Cognition. In R. Glaser (Ed.), Advances in Instructional Psycholgoy, Hillsdale, N.J.: Lawrence Erlbaum Associates, 1978.

Clark, H. H. Linguistic Processes in Deductive Reasoning. Psychological Review, 1978, 72(4), 387-404.

Erman, L. D., London, P. E., & Fickas, S. F. The Design and Example Use of Hearsay-III. In Proc. 7th Joint Conf. on Artificial Intelligence. IJCAI, Vancouver, B. C., 1981.

Feigenbaum, E. A. The Art of Artificial Intelligence. In Proc. 5th Int. Joint Conf on Artificial Intelligence, pages 1014-1029. IJCAI, Cambridge, MA, 1977.

Flavell, J. H. Meta-cognitive Aspects of Problem Solving. In L. B. Resnick (Ed.), The Nature

of Intelligence, Hillsdale, N. J.: Lawrence Erlbaum Associates, 1976.

Kolers, P. A. Some Problems of Classification. In J. F. Kavanaugh & I. G. Mattingly (Eds.), Language by Ear and by Eye: The Relationships between speech and Hearing, Cambridge: M.I.T. Press, 1972.

Newell, A. Reasoning, Problem Solving, and Decision Processes: The Problem Space as a Fundamental Category. In R. S. Nickerson (Ed.), Attention and Performance, Hillsdale, N. J.: Lawrence Erlbaum Associates, 1980.

O'Neil, H. F. (Ed.). Learning Strategies. New York: Academic Press 1978.

O'Neil, H. F. & Speilberger, C. D. (Eds.). Cognitive and Affective Learning Strategies. New York: Academic Press 1979.

Revlin, R. & Leirer, V. O. The Effect of Personal Biases on Syllogistic Reasoning: Rational Decisions from Personalized Representations. In R. Revlin & E. Mayer (Eds.), Human Reasoning, Washington, D.C.: Winston, 1978.

Rummelhart, D. E. Human Information Processing. New York: Wiley 1977.

Shafer, G. A Mathematical Theory of Evidence.: Princeton University Press 1976.

STYLES OF THINKING: FROM ALGEBRA WORD PROBLEMS TO PROGRAMMING VIA PROCEDURALITY[1]

Kate Ehrlich
Elliot Soloway
Valerie Abbott

Department of Computer Science
Yale University
P.O. Box 2158
New Haven, Connecticut 06520

1. ABSTRACT

Algebra word problems are often surprisingly hard for college students to solve. However, more students are able to solve these problems correctly when asked to write a computer program, than when asked to write an equation. We have also found that programmers, with the same level of math experience as non-programmers, do consistently better on the algebra word problems, after only one semester of an introductory programming class. We argue that some of the difficulty associated with the algebra word problems can be traced to misconceptions about what the algebraic expression represents. Students often appear to use an algebraic expression as if it were a static description rather than as denoting an active operation being performed by one number to get another number. Although programmers may be equally prone to such misconceptions, it seems that experience with programming helps them to overcome these misconceptions, by encouraging them to develop a more active, procedural view of the problem.

2. INTRODUCTION

In recent work, Clement and Lockhead [Clement, Lochhead, and Monk, 1980] have demonstrated that there is a class of apparently simple algebra word problems that students find very difficult to solve correctly. A typical problem is shown as Example 1 in Table 2-1.

When Clement et al gave this problem to a group of engineering students they found that only 63% of the group gave the correct response of S = 6P. The most common wrong answer was: 6S = P, the reversal of the correct answer. In another problem, also shown in Table 2-1, in which there are two integrals, only 27% of the class were able to produce a correct answer. These findings are very robust and have been replicated in a number of studies (e.g. Soloway et al, 1982; Clement et al, 1980; , Kaput, 1979).

Clement and his colleagues [Soloway et al., 1982, Clement, Lochhead, and Monk, 1980]. carried out videotaped interviews with some of these students to try and find the source of the errors They identified two principal kinds of strategies that students were using to solve the problems. Some students used a syntactic, word order matching strategy, in which the order of key words, such as "student" and "professor" and the numbers from the problem description, were mapped directly onto the order of symbols appearing in the equation. Paige and Simon [Paige and Simon, 1966] have also argued for the weaknesses inherent in this kind of direct translation strategy.

Another strategy that students adopted can be characterized as "static comparison". For instance, one student described the equation in the following manner:

> There's six times as many students, which means it's six students to one professor and this (points to 6S) is six times as many students as there are professors (points to 1P).

What's wrong with these strategies? Their main problem is that students seem to have a static, descriptive view of algebra. For instance, students who espouse the strategy denoted as "static comparison" seem to want the algebraic expression to represent directly the relative sizes of the objects in the problem. In so doing, they treat the variables such as S and P as standing for "students" or "professors" rather than for the *number* of students or the *number* of professors. But, algebra does not function as a description in the same way as English provides descriptions. The correct equation, S = 6P does not describe the sizes of the groups, rather it denotes an equivalence relation that would obtain if one of the groups, the professors, were made six times larger. In this way, the algebraic expression represents an active operation that is performed on one number to obtain another number.

3. IMPACT OF PROGRAMMING: PROCEDURALITY

If the correct conception of algebra is an active, procedural one, then putting students in an environment that encourages them to adopt a more procedural approach should help them to generate correct solutions to the algebra word problems. Programming is such an environment. Indeed, Papert [Papert, 1971] has claimed for some time that learning to program can enhance problem solving skills.

In previous research [Clement et. al., 1980, Soloway et al., 1982], we found that significantly more students could solve the problems correctly when the problem was presented in the context of writing a computer program than in the context of an equation. We have also conducted videotaped interviews with some of the students who were unable to write the equation [Soloway et al., 1982]. In several cases, we found that the same student was able to solve the problem in the context of a computer program but not in the context of an equation, even when there were only a few minutes separating the two solutions. These results support the claim that it is easier to write a program to solve a certain class of problems that to write an equation.

4. PROGRAMS VS EQUATIONS: THE CONTRIBUTION OF PROCEDURAL WORDING

In the study reported in [Soloway et al., 1982], the instructions for the two versions of the problem are worded a little differently. In particular, the instructions for the program version are themselves more procedural than the instructions for the equation version. Thus, it may be that the critical factor in the study was the wording of the instructions rather than any difference between writing an equation or a program. If the wording was the critical factor, however, there should be a difference between the two kinds of wording for non-programmers as well as programmers.

In the new study we used three versions of the algebra word problem; these are shown in Table 4-1. The equation and the program version are the ones used in the previous study; the

[1]This work was supported by the National Science Foundation, under NSF Grant IST-81-14840.

function version is new. We ran this study with students who had no programming experience as well as with students who had taken at least one programming course. The programmers received all three versions, while the non-programmers were given the equation and the function versions of the problem. Each student saw only one version of the problem.

The data, which are shown in Table 4-2, show that the procedural wording of the instructions had no effect on accuracy for the non-programmers. The programmers, on the other hand, did write more correct equations when given the procedural instructions than when given the original, equation version. As in the previous study, there was also a significant improvement for writing programs over writing equations with non-procedural instructions. The results show that the procedural wording of the instructions only improves performance if students have had programming experience. One implication of these results is that procedural wording alone is not sufficient to induce people to adopt a more active view of algebra; people need experience in a procedural domain such as programming.

5. TRANSFER EFFECTS FROM PROGRAMMING TO ALGEBRA

The results of the previous study suggest that it is experience with programming rather than the procedurality of the instructions that is critical. In the next study we examined more directly whether programmers do better on the algebra word problems than non-programmers, when the problems are presented in the standard non-procedural context.

We constructed a large diagnostic test containing 17 algebra word problems as well as filler items. The test was administered to 28 people with no programming experience and 32 people who had just completed a semester of an introductory programming course. The groups were equated for level of math experience and in many other respects had similar academic backgrounds. Many of the people taking the programming course had majors in non-scientific subjects such as History or English, while some of the non-programmers had majors in fields such as psychology which includes some math experience in the form of statistics.

EXAMPLE 1

Given the following statement:

"There are six times as many students as professors at this University"

Write an equation to represent the above statement. Use S for the number of students and P for the number of professors.

- Result: 63% correct
- Typical wrong answer: 6S = P

EXAMPLE 2

Given the following statement:

"At Mindy's restaurant, for every four people who order cheesecake, there are five people who order strudel."

Write an equation to represent the above statement. Use C for the number of cheesecakes ordered and S for the number of strudels ordered.

- Result: 27% correct
- Typical wrong answer: 4C = 5S

Table 2-1:
EXAMPLES OF ALGEBRA WORD PROBLEMS

All the problems were presented in a non-programming context and none of the problems had procedural wording. Over the 17 problems, the non-programmers got an average of 64.5% of the problems correct while the programmers got an average of 75% of the problems correct. This difference between the groups was significant ($t = 4.7$, $p < 0.0005$). Although the average performance between the two groups differed by only 10%, the significance of the difference reflects a small but consistent improvement over all the problems for the programming group.

It may be argued, that although we controlled for level of math experience, and academic background, the programming group as a whole were smarter than the non-programmers. If this is the case, we should expect to find a fairly constant rate of improvement over all the problems. However, the data do not support that argument. Some sense of the kind of advantage conferred by programming experience can be illustrated by examining one set of problems that were included in the test.

There are three main forms in which the solution equation can be expressed. It can be expressed as a multiple , e.g. 5C = 4S; as a ratio, e.g. C/S = 4/5; or with a single variable on one side, e.g. C = 4/5 S. The wrong solutions are most often expressed in the form of a multiple, the ratio is the form students seem most familiar with, and the third form is the one appropriate to the equation written in a computer program. We included in the test, a set of problems in which people were given solution fragments in each of these three forms.

The percent correct completions for the two groups of subjects are shown in Table 5-1. When we compared performance on each version of the problem across the two groups, we found that there was no reliable difference between

PROBLEM

"At Mindy's restaurant, for every four people who order cheesecake there are five people who order strudel."

1. EQUATION

Write a mathematical equation to represent the above statement.

2. PROGRAM

Write a computer program which can be used to calculate the number of cheesecakes ordered when supplied with the number of strudels ordered.

3. FUNCTION

Write a mathematical function which can be used to calculate the number of cheesecakes ordered when supplied with the number of strudels ordered.

Table 4-1:
EXAMPLES OF WORDING

NON-PROGRAMMERS

	FUNCTION	EQUATION
CORRECT	82	73
INCORRECT	40	49

Equation vs Function: N.S.

PROGRAMMERS

	FUNCTION	EQUATION	PROGRAM
CORRECT	71	48	77
INCORRECT	32	53	22

Equation vs Function: $p < 0.01$
Equation vs Program: $p < 0.001$
Function vs Program: N.S.

Table 4-2:
The number of people given each problem type who produced a correct and incorrect solution

	NON-PROG	PROG
(multiple) ? C = ? S	36%	50%
(ratio) ? C = - - ? S	68%	78%
(single letter) C = ? - S ?	36%	59%

Table 5-1: EQUATION FRAGMENT:
Percent correct responses for each solution type

the programmers and the non-programmers except on the fragment that had the form of a single letter on the left hand side ($\chi^2 = 3.35$, $p < .10$). These data mitigate against claims that the programmers may have done better because they were smarter. Moreover, the data suggest that experience with programming confers quite specific problem solving skills to other domains such as algebra word problems.

6. CONCLUSIONS

There are a number of reasons why programming may enhance certain problem solving abilities. These reasons range from the explicitness required by the syntax of programming languages, through to the practice of "debugging" and number checking that is encouraged in programming. However, perhaps the main benefit of programming is that it provides the student with a model of an active input/output transformation which functions as a metaphor of change. It seems clear that people should be encouraged to develop skills that help them to construct these kinds of models. The results of the studies we reported, suggest that these skills are best developed in the context of learning to program.

References

Clement, J., Lochhead, J., and Soloway, E. . ***Positive Effects of Computer Programming On Students' Understanding of Variables and Equations.*** Proceedings of the National ACM Conference, Nashville, Tenn., 1980.

Clement, J., Lochhead, J., and Monk, G. Translation Difficulties in Learning Mathematics. *American Mathematical Monthly*, 1980, *88(4)*, 26-40.

Kaput, J. Mathematics and Learning: Roots of Epistemological Status. In J.Lochhead & J. Clement (Eds.), *Cognitive Process Instruction*, : Franklin Institute Press, 1979.

Paige, J. and Simon H. Cognitive Processes in Solving Algebra Word Problems. In *Problem Solving Research, Method and Theory*, : John Wiley and Sons, New York, 1966.

Papert, S. ***Teaching Children to be Mathematicians Versus Teaching About Mathematics.*** Technical Report 249, MIT AI Lab, 1971.

Soloway, E., Lochhead, J., Clement, J. Does Computer Programming Enhance Problem Solving Ability? Some Positive Evidence on Algebra Word Problems. In R. Seidel, R. Anderson, B. Hunter (Eds.), *Computer Literacy*, New York, NY: Academic Press, 1982.

Arithmetic Procedures in Everyday Situations

Jean Lave
University of California, Irvine

The ubiquity and unremarkable character of routine activities such as grocery shopping qualify them as apt targets for the study of thought in its customary haunts. For the same reasons, such activities are difficult to analyze. I approach the task, however, in the conviction that the understanding of problem solving depends on an integrated conceptualization of the culturally crystallized activity-in-setting within which problems are realized. I have chosen to focus, therefore, on a social institution, the supermarket, which is highly structured in relation to a clearly defined activity-in-setting, grocery shopping.

The Adult Math Skills Project at U.C. Irvine has as its goal to explore arithemetic practices in the daily lives of their users. One branch of the project seeks to develop both theory and method for analyzing decision-making processes in grocery shopping, including the role of arithmetic in these processes. Michael Murtaugh's project, on which I draw heavily here, involved extensive interviewing, observation, and experimental work with twenty-five adult, expert grocery shoppers in Orange County, California. Detailed transcribed observations of preparation for shopping, a major shopping trip, and its aftermath provide data for the analysis sketched here (and set out in detail in our recent paper, Recounting the Whole Enchilada: The Dialectical Constitution of Arithmetic Practice. The Orange County residents vary in age from 21 to 80, in income from $8,000 per family to $100,000, and in schooling from 8th grade to an M.A. Twenty-two are female; all are native speakers of English whose schooling took place in U.S. public schools.

Certain aspects of activity settings have durable and public properties. For example, the supermarket is a durable entity--a physically, economically, politically and socially organized space-in-time. The supermarket, in this sense, is called an "arena." The supermarket as arena is outside of, yet encompasses, the individual, providing a higher order institutional framework within which "setting" is constituted. The setting of grocery shopping is the arena as acted in by the individual. The setting is the shopper's edited version of the arena, generated by his or her routine grocery shopping activity in the supermarket. As setting, some aisles of the supermarket do not exist as part of a shopper's field of action, while others are fine-featured areas in which the shopper routinely makes several choices and still others serve only as broad cues to a particular, routinely purchased items.

It is in this sense that it is possible to talk about the dialectical relation of setting and activity. A shopper passes the generic products with a sudden coming-into-focus of their funny, plain appearance. She stops to investigate, realizes there is a tradeoff between the comforts of known products and the possibility of lower prices. This creates a new category in her repertoire of money-saving shopping strategies, which in turn leads her to attend to it on the next trip, and on later trips perhaps, to make a regular check at this aisle before proceeding elsewhere. The setting for future shopping trips is thereby transformed into a more extensive routine route, and the activity of grocery shopping is transformed by change in the setting. On future visits a review of price-saving possibilities on a small but diverse set of products will precede consideration of the brand name projects in their usual locations.

Grocery shopping is composed of repeated processes of decision making which have the effect of reducing numerous possibilities to single items in the cart on the basis of qualitative characteristics which differentiate items. Arithmetic problem solving is both an expression of and a medium for dealing with stalled decision processes. It is, among other things, a move outside the characteristics of the product to its characterization in terms of a standard of value, money. It brings the particular decision process to an end if arithmetic calculation leads to a decision to purchase a particular item.

Given these circumstances and the predicament shoppers face, presented with an abundance of goods to choose from but no choice other than to make choices, arithmetic problem solving very often acts as a rationalization of essentially arbitrary "choices." Support for this interpretation of the role of arithmetic calculation in routine decision making (as serving to produce rational accounts for choices which are only apparent) comes from Murtaugh's research on decision processes used by shoppers in choosing grocery items. He demonstrates that arithmetic, if utilized in the course of choosing a particular grocery item, is employed near the end of the process, when the number of choices still under consideration is no greater than three and rarely greater than two. Thus, a partial analysis shows that thirteen shoppers purchased 450 grocery items. Of these items, 185 involved snag repair of some variety, and 79 of these latter items involved problem solving which utilized arithmetic. In all there were 162 episodes of calculating, approximately two calculations per item on which calculation occurred. It would be difficult to picture arithmetic procedures as major motivations driving shopping activity. Justifying choices just before and after the fact is a more appropriate description of the common role of arithmetic in shopping.

So far I have said that a "problem" in routine activities is an interruption or snag in individually constituted routine and that arithmetic is often used in a rationalizing capacity to overcome snags. A third critical characteristic of problem solving follows from the character of activity-setting relations as a whole (as analyzed in the full version of the paper): The relation between activity and setting is a dialectical one; (arithmetic) problem solving is part of that activity-in-setting and thus must conform to the same dynamic. It follows from this position that the activity-in-setting of grocery shopping is crucial in shaping problem-solving activities. The data support this view.

In the course of our research, shoppers took an extensive paper-and- pencil arithmetic test, covering integer, decimal, and fraction arithmetic, using addition, subtraction, multiplication and division operations. The sample of shoppers was constructed so as to vary in amount of schooling and in time since schooling was completed. Problem-solving success

averaged 59% on the arithmetic test, compared with a startling 100%--error-free--arithmetic in the supermarket, and this in spite of the fact that a number of problems on the test were constructed to have exactly the same arithmetic properties as problems grocery shoppers successfully solved in the supermarket.

Subtest scores on the math test are highly correlated with each other, but none correlates significantly with frequency of arithmetic problem solving in the supermarket. Number of years of schooling is highly correlated with performance on the math test but is not significantly correlated with frequency of calculation in the supermarket. Years since schooling was completed is significantly correlated with math test performance but not with grocery shopping arithmetic.

However it may be noted that my position is not one of extreme situational specificity. Although there is not time to discuss it here, I take the view that any activity-in-setting is interelated with interpenetrates, other activities-in-settings. These relations are the basis of the generality, in the sense of spread, or multiple use of, knowledge across situations, including arithmetic.

But the main point here is to illustrate the dialectical form of arithmetic problem solving in the routine activity-setting of grocery shopping. A successful account of problem-solving procedures will explain two puzzles uncovered in preliminary attempts to analyze grocery shopping arithmetic. The first is the error-free arithmetic performance in the supermarket by shoppers who made frequent errors in parallel problems in formal testing situations. The other is the frequent occurrence of more than one attempt to calculate during a single decision segment of shopping.

First it is useful to make explicit what is dialectical about the process of problem-solving. The routine nature of grocery shopping activity and the location of arithmetic at the <u>end</u> of decision-making processes about grocery items within the activity of grocery shopping suggests that there must be rich content and shape to a problem solution at the time arithmetic becomes an obvious next step. Problem-solving under these circumstances is an iterative process involving moves between what the shopper knows and the setting holds that might help, on the one hand, and what the solution looks like, on the other, since many of the solution's parameters are already in place as the result of the prior process of deciding, up to a point, what to purchase. The dialectical process is one of gap-closing between strongly specified solution characteristics and the inputs and procedural possibilities for solving the problem.

Thus, a change in either solution shape or resources of information leads to a reconstitution of the other: The solution shape is generated as the product of the decision process up to the snag. Problem identification changes the salience of setting characteristics. This in turn suggests, more powerfully than before, procedures for generating a specific solution; information and procedural knowledge accessed by mind and/or eye make possible a move towards the solution or suggest a change in the solution shape that will draw it closer to the information at hand.

These basic points are illustrated by a shopping episode in which the shopper, J. (a 43 year old woman with 4 children), discusses the price of noodles--noodles last week and this, big packages and little ones, different brands, and so on, as she replaces the family supply.

She begins by taking a package of noodles off the shelf and putting it in her cart. It is the kind she customarily buys, Perfection elbow noodles, 32 ounces, $1.12. As she does so she comments that it is cheaper than American Beauty noodles. It is clear from her action of placing the package in the cart that a decision has been made, and the decision prefigures and shapes the course of calculations to come. The arithmetic problem J. will work on during the rest of the segment is to decide which is the better buy, which gives more for the money: The one she purchased, or one of three sizes of American Beauty Noodles: 24 ounces for $1.02, 48 ounces for $1.79, or 64 ounces for $1.98. After a digression about goulash, J. and the anthropologist, (M.), get back to noodles.

> J.: There's large elbow /noodles/. This is really the too--large economy bag. I don't know if I, probably take me six months to use this one.... I don't know, I just never bought that huge size like that. I never checked the price though on it. But being American Beauty it probably costs more even in the large size.

J. here has reiterated somewhat more firmly than before her opinion that American Beauty is more expensive than other brands. The resolution of the numerical comparison is taking on clearer outlines. The next interchange starts a process of simplification of the arithmetic comparison. She transforms large number of ounces into a small number of pounds.

> M.: That's what, that's 6. . . J.: It's 4 pounds and what did I buy, 2?

That this is phrased as a question suggests that she is making a comparable change from ounces to pounds for the 32 ounce package in her cart as she has just made for the 64 ounce package on the shelf. The problem now looks like this:

Perfection noodles,	2 pounds	$1.12
American Beauty noodles	4 pounds	$1.98

She eventually simplifies the problem to

> 2 pounds for $1, 4 pounds for $2.

She concludes that they are equivalent buys, at 50 cents a pound. But she does not stop there. Her point is to demonstrate a difference in price per pound, so she starts yet another round of calculation with more specific prices, going back to $1.12 in order to produce a precise enough calculation to demonstrate the difference. Simplification does not become an end in itself, then. In these calculations it is just one possible step whose relation with the solution shape may lead either to an end to calculating, a return to more complex forms of calculation, or to a change in the solution shape.

All this goes by so fast that only repeated analysis of transcripts make clear that calculation has taken place at all. Meanwhile, in the course of the discussion there is yet another price comparison. J. looks at two packages of American Beauty spaghetti noodles, and sees what appears to be a justification for not buying a large bag:

J.: But this one, you don't save a thing. Here's 3 pounds for a dollar 79, and there's 1 pound for 59.

Having a solution, "you don't save a thing," confirmed, "Here's 3 pounds for a dollar 79 and 1 for 59," the process of looking at the bags while reading off the information required to justify the conclusion, leads to reassessment of the information: For the "1 pound package" in fact does not weigh a pound. Immediately she adds a second round of calculations:

J.: No, I'm sorry, that's 12 ounces. No, it's a savings.

Two rounds of calculation have just occurred. The first produced the conclusion that in both cases the noodles were essentially 60 cents per pound. Recognizing the weight error, only a "less than" inference would be required to move to the conclusion that the big bag is in fact a saving. And in the second round this is just what she does.

However, the "only" is deceptive, as is the conciseness of the transcript, if they convey the impression that the arithmetic is simple in paper-and-pencil, place-holding algorithmic terms. The problem in these terms would be to discover if one point seven nine divided by three is equal to point fine nine. An active process of simplification is required to transform this set of operations into the form that J. achieves. This kind of simplifying transformation, which preserves relations and simplifies numerical representations, is characteristic of grocery shopping arithmetic.

The pattern of moves made in the course of J.'s calculations is something like this: She starts with a probable solution, but inspection of evidence and comparison with the expected conclusion cause her to reject it. Given corrected information, she recalculates and obtains a new result. This whole process is what is meant by "gap-closing:" the weaving back and forth between the expected shape of the solution and the information and calculation devices at hand, in the course of which each is repeatedly transformed by the other.

One characteristic of the preceding account has been the need to assign multiple functions to individual moves in gap-closing arithmetic procedures. It seems to be the nature of dialectically constituted processes to pose severe problems of description. Perhaps one must give up the goal of assigning arithmetic problems to unique locations--in the head or on the shelf--or labelling one element in a problem-solving process as a calculation procedure, another as a checking procedure; or even distinguishing the problem from its answer. In such circumstances statement of the problem, solution to the problem, procedure for solving the problem, and checking activity, may be analytically indistinguishable.

In discussing these implications of a dialectical model of problem solving I have, among other things, been developing an explanation of the multiple-calculation, error-free arithmetic practiced in grocery shopping. Error-free arithmetic is not error-free because people do not make mistakes. Indeed, multiple calculations to repair initial difficulties, are the rule rather than the exception. Typical gap-closing procedures occur in "rounds." Dialectical processes of problem solving account for the multiple calculation phenomenon.

Why is the end product of calculating so extraordinarily accurate? The analysis cannot be presented in complete form here. But a major reason is that dialectical processes of problem-solving make possible powerful monitoring by the problem solver, due to the juxtaposition of problem, problem-solving procedure, solution and checking activity.

I have tried to cover a great deal of ground in a very short time. The talk can hardly do more than indicate the nature of the issues taken up in the paper itself. But in closing it might be useful to stress the major point of the exercise: the dialectical constitution of problem-solving in any particular activity setting grows out of the encompassing dialectical relation between the activity and setting within which it takes place. The nature of the dialectical relation between grocery shopping and the supermarket generate the routiness of the activity in setting in relation to which problems are constituted as snags or interruptions. Likewise, the dialectical relation between shopping and market setting generates the overdetermined nature of choice and the rationalizing character of problem solving; and the activity-in-setting directly gives the dialectical character to problem solving for it is part of that activity-in-setting.

Arithmetic problem solving is not "the same" everywhere and at all times. But this in no way negates the possibility of developing general theory about the constitution and reconstitution of activity in setting.

How novices solve physics problems

Eileen Scanlon
and
Tim O'Shea

Open University
Milton Keynes
Bucks
England

Abstract - The paper outlines ten claims about the performance of novices solving problems in physics. The claims are then evaluated from the literature, and from the results of a study where synchronised audio tape and paper and pencil working records of novices solving kinematics problems were made. Some alternative methodologies for investigating these claims are discussed and the future direction of the work indicated.

Introduction - The longterm objective of this study is to design instruction to improve physics problem solving. Various claims about how novice students go about solving physics problems can be made. Here are some of them.

1. Novices solve physics problems more slowly than experts and pause more frequently between the retrieval of successive equations or chunks of equations than experts do.

2. Novices have erroneous ideas about basic physics concepts.

3. Novices make meta statements (comments about the problem solving process).

4. Novices never check back or use real world checking.

5. Novices work backwards.

6. Novices don't apply physical intuition to a problem before actually trying to solve it.

7. Novices don't possess rich internal representations for complex problems.

8. Novices are not goal directed.

9. Novices use consistent strategies in problem solving.

10. Novices can be taught helpful problem solving strategies.

These claims will be discussed using two sources of evidence - reports in the literature, and the results of a study of solution protocols in kinematics. The claims are stated in order of certainty. This paper will take each claim in turn and assess its validity. Some are as yet unsubstantiated. Future work which might substantiate them will be discussed.

The literature - Previous empirical studies of how physics problems are solved have examined the knowledge structures discussed in the basic concepts (Shavelson 1974, Reif 1981), examined students prior conceptions of the physical world (misconcepts) (Champagne & Klopfer 1980, Gilbert & Osbourne, di Sessa 1981) and examined solution protocols (Larkin et al, 1981).

The Cyclops study - The study reported here involved the collection of solution protocols and their analysis in terms of problem solving strategies displayed and misconcepts revealed (Scanlon, 1981). Some recordings of Open University (OU) first year students attempts to solve physics problems were made. The equipment consisted of a summa graphics bitpad and microphone connected via an interface box to a stereo cassette recorder. This equipment based on the OU's Cyclops technology allows recordings of pencil and paper working to be made on one track of the cassette tape while the other track records any words spoken during the process. The equipment has been used to record children's mathematical behavior (O'Shea & Floyd, 1981). The system combines in a convenient form the students voice with a synchronised dynamic record of what he or she writes. This study has established that the system was suitable for recording the mixture of handwriting, diagrams and numbers present in a typical adult physics problem solving protocol.

The subjects were seven first year Open University students who had just completed three weeks of study on elementary mechanics. Their backgrounds varied from no previous experience of physics to A level physics. Open University students are adults returning to study after some work experience. In the attempt to attain an understanding of problem solving skills in physics there are advantages in using adult students. Skill at solving physics problems is not a natural competence but a learned skill - and one learned with considerable difficulty. Adults language competence is fully developed so the notorious difficulty of achieving verbalisation in protocols should be simplest with them (Horowitz 1980). The problems selected for the students were simple kinematics problems. From the replay of the Cyclops tape the sequence of operations, timing information on each individual step, and the verbal protocol indicates problem solving decisions made.

Discussion of the claims

1. Novices solve problems more slowly than experts, and pause more frequently.

Expert and novice protocols have been compared to highlight the differences (Simon & Simon, 1978, Larkin, 1981). Experts have been found to be 4 times faster at solving problems than novices. The pause times between the retrieval of successive equations or chunks of equations were quite different (Larkin 1979). Experts produced streams of equations without pausing while novices paused most of the time. In the Cyclops study the students experienced many difficulties with the problems.

2. Novices have erroneous ideas about basic physics concepts.

Trowbridge (1979) describes students problems with

the concepts of velocity and acceleration. The weaker students in the Cyclops study also had a very hazy notion of acceleration and constantly confused it with velocity. Velocity they confused with speed and average speed. See Fig. 1. However, the fact that students have an imperfect understanding of some of the concepts they need to use doesn't seem particularly surprising. When their understanding drops below a certain level - its obviously the most important thing to worry about. If you can't tell acceleration from velocity you're going to have trouble doing problems about either. However, what does it mean to understand a concept completely? Wouldn't some level of understanding be good enough for all practical purposes? We have to solve problems in real life in the absence of complete understanding. Some of the 'misconcepts' research seems also to draw questionnable conclusions. In Andy di Sessa's (1980) study of how high school students manipulate a dynaturtle he says reveals misconcepts about force and acceleration - but these students score highly on conventional tests. It reveals something about the physics not having got 'into the musculature' but who plays tennis using Newton's Laws?

3. Novices make meta statements

Simon & Simon (1978) observed the difference in the number of meta statements made. By meta statements they mean comments made by the students about the problem solving process. Experts made fewer meta statements than novices who made more frequent comments on errors made, the physical meaning of an equation, or overall direction. This finding is on the surface surprising but may be to do with the novice voicing uncertainties that an expert doesn't share. In the Cyclops study students made many such comments.

4. Novices never check back

The weaker students in the Cyclops study made many mistakes due to not carefully reading the problem statement. They misread distances for speeds, final speeds for average speeds etc. and despite the fact that these mistakes led them into numerous problems never looked back to check. Having struggled through to an answer to the problem the novices never checked back to see whether the answer made sense in terms of the original problem statement.

The better students in the Cyclops study highlighted the behaviour of the novices. They checked back to various stages - both during the problem to make sure they'd solved a sub-problem checked back to see if their answer made sense in terms of the numbers given in the problem. They also tried various ways of doing a problem and if something didn't seem to be working out they were prepared to start again in a different direction. They seemed less prepared just to plod on regardless of whether the solution path they'd chosen seemed to be successful or not.

5. Novices work backwards

The most contentious difference quoted in the literature is the difference in solution path - 'working forward - working backward' (Simon & Simon 1978). The expert works from the information given in the problem, producing equations which can be solved using the information given. The novice starts by generating an equation which contains the unknown he is trying to find and works backward. This finding seems strange but may be is explained by the confidence felt by the expert that the problem is soluble. This behaviour was not observed in the Cyclops study. Students mostly started by writing down the equations they knew.

6. Novices don't apply physical intuition

Experts seem to apply to prior qualitative analysis or physical intuition to the problem before actually starting to solve it. What seems to characterise this analysis is the ability to represent the problem physically in terms of some real world mechanisms (Larkin & Reif 1979). If novices relied on their physical intuition they might create a false analysis, (as they have erroneous ideas about basic physics concepts). In the Cyclops study among the novices no connection with the real world in solving the problem was apparent.

7. Novices don't categorise problems into types and don't possess rich internal representation

The expert has built up a set of fundamental sets of subroutines for basic types of problems and this classification into problem types takes place very quickly (Chi, Feltovich & Glaser 1980). An investigation of this appears in Chi, Glaser & Rees (1981). In answer to the question 'how does an expert construct a more efficient subroutine for a complex problem?' they reply that 'the facility lies in the rich internal representation the expert has generated'. The Cyclops study did not investigate this claim.

8. Novices are not goal directed

An important difference between experts and novices is that experts are confident enough that they will eventually succeed to be willing to try out various approaches. In the Cyclops study, the novices were playing a game of pretending to solve the problems. However they knew that really they couldn't so it didn't really matter what they wrote down. They appeared to conspire with the experimenter to pretend that they were looking for a solution path and made all sorts of meta comments. "I see. . . well suppose I try", but they were just trying to get any answer so that the problem will go away.

9. Novices use consistent strategies in problem solving

Several of the weaker students in the Cyclops study had 'a way of doing problems'. The protocols are littered with statements like: "This is how I always do problems" "I always draw a diagram" or "write down all the equations I know" or "write down everything in sentences". The last example is very interesting and came from a student who has a great deal of trouble with mathematics. She says that she never knows whether something makes sense unless she can write it down in the form of a sentence so this is how she argues her way to a solution.

The surprising result of the Cyclops study is that the poorer students did seem to be exhibiting some sort of consistent way of coping with being asked to do physics problems which they didn't know how to do. This is reminiscent of Kathy Larkin's experience of adults doing arithmetic problems.

They could remember how to do some things - they had 'Islands of Knowledge' (Larkin, 1978). The adults in the Cyclops study had 'Islands of tactics'. They were not basing their behaviour on understanding of physics but on some sort of 'coping strategy'.

Discussion - The first four claims seem incon-

trivertable. The fifth is substantiated in the literature but seems in contradiction to the eighth claim from the present Cyclops study that novices aren't goal directed. They only occasionally conspire with the experimenter to pretend they are. The sixth and seventh claim are also substantiated though what 'a rich internal representation' means has yet to be defined or demonstrated. Most of these claims are in fact disclaimers - they're statements about what the novices don't do. The ninth claim is made on the basis of the present study and remains to be fully substantiated - and it is a positive claim. The tenth claim is in fact a pious hope. Larkin & Reif (1979) have designed instruction based on their models of expert physics problem solvers but the effects of the instruction have not been extensively tested.

The Cyclops study will be developed to investigate how best instruction can be designed to improve the performance of novice physics problem solvers. Many of the claims discussed above while well substantiated don't seem to provide many clues about how to do this. Correcting erroneous ideas about basic physics concepts is highly relevant and may even be related to the question of physical intuition and rich internal representation. (Reif & Heller 1982). Also important are questions of strategy checking back etc. To proceed further models must be built which reflect the features of novice problem solving which instruction would be designed to remedy.

Three options for this modelling are possible.

- construct models based on the means ends knowledge development distinction (e.g. Larkin & Simon, 1981)
- take an expert system and alter it to generate the types of errors which students make (e.g. Priest 1979)
- construct models based on the notion of a direct translation model of physics problem solving.

The first option is one which has already been explored. Larkin, McDermott, Simon and Simon (1980) describe two related models - the knowledge development model which simulates expert behaviour and the Means End model for novices. These are a development of the Simon and Simon working forward and backward models which solve dynamics (as well as kinematic) problems and are more elaborated to simulate behaviour more closely. The similarities between these two methods are more important than the differences. Both require an overt statement of goals. In the Cyclops study the novice students didn't have goals however. These models seem too sophisticated to ever generate the types of error seen in the study.

A similar objection can be raised to the second option. Mecho is a program written in Prolog which solves a wide range of mechanics problems from statements in English (Bundy 1979). Both Mecho and also Isaac (Novak 1976) could in principle be altered to generate the types of errors described above (Priest 1979). However the behaviour of these novices seem much too inexpert for that to seem psychologically valid.

We propose to take a direct translation program like STUDENT (Bobrow 1968) which operates in the domain of algebra story problems and alter it to handle these limited physics problems. Students in the Cyclops study confused velocity with acceleration, treated any quantities in the problem almost as being completely inter-changeable. This program would be able to generate such errors and account for many of the errors observed in the study. If such a model could generate a large proportion of the errors observed, this would provide strong evidence of the need for instruction to correct the misconcepts.

Assuming this activity was successful how could it be used to advantage to design some physics instruction? There are two complementary approaches.

Firstly it is necessary to build confidence. The consistency of strategies observed among novices is in fact a weakness which needs to be corrected. They were probably suffering from a lack of confidence which would allow them to explore alternative methods of solution. They need more opportunities to explore these.

Secondly misconcepts should be corrected. The literature on computer games applied to physics (White 1980) is attractive. These provide a way of combining an aid for the exploration of concepts with a way of flexibly exploring how to solve a problem that might be enjoyable. The modelling activity described above would provide a basis on which the exploration of concepts in the game would be designed.

Conclusion - Many claims about how novices solve problems have been made. By using synchronised audio tape and paper and pencil working records, it has proved possible to investigate more carefully the extent to which some of these claims are true. A stronger test will be to base instructional material directly on the types of misconcepts and affective features associated with this view of novices physics problem solving behaviour.

Acknowledgement - We are grateful to Jon Slack for helpful comments on drafts of this paper and to Andy di Sessa, Paul Feltovich, Jill Larkin, Fred Reif and Richard Young for helpful discussions. Our thanks are also due to Claire Jones for typing this paper.

References

Bobrow, D.G. 'Natural language input for a computer problem solving system' in Semantic Information Processing edited by Marvin Minsky MIT (1968)

Bundy, A. et al Solving mechanics problems using meta level inference in IJCAI - 6 pp 1017-1027 (1979)

Champagne, A.B., Klopfer, L.E., Anderson, J.H., 'Factors influencing the learning of classical mechanics', American Journal of Physics 8, 1074-1975 (1980)

Chi, M. Feltovich, P. & Glaser, R. 'Categorisation and representation in physics problem solving' Cognitive Science 5, 121-152 (1981)

Chi, M. Glaser, R. and Rees E. 'Expertise in problem solving' in R. Sternberg (Eds.) Advances in the Psychology of Human Intelligence Hillsdale, N.J.: Erlbaum

di Sessa, A. Unlearning Aristolelian physics A study of knowledge based learning DSRE, internal report, MIT, Boston (1980)

Gilbert, J.K. Osbourne, R.T. 'Identifying science students concepts: the interview about instances approach' in W.F. Archenhold (Ed.) Cognitive development research in Science and Mathematics, Leeds (1980)

Horowitz, L. 'A study of adults solving algebra word problems' unpublished Ph.D. Thesis MIT (1980)

Larkin, J.H., McDermott, J., Simon, D.P., Simon, H.A., Models of competence in solving physics-problems, Cognitive Science 4, 307-345 (1980)

Larkin, J.A., Reif F. 'Understanding & teaching problem solving in physics' European Journal of Science, Vol. 1, No. 2. 191-203 (1979)

Larkin, K.M., An analysis of adult procedure synthesis in fraction problems ICAI Report No. 14 B.B.N.

Novak, G. Computer understanding of physics problems stated in natural language. Tech. Report TR NL 30 Report of Computer Science Austin, Texas (1976)

O'Shea, T. & Floyd A. 'Recording childrens' mathematical behaviour' in J.A.M. Howe, P.M. Ross (Ed.) Microcomputers in Secondary Education: Issues & Techniques, Kogan Page (1981)

Priest, T. 'A design for an intelligent mechanics tutor' CAL Research Group Technical Report 29 Open University (1981)

Reif, F. Heller, J. 'Knowledge structures in physics' unpublished internal report, SESAME project, University of California (Berkeley) (1981)

Reif, F. & Heller, J. 'Cognitive mechanisms facilitating human problem solving in physics: formulation and assessment of a prescriptive model' unpublished internal report SESAME project University of California (Berkeley) (1982)

Scanlon E. 'Improving problem solving in physics' CAL Research Group Technical Report No. 22, Open University (1981)

Shavelson, R.J., 'Methods for examining representation of a subject matter structure in a students memory' Journal of Research in Science Teaching, 11(3) 231-249 (1974)

Simon, D.P., Simon H.A. 'Individual differences in solving physics problems' in R. Siegler (Ed.) 'Children's thinking: what develops? Hillsdale N.J., Erlbaum (1978)

Trowbridge, D.E. and McDermott L. 'An investigation of student understanding of the concept of velocity in one dimension' American Journal of Physics 48(12) (1980)

White, B. 'Designing computer games to facilitate learning in physics' unpublished Ph.D. thesis, D.S.R.E., MIT (1980)

Fig. 1 Seven types of errors identified in the Cyclops study

1. Confusing the meaning of the various terms used (velocity with acceleration, velocity with speed, speed and acceleration with position, average velocity with instantaneous velocity)
2. Incorrect interpretation of the word 'uniform'
3. Misreading of items in the problem statement
4. Drawing misleading diagrams
5. Incorrectly remembering equations of motion to be used
6. Substituting the wrong values into the equations of motions
7. Misunderstanding the meaning of a variable in an equation

ASSOCIATIVE ENCODING AT SYNAPSES, William B Levy, Center for Cognitive Science, Brown U., and Dept. of Neurosurgery, U. Virginia Med. Sch., Charlottesville, Virginia

The last 10 years has seen publication of several neural models capable of performing concept formation, associative learning and recall, and pattern recognition. At the base of all these models is one or another rule for associative synaptic modification. Thus the exact modification rule seems to distinguish one model from another. Certainly specifying such a rule severely restricts the remaining degrees of freedom left to the modeler.

Our neurophysiological research has concentrated on establishing the existence of at least one such "synaptic learning rule" and, further, on specifying the properties of this rule sufficiently so that a differentital equation describing the modification rule could be reasonably proposed.

The simplest form of the equation is

$$\frac{dm_{ij}}{dt} = y_j(cx_i - m_{ij})$$

m_{ij} is the strength of the synapse formed by the afferent i and the postsynaptic cell j; y_j is the net excitation of the j^{th} postsynaptic cell; c is a positive constant; x_i is the frequency of the i^{th} afferent which by definition is nonnegative. The exact form of y is not known though it does appear to be a nonnegative function that increases with postsynaptic excitation and decreases with postsynaptic inhibition. Often y is assumed to be a linear function of synaptic excitation.

By performing the indicated multiplication it is seen that the term ycx corresponds to Hebb's predicted encoding of correlated pre- and postsynaptic activity. The other term (-ym) is needed to account for the erasable aspect of these synapses. With the linear assumption for the size of y, the equation predicts a globally asymptotically stable solution in which the value of the synapses on a cell go to the dominant eigenvector of the autocorrelation matrix of the inputs.

The initial discorvery of long term potentiation by Bliss and Lømo provides the first clear neurophysiological evidence for a cellular analog of memory storage. Today this experimental model is an even better analog since long term potentiation in the dentate gyrus of the hippocampus is known to be an associative phenomenon dependent upon the correlated activity of convergent excitatory afferents. The combined co-activity of a presynaptic input and sufficient synaptic excitation of a postsynaptic cell produces an increase of the synaptic strength of the particular synapses involved in this co-activity. Moreover, this potentiation is accompanied, at other converging synapses, by the phenomenon of long term depression an erasure-like process that decreases synaptic strength. Those synapses which are convergent to an activated postsynaptic structure but which are themselves inactive during the postsynaptic activity lose synaptic strength.

The experiments are performed using anesthetized rats. The response studied is the extracellularly recorded monosynaptic response elicited when the entorhinal cortex is stimulated and the recording electrode is in the dentate gyrus of the hippocampus. Both a synaptic waveform and, should enough synapses be active, cell firing are measured. It is the synaptic response which corresponds to the synaptic strength of the differential equation. This synaptic response takes place almost immediately after stimulation of the entorhinal cortex so there is no time for disynaptic circuitry to confuse the interpretation of the response we measure.

Critical to these experiments is the fact that both the left and right entorhinal cortices project to both the left and right dentates. This arrangement allows the insertion of two quite distant and independent stimulating electrodes. Thus one electrode is used to activate a small number of synapses which generate our dependent measure. The other stimulating electrode is used to control a very large number of converging excitatory synapses. Stimulation with this second electrode quite effectively fires the postsynaptic granule cells in the dentate. In most situations the test electrode does not activate enough synapses to fire these cells.

"Conditioning" stimulation consists of brief, high frequency trains of duration and frequency within the range that has been observed in the entorhinal cortex of behaving rats.

The initial important observation is that high frequency conditioning stimulation through the test system alone does not alter the test system itself. However, when high frequency conditioning of the test system is paired with high frequency stimulation at the other electrode (which is able to produce a powerful postsynaptic response), then long term potentiation obtains in the test system. That is, paired conditioning through both stimulating electrodes produces an increased synaptic response when the the synaptic response of the weak test system is measured alone. Importantly, high frequency conditioning of the powerful system alone depresses the size of the synaptic response of the weak test system even though the powerful system through which the conditioning stimulation is delivered is itself potentiated.

These experiments, then, show that the powerful synaptic activation is permissive for change while the exact type of change that occurs is a function of the actual activity at each particular synapse.

Although we cannot stimulate and record from a single synapse, the conclusions can be advanced and defined by using logical arguements and the natural advantages of the entorhinal-dentate system. In particular it should be realized that because of the totally bilateral nature of this system there are four responses that can be measured when recording and stimulating bilaterally. In fact the synapses of one weak pathway are totally intermingled with the synapses of the strong pathway which provides the permissive stimulus and in addition are themselves collaterals of the strong pathway terminating in the other dentate. From experiments as described above that take advantage of these facts we draw three conclusions.

1. Long term depression occurs at a synapse that is surrounded by many other synapses that have simultaneously undergone long term potentiation. Calculations show that one such depressed synapse centered within a cubic micrometer is surrounded by 20 synapses that potentiated.
2. Potentiation and depression can be differentially induced at different synapses of the same granule cell. This is deduced from experiments in which electrophysiological convergence is well demonstrated.
3. Potentiation, depression, or no change can occur simultaneuously at sister synapses of one individual afferent.

Such conclusions lead to the further conclusion that individual synapses are individually modu-

lated. In fact by extrapolation we argue that such individual modulation has practically been demonstrated. For no matter how small the weak response, so long as it is measureable, it can be potentiated by the proper paired conditioning.

Concluding that long term potentiation requires convergent co-activity gives rise to several questions including "What is the meaning of co-activity?" Varying the relative time that the conditioning stimulations are delivered through each of the two stimulating electrodes, produces a quantitative definition of co-activity. Using conditioning trains of 17.5 msec duration, simultaneous (±0.5msec) conditioning through the two stimulating electrodes produces the most potentiation of the weak test system. If conditioning of the weak test system follows immediately, or later, conditioning of the powerful input, then the test system is depressed. If the weak system is conditioned and then with a delay of 100 msec or more the strong system is conditioned, the test system is depressed. However, if the weak system is conditioned and within 20 msec of the end of this conditioning train the powerful system is conditioned, then the weak test system is potentiated. Thus co-activity is well-defined to a 37.5 (20+17.5) msec window. It might be seen that the temporal requirements have some qualitative resemblance to classical conditioning. However the result places a very specific contraint on neural models of associative learning. In particular association of external events separated by all but the shortest time requires the use of circuitry that performs as a delay line.

One final issue concerns the unit of postsynaptic integration which makes a decision about the sufficiency of converging stimulation and then goes on to permit synaptic modification. Rather than the cell body as Hebb proposes, our current evidence indicates that individual dendritic domains or branches can function independently. By taking further advantage of entorhinal-dentate anatomy, it is possible to activate synapses on different parts of the granule cell dendrites in a controlled and specific manner. When this is done with minimally sufficient postsynaptic responses, we find that it is possible to independently potentiate or depress synapses in one of the two dendritic domains. At high intensities, however, the dendritic domains show an interaction for potentiation.

If this independence is the normal functioning mode, then this adds substantial complexity to models of the nervous system, perhaps increasing the number of functional units ten-fold.

REFERENCES

Bliss, T.V.P. and Lømo, T., J. Physiol. 232, 331-356, 1973.

Levy, W.B., Brassel, S.E., and Moore, S.D., Neuroscience, 1982, in press.

Levy, W.B. and Steward, O, Brain Res. 175, 233-245, 1979.

Levy, W.B. and Steward, O. Neuroscience, 1982, in press.

McNaughton, B.L., Douglas, R.M. and Goddard, G.V., Brain Res. 157, 277-293, 1978.

Wilson, R.C., Levy, W.B. and Steward, O. Brain Res. 176, 65-78, 1979.

Wilson, R.C., Levy, W.B., and Steward, O., J. Neurophysiol. 46, 339-355, 1981.

NEURAL HARDWARE AND THE PRESUMED AUTONOMY OF PSYCHOLOGY

William Bechtel
Bernard Ecanow
University of Illinois Medical Center

Two types of arguments are commonly given in support of the claim that cognitive psychology can predict and explain cognitive processes without troubling itself with the details of neurophysiology. The justified conclusion of these arguments is often thought to be that artificial intelligence research, which tries to model human thought on electronic hardware, "can be regarded as psychology in a particularly pure and abstract form [since] the same fundamental parameters are under direct experimental control (in the programming), without any messy physiology or ethics to get in the way" (Haugeland, 1981, p. 31). This paper will challenge the validity of both arguments for this claim and propose how features of neurological hardware may have consequences for the performance of human cognition.

The first argument for the autonomy of psychology originated with Putnam (1975) and has been developed most extensively by Fodor (1974 and 1979). Putnam noted that in the case of computers the same programme can be run on very different types of hardware. Fodor extended this argument by noting that the same hardware can run alternative programmes. Thus, reduction of programme states to hardware states or of psychological states to neurophysiological ones is impossible. Psychology must thus remain a "special science" seeking its own explanatory scheme.

The second argument for the autonomy of psychology is also designed to establish the additional claim that programming computers is a particularly apt way to learn about human cognitive performance. This argument starts with the assumption that all the information humans can employ in their cognitive operations must cross their sensory thresholds and be encoded within them. Since it is only this encoded information that the mind-brain can employ in its information processing, Dennett describes the mind-brain as a "syntactic engine" Dennett, 1981). This argument then construes thought processes as formal processes in which one manipulates the symbols in which the information is encoded. Assuming that the mind-brain has an effective procedure for these formal processes, Church's thesis claims that there is a recursive process for computing it. Each formal process can therefore be computed by a Turing machine. Invoking the concept of a universal Turing machine (i.e., one that can imitate every specific Turing machine), the argument concludes that thought processes can be modelled on a universal Turing machine. Psychology can direct itself to producing computer or Turing machine models that replicate human thought and not concern itself with neurophysiology.

As enticing as these arguments make the prospects of an autonomous psychology seem, they are seriously flawed. As Richardson (1979) has argued, even if the mapping between neurophysiological states and psychological states is many-many, that does not eliminate the possibility of an informative reduction of psychology to neurophysiology. All that is required are neurological conditions that are sufficient for determining the psychological states. Moreover, if the argument Putnam and Fodor use against the explanatory relevance of neurophysiology works, it also undercuts the simple appeal to programming models to explain cognitive functions. Just as the same programme can be run on different hardware, different programmes can account for the same behavior. Therefore, even if a programme perfectly minics human behavior, one has no assurance that it actually describes how humans manipulate symbols (cf. Bechtel, forthcoming).

The second argument for the autonomy of psychology is just as flawed. This argument moved from claiming that symbol manipulation can be modeled on a universal Turing machine to using actual computer programmes to model that process. Haugeland notes what is assumed in making that move: "with one qualification, universal machines can be built, that is what digital computers are. This one qualification is that a true universal machine must have unlimited storage, whereas any actual machine will have only a certain fixed amount" (Haugeland, 1981, p. 13). That qualification, however, has very far reaching consequences. Neither our brains nor digital computers come close to having the unlimited resources required by a universal Turing machines. With limited resources, however, neither brains nor computers can employ the kinds of algorithms that Church's thesis assures us exist for all decidable processes. So the use of Church's thesis and the concept of a universal Turing machine to justify using computer similation as a way of studying human psychology is unjustified.

Neither of these responses to the arguments for the autonomy of psychology from neurophysiology shows that psychology might not profitably be pursued in this autonomous manner or that computer modelling might not be the most powerful means of doing that. But they do undermine the assumption that artificial intelligence models provide an adequate basis for understanding human cognition. While not denying that such models can show us interesting features about cognition, we shall now argue that there is reason to believe that significant differences exist between human cognition and computer models of it.

Limited resource capacity for problem solving dictates that one cannot always use procedures that guarantee correct results. For complex problems one must choose methods that yield correct results much of the time but are fallible. There are two fallible ways of using limited resources for dealing with problems whose optimal solution requires greater resources. One that has been studied much in recent years has been the use of heuristics (Cf. Simon, 1969). Heuristics are rules that are simpler than optimal algorithms, produce the same answers as the optimal algorithm much of the time, but that are subject to systematic errors because of the simplifications they use (Wimsatt, 1980). Tversky and Kahneman (1974) have developed an empirical research programme to discover the kinds of heuristics humans use in handling certain kinds of judgment tasks. A second way of solving the problem of limited resources is to manipulate the hardware of one's system to approximate the performance of a richer hardware system. As in the case of heuristics, a simplified hardware system that is developed to approximate a richer one may allow one to reach

correct answers much of the time, but will do so at the cost of making errors under certain conditions.

The hardware system of a Turing machine or a computer is linear and digital--information is processed by linearly transmitting and modifying information units which are perfectly distinct and so engender no ambiguity. One basis for the analogy between brains and computers is the assumption that the brain also utilizes a linear and digital processing mechanism--the neuron (von Neumann, 1958). Like the components of computers, neurons transmit electrical impulses linearly down their dendrites and axons with the action potential in the axon being comparable to a digital binary signal in a computer. (Dendritic processes allow for a spectrum of responses, but these functions have been viewed as weighting and gating functions, which are easily replicated in computer hardware.)

In addition to these neuronal processes, which seem comparable to those realized in a Turing machine or computer, though, there is another transmission mechanism in the brain. This mechanism involves a form of transmission quite different than the linear and digital transmission of neurons and may provide a means for the brain to approximate the performance that would require a far richer linear and digital mechanism. One can best appreciate this mechanism by considering earlier stages in evolution.

Long ago Hughlings Jackson (1884) insisted that to understand the function of the brain one had to attend to its evolution. The brain is organized in an evolutionary hierarchy in which the lowest and first evolved parts of the brain regulate all bodily activities. The later evolved higher centers function by modifying and regulating the earlier evolved lower centers. Before there were nerve cells, however, there existed a mechanism for transmission between cells. According to Oparin's (1965) model, cells originated when water interacted with macromolecules and electrolytes to form a more fat like substance--protoplasm. The water around the molecules becomes structured in much the same manner as occurs in jello and the whole unit behave like an oil drop with respect to the intercellular plasma. Ecanow (1982) has proposed that the same process is responsible for forming multicellular aggregates. In these aggregates different thermodynamic states are found in the cytoplasm of the various cells (including a different state in the cellular interface or membrane) and in the interstitual fluids.

Already within these early cellular aggregates a mode of signal transmission existed. The different thermodynamic states of the cytoplasm, membrane, and interstitual fluids are in dynamic equilibrium with one another, with a constant exchange of molecular substances occurring between the different structural units. This exchange allows for a kind of transmission between cells: a change in the thermodynamic conditions in one cell will propagate rapidly to surrounding cells. This kind of transmission still occurs even after nerve cells have evolved with their more digitalized and linearly directed transmission capacities. This is particularly true in places where nerve cells are tightly bound together. This tightly organized pattern causes the water in the plasma surrounding the cells to become highly structured itself, affecting, in particular, the solubility of ions in the plasma. Both the cells and the surrounding plasma become highly susceptible to any thermodynamic changes that are induced. One of the prime causes of thermodynamic changes is electrical activity propagated along neurons. Electrical energy alters the physical-chemical structure around the nerve cell. Once the change has occurred, the physical-chemical organization elsewhere will modify until equilibrium is once again achieved. Not only are these physical-chemical changes initiated by neural activity, they also reciprocally affect that activity. Neural activity depends on ion transfer, and this ion transfer is governed by the degree of structuring found at the cell-plasma interface. One cell's firing changes this structuring around other cells and hence their potential to fire.

There is, at this point, reason to believe this physical-chemical transmission mechanism is efficacious in humans. Since most anesthetic agents are biochemically inert, it is generally recognized that a physical-chemical mechanism must be involved. Following a suggestion from Bernard (1875), Ecanow et. al. (1979 and Ecanow, 1981) propose that anesthesia involves the formation of a highly structured matrix at the cell surface which becomes non-polar and fat-like. Ion exchange is a polar process and so is blocked in such a matrix. This model predicts that substances which decrease the structuring of water, generally polar molecules, chaotropic ions like urea and vitamin C, or increased temperatures, will produce an increase in mental activity. These effects have been observed in vivo. The insight of this model has been extended to account for the fluctuation between increased and reduced mental activity found in manic-depressive patients (Ecanow and Klawans, 1974).

This physical-chemical mode of transmission proposed by Ecanow (1982) differs from neural transmission in propagating three dimensionally from the initial site and in invoking a degree of response that can vary over a continuous spectrum. It is also very rapid. We cannot, at this point, make definitive claims as to its direct role in cognition, but we conclude with a speculative suggestion. Kandel (1978) has found that long term and short term habituation and sensitization in Aplysia (processes he takes to be forms of memory and learning) result from changing the amounts of calcuim ions (needed for transmitter release) available at the pre-synaptic cleft. Kandel does not account for the change in calcium availability that habituation and sensitization produce, but one possible mechanism would be through alternation of the physical-chemical structures near the pre-synaptic cleft. Such structuring can occur in degrees and so account for the gradual "learning" of these responses. Moreover, such structures would be appropriately subject to change when new experiences produce neural activity in the area around the pre-synaptic cleft.

The physical-chemical transmission mechanism provides the mind-brain with capacities for information processing quite different from the linear and digital capacities of neurons. Given the hardware limitations of the brain, it may well be that this three dimensional analogue mechanism of physical-chemical transmission provides the mind-brain a powerful tool for overcoming resource constraints. The power of this mechanism, however, cannot be studied by modelling with digital computers that lack such transmission capacities.

REFERENCES

Bechtel, William (forthcoming), "Two Common Error in Explaining Biological and Psychological Phenomena," Philosophy of Science.

Bernard, Claude (1875), Lecons sur les Anesthetiques. Paris: Bailliere.

Dennett, Daniel C. (1981), "Three Kinds of Intentional Psychology." In R. A. Healey (ed.) Reduction, Time and Reality: Studies in the Philosophy of the Natural Sciences. Cambridge: Cambridge University Press.

Ecanow, Bernard (1981), "A Comprehensive Theory of Anesthesia." Physiological Chemistry and Physics Journal 13: 23-27.

Ecanow, Bernard (1982), "Interstitial Conduction and the Emergent Mind." Journal of Pharmaceutical Sciences 71: viii.

Ecanow, Bernard and Klawans, H. L. (1974), "Physical-Chemical Properties of Cellular Constituents and their Contribution to Neuronal Function," in H. L. Klawans, (ed.) Models of Human Neurological Diseases. Amsterdam, Excerpta Medica.

Ecanow, Bernard, Gold, B. H., and Ecanow, C. S. (1979), "Unified Theory of Anesthesia," Journal of Pharmaceutical Sciences 68: iv-v.

Fodor, Jerold (1974), "Special Sciences," Synthese 28: 97-115.

Fodor, Jerold (1979), The Language of Thought, Cambridge: Harvard University Press.

Haugeland, John (1981), "Semantic Engines: An Introduction to Mind Design." In J. Haugeland, (ed.) Mind Design. Montgomery, VT: Bradford Books.

Jackson, J. Hughlings (1884), The Croonian Lectures on the Evolution and Dissolution of the Nervous System, London.

Kandel, Eric (1978), "Small Systems of Neurons," Scientific American 238: 66-76.

Oparin, A. I. (1965), "The Pathways of the Primary Development of Metabolism and Artificial Modeling of this Development in Coacervate Drugs." In S. W. Fox (ed.) Origins of Prebiological Systems and of their Molecular Matrices. New York: Academic.

Putnam, Hilary (1975), Mind, Language, and Reality: Philosophical Papers, Volume 2. Cambridge: Cambridge University Press.

Richardson, Robert C. (1979), "Functionalism and Reductionism," Philosophy of Science 46: 533-558.

Simon, Herbert A. (1969), The Science of the Artificial, Cambridge: M.I.T. Press.

Tversky, Amos and Kahneman, Daniel (1974), "Judgment Under Uncertainty: Heuristics and Biases." Science 185: 1124-1131.

von Neumann, John (1958), The Computer and the Brain, New Haven: Yale University Press.

Wimsatt, William C. (1980), "Reductionistic Research Strategies and Their Biases in the Units of Selection Controversy," in T. Nickles (ed.), Scientific Discovery: Case Studies, Dordrecht: Reidel.

The Integrated Implementation of Imaginal and Propositional Data Structures in the Brain

John Barnden
Computer Science Department
Indiana University, Bloomington, Indiana

1. Introduction

I sketch a speculative model (to be presented in greater detail in [1]) of the human brain's implementation of the temporary data structures appearing in cognition. I assume the following working hypothesis:

> The Representation Hypothesis. Much of human cognition is to be explained as the manipulation of data structures, in as literal a sense as the sense in which computers manipulate data structures.

The model is not committed to any particular data structure 'language' in the brain, but it leads to interesting suggestions concerning such languages. The model unifies 'propositional representation', 'imagery' [2] and perception.

A background assumption I make is that the temporary data structures in the brain are physically implemented as short-lived patterns of 'neural enhancement'. These patterns can cause particular events (e.g. changes in the patterns) to occur in the brain. The vague term 'neural enhancement' is intended to encompass possibilities such as higher than normal pulse activity (cf Hebb [3]) and disturbed dendritic-potential microstructure (cf Pribram [4]). To avoid making unnecessary hardware commitments, however, I cast the model at a higher level of description which is intended still to allow relatively easy mappings down to the hardware level.

2. The Main Ideas of the Model

It will become clear that the model postulates sharing of the mechanisms used in perception and those used in the implementation of temporary data structures. For the purposes of this paper, let us simplify matters by taking monocular vision to be the only sense. (A fuller account will be given in [1].) The following hypothesis is a proposal about visual mechanism, bearing family resemblances to proposals such as Marr's primal sketches [11]. Again for simplicity, we assume the retina can be considered to be a 2D rectangular array of (possibly overlapping) receptive fields (finite in number).

> The Vision Hypothesis.
>
> a) The brain contains a number of permanent abstract entities called 'perceptual pattern matrices' (PPMs). Each PPM is a 2-dimensional rectangular array isomorphic to the retinal array of receptive fields. There is a set of 'enhancement types', and at any moment each element of each PPM has a 'degree of enhancement' for each type. The 'state' of a PPM is the current pattern of degrees of enhancement over the PPM. Retinal stimulation is converted by low-level preprocessing into a state of some PPM. The enhancement degrees at an element in the PPM for some enhancement types encode the presence of features in the element's corresponding receptive field. Examples of such features are line segments, edges, corners, textures, colours, etc.
>
> b) The possession of more than one PPM allows the brain to maintain very short term iconic memory (cf [12]) of retinal input, and to integrate succesive views.

Now it has been suggested that (conscious or unconscious) visual imagery is based on states of retina-like data structures (e.g. Kosslyn [6]). It has also been mooted that the mechanisms used in visual imagery are shared with visual-perception mechanisms. Suppose we adopt these suggestions, in the sense of allowing the internal generation and manipulation of states of PPMs. If we closely followed the examples used by Kosslyn and others, the PPM states so manipulated would be spatial-analogue images, i.e. would picture physical objects, crude maps, etc. I now claim that, assuming the brain can internally generate such images, there is a priori no reason to think that the PPM states it can generate are restricted to be such images. For instance, there is no reason to think that the brain cannot generate ('pictures' of) written words, abstract diagrams (perhaps depicting abstract network structures), or other symbolic shapes of non-pictorial, non-lexical form. (Once generated, the presence of such PPM patterns is no more bizarre than if the patterns had resulted from seeing words, diagrams, etc.) These observations suggested to me the central postulate of the model:-

> The Main Hypothesis.
>
> a) Any temporary data structure considered to reside at some moment in the brain is implemented as (part of) a state of a single PPM or as a vector of (partial) states of several PPMs.
>
> b) There exist processes which examine PPM states and can, if they detect suitable subpatterns, cause PPM state changes. These processes together with the PPMs are regarded as a production system [5], with pos-

sible concurrent firing of productions. This production system constitutes the entire machinery the brain has for the internal manipulation of temporary data structures.

c) One enhancement type is called 'attention'. Elements with higher degrees of enhancement of attention receive preferential treatment by PPM manipulation processes. A locus of high attention values in a PPM can be slid around in a PPM to effect scanning.

d) The response by pattern-detection processes to PPM patterns is spatially continuous in the sense that the effect of 'spatial' deformations of patterns can be made arbitrarily small by making the deformations sufficiently small.

e) To a first approximation, the effect of the presence of a pattern in a PPM is independent of the identity of the PPM.

f) If approximately the same subpattern is simultaneously present within two different PPMs, and the attention enhancement of the elements used by the subpattern in at least one of the PPMs is sufficiently high, then the attention enhancement of both pattern instances can become boosted. Thus there may be implicit associations among PPM states. (No direct 'pointers' between PPMs are proposed.)

g) There may exist considerably more PPMs than are required by the Vision Hypothesis (for the purpose of receiving preprocessed retinal stimulation, maintaining iconic memory and integrating views).

h) The issue of consciousness is not addressed by the model. There is no assumption that the brain is conscious of any of the PPM states existing at a given time. There is no assumption that when the brain is conscious of a visual image it is conscious of a single PPM state.

It is sometimes suggested that a neural enhancement pattern might be some form of node/link structure representing propositional information. Lifting this idea to the abstract level of PPMs, it is quite conceivable that abstract, propositional information is represented in the form of diagrammed nets. That is, nodes are localized groups of contiguous enhanced PPM elements, and links are chains or ribbons of such elements. (It is not, however, suggested that net patterns are particularly close to the precise net diagrams to be found in the literature, e.g. [7].) Nodes and links in a net-like PPM state can be considered to be associated to long-term knowledge by virtue of labels they are adjacent to, in that the labels are subpatterns which can be detected by some productions (see Main Hypothesis, part (b)). For example, a node label might be a special pattern which has (for us as theorists) the meaning 'dog': by virtue of suitable productions detecting the subpattern, the brain would take actions consistent with the node's representing a dog. It is worth noting that the 'dog' label could be either a stylized picture of a dog or the word 'dog' itself! It could, however, be a subpattern of non-lexical non-pictorial form.

The basic actions in the productions of Main Hypothesis part (b) include: movement of subpatterns within and between PPMs, deletion and creation of subpatterns, changes of enhancement degrees (especially of attention), etc. The action part of a production is tentatively proposed to have a simple sequential form. The productions are thought of as constituting LTM. The model allows, as a detail of this LTM, the existence of a long-term store of encoded PPM states: these can be decoded and read into PPMs, and can be encoded from the contents of PPMs.

Some detection of subpatterns must be primitive in that it is achieved without the need to examine other data structures. I propose that, at least, some simple geometrical shapes, some stylized pictures, some words, and some specialized non-pictorial non-lexical graphic items (including nodes and links) can be primitively detected. (Much of this ability would arise form maturation and experience.) But non-primitive forms of detection can be proposed. For example, by sliding a locus of high attention enhancement around in a PPM, a detection process (perhaps itself made up of production firings) could check for the presence of a piece of network by tracing it out. Also, the associative mechanism of Main Hypothesis part (f) allows the matching of two (not necessarily primitively detectable) subpatterns in distinct PPMs, where one of the subpatterns might be taken to be a template (of pictorial, network, orthographic or any other form). Note that the PPM production system can construct transformed versions of patterns to facilitate further processing. For instance, in the course of visual perception an abstract net representation of a scene could be constructed from a picture of it in a PPM.

3. Selected Implications

The model unifies unconscious spatial imagery and propositional representation at the same time as providing an (intermediate level) implementation of propositional representation. A particular consequence of the Main Hypothesis is that abstract symbolic representations, spatial-analogue images constructed in visual imagery, and images resulting directly from retinal stimulation are just special cases of PPM states. (A more popular route to unification - annotating propositional structures with spatial information [8] - does not address the

issue of implementing propositional structures.) The model can incorporate, in a natural way, hybrid forms of symbolism such as are found in, for instance, maps, cartoons (especially those which include words), road signs, and many forms of semi-abstract sketch and diagram. Moreover, the internal presence of such hybrid symbolism may be closely related to the fact that we deal with it externally with such naturalness, ease and frequency.

The model may help to explain how the human capacity for abstract cognition evolved. That is, assuming that at some stage of primate evolution the Vision Hypothesis held and spatial-analogue PPM states could be internally generated and manipulated, it is plausible that the necessary pattern detection and manipulation operations could have evolved into a form which could deal with more abstract PPM states. (See Minsky [9], Section 6.5.4, for another proposal in which abstract symbolic manipulation evolves from perceptual operations.)

I am just embarking on a computer simulation of a simplified, precise version of the model. This paper has only sketched a 'model schema' in which many parameters (e.g. number and size of PPMs) remain unspecified. The first stage in the project is the exercise of developing a diagrammatic version of a simple production system derived from PSG [10].

Acknowledgments

I am grateful for suggestions and criticism from B. Chandrasekaran, G. Clossman, R.E. Filman, D.R. Hofstadter, M.J. Intons-Peterson, S. Kwasny, J.T. O'Donnell and D. Robinson.

References

[1] Barnden, J.A. 'Imaginal Symbolism'. In preparation for journal submission.

[2] Block, N.J. (ed.). Imagery. MIT Press, 1981.

[3] Hebb, D.O. Organization of Behaviour. Wiley, 1949.

[4] Pribram, K.H. Languages of the Brain. Prentice-Hall, 1971.

[5] Waterman, D.A. and Hayes-Roth, F. Pattern-Directed Inference Systems. Academic Press, 1978.

[6] Kosslyn, S.M. Image and Mind. Harvard University Press, 1980.

[7] Findler, N.V. Associative Networks. Academic Press, 1979.

[8] Waltz, D.L. 'Toward a Detailed Model of Processing for Language Describing the Physical World'. IJCAI-7, 1981.

[9] Minsky, M. 'A Framework for Representing Knowledge.' In Winston, P.H. (ed.), The Psychology of Computer Vision. McGraw-Hill, 1975.

[10] Newell, A. 'Production Systems: Models Of Control Structures'. In Chase, W.G. (ed.), Visual Information Processing. Academic Press, 1973.

[11] Marr, D. 'Early Processing of Visual Information'. Phil. Trans. Roy. Soc. London, Series B, 275, No. 942, 1976.

[12] Brown, R. and Herrnstein, R.J. 'Icons and Images'. In [2].

PROGRAMMERS' MENTAL MODELS OF THEIR PROGRAMMING TASKS: THE INTERACTION OF REAL-WORLD KNOWLEDGE AND PROGRAMMING KNOWLEDGE

Hank Kahney & Marc Eisenstadt
The Open University
Milton Keynes, England

INTRODUCTION

This paper describes our ongoing research into the behaviour of novice programmers. We are interested in the mental processes which occur when novices are confronted with a problem statement, and the mechanisms by which they understand the problem, design an algorithm, code it, and (if necessary) debug it. Our research is a development of earlier work on problem understanding (Hayes & Simon, 1974), models of programmers' coding processes (Brooks, 1977), and debugging (Sussman, 1975; Goldstein, 1975; Laubsch & Eisenstadt, 1981).

We investigate students attempting to write recursive inference programs using a LOGO-like database-manipulation language called SOLO (Eisenstadt, 1978; Eisenstadt, Laubsch, & Kahney, 1981) Students are presented with a prototypical problem and solution couched in everyday terms in order to simplify the explanation of recursion: "Imagine a chain of 'KISSES' relations, e.g. JOHN KISSES MARY KISSES FRED KISSES JANE, etc. A procedure called INFECT can propogate FLU all the way through the chain of KISSES relations, so we end up with JOHN HAS FLU, MARY HAS FLU, etc." The example is explained to the students in great detail, including several pages of text, diagrams, and a worked-through trace of a sample invocation of INFECT.

As one might expect, some students 'get it' (i.e. understand this simple form of tail-recursion and the notion of propogating side-effects through the data base), and some don't. The difference between those who 'get it' and those who don't can be accounted for by differences in (a) the abstractions they make from their first detailed example, and (b) the evaluation rules inherent in the mental models they use to 'run through' trial solutions.

A SAMPLE PROBLEM AND SOLUTION

We investigated students solving several recursive inference problems, including one based on a real-world example so compelling that we could be 'certain' the nature of the task was perfectly understood. Here is a concise summary of the problem:

Given a database describing objects piled up on one another as follows:

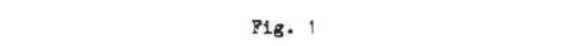


Fig. 1

write a program which simulates the effect of someone firing a very powerful pistol aimed downwards at the topmost object (SANDWICH), yielding the final database shown below:

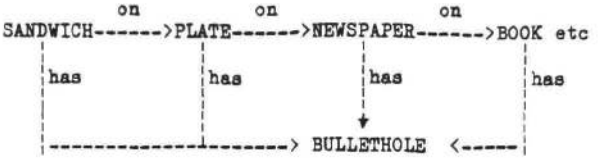


Fig. 2

As it turns out, even our 'crystal clear' example (fleshed out in considerably more detail) causes difficulty-- it appears that those students who 'get it' can cope with either 'crystal clear' or 'muddy' recursive inference problems, whereas those who don't are stuck in either case.

Fig. 3 below shows the solution eventually produced by subject S8, one of the subjects who 'got it':

```
TO SHOOT /X/
 1 NOTE /X/ HAS BULLETHOLE
 2 CHECK /X/ ON ?
  2A If present: SHOOT *; EXIT
  2B If  absent: EXIT
```

Fig. 3 S8's solution (the '*' and '?' are co-referential)

Below is a summary of the protocol of subject S8 during the course of reading and solving this problem, but before any attempt to write the code shown in Fig. 3. Problem statements are underlined. The numbers are segments from the actual protocol. It has been condensed for expository purposes in this brief paper, but captures the highlights of the protocol. A complete version is described in Kahney (1982).

"On page 80 of Units 3 to 4 we looked at a method for making a particular inference 'keep on happening'."

> 2 Is that called 'iteration'? No, 'recursion'... I think this is going to say something about what happens when you keep on applying a function...through a database

"In this option you are asked to imagine a state of the world in which there are six objects: ... this hypothetical world is highly structured: the sandwich is lying in the centre of the plate, which is sitting on the newspaper, which is lying on the book ..."

> 4 ... well you could also get out things like... sort of making inferences about 'if the sandwich is on the plate which is on the newspaper [then] the sandwich is on the newspaper'.

"A database representing this state of affairs looks like this [Fig. 1]. Now imagine someone standing beside the table with a .357 magnum pistol."

8 Well, I would expect him to shoot through all that lot then. I don't know why he wants to do it though...

S8 began with a working knowledge of recursive procedures. At successive sentences S8 set up expectations about what would come next and usually was in the position of predicting the information contained in the next sentence or two: she was always just slightly ahead of the game. S8 is apparently using a 'recursion' schema to direct her attention during the reading process to important aspects of the problem statement. The first line of the problem statement has clearly triggered off an expectation of recursion [protocol segment 2], with a concomitant expectation of some 'function' to be applied 'through a database' [segment 2]. The database structure [Fig. 1] is consistent with her expectation of a standard transitivity problem [segment 4], even though this is not the problem to be posed. Her real-world knowledge about pistols and the spatial relationship of the objects in the problem combines with her expectations about transitivity problems to yield an expectation about what the protagonist in the problem statement will do [segment 8]. This expectation does not mesh with her knowledge of human motivations and intentions [segment 8].

Figure 4 depicts our representation of S8's internalized schema for recursion. The details of the schema are derived from a variety of sources: transcription tasks, concept rating and sorting tasks, problem-solving tasks, and verbal protocols.

RECURSIVE-PROCEDURE

```
 ...
GOAL: (ForEvery x In (my applies-to) do
         (achieve (my action) x))
ACTION: (a side-effect {DEFAULT (a NOTE)})
APPLIES-TO: (a transitive-chain)
SURFACE-TEMPLATE:
   TO (name1 =(a name)) (a parameter {default: X})
   (my action)
   CHECK (a node) (a relation) (a wild-card)
    IF PRESENT: (a procedure
                  with name = name1
                  with parameter = "*");EXIT
    IF  ABSENT: EXIT
   DONE
EVALUATION-RULES:
  1) (let parameter = the startnode from
                      (my applies-to))
  2) (apply (my action) parameter)
  3) (assert `(ACHIEVED ,(my action) ,parameter))
  4) (let parameter = (GetNextNode))
  5) (ForEvery x In (GetRestOfNodes)
        (assert `(ACHIEVED ,(my action) ,x))
TRIGGERS: "keep on happening"; re-apply
```

Figure 4: S8's schema for recursion

Bearing in mind that slot-names are displayed against the left-hand margin (e.g. GOAL, ACTION, etc.), and that the function "my" is a cross-reference to a slot-filler (e.g. (my action)) we can paraphrase S8's schema for recursion as follows:

The GOAL of a recursive procedure is to perpetrate a side effect on every element of the data structure to which it is applied, i.e. a 'transitive' chain. (Knowledge about such structures is contained in S8's TRANSITIVE-CHAIN schema, not depicted here, which indicates that a collection of nodes standing in a particular relation to one another is an essential component of recursive processing-- S8 has abstracted this notion, although KISSES is not a transitive relation.) The ACTION involved is typically the application of a NOTE primitive (which performs a database 'ASSERT'). The SURFACE-TEMPLATE depicts raw SOLO code, with its own slots to be filled in during actual coding. It is based upon an exempla given in the textbook, and corresponds to rote learning of 'how to do it', rather than understanding of 'how it works'. (Subjects like S5, discussed below, have a poorer grasp of recursion and need only have a mental pointer to a place in the textbook where they can find a typica example to copy.)

'How it works' understanding is reflected primarily in the GOAL and EVALUATION-RULES slots. The GOAL slot captures the essence of the 'generator plan' used in the program-understanding plan-libraries of Waters (1978) and Laubsch & Eisenstadt (1981). The EVALUATION-RULES slot depicts S8's technique for working through a mental model of the succession of effects carried out by a body of SOLO code. The rules are clearly not sufficient to work as a SOLO interpreter, but rather depict the subject's own naive strategy for convincing herself that the code 'works'. The rules behave as follows: (1) instantiate the parameter, pretending that it's the first node in the chain (i.e. SANDWICH); (2) imagine the main action being performed on that node; (3) make a mental note that the action has been achieved; (4) see what node is next in the database, traversing the crucial 'transitive' relation; (5) make a mental note that the action is achieved on every node reachable along the 'transitive' chain.

Below we present S8's protocol corresponding to the above evaluation rules, along with the relevant rule listed in square brackets, e.g. [ER1], [ER2], etc. These protocol segments were recorded after S8 had written the program, but before she ran it.

208 TO SHOOT... X, let's say X is a SANDWICH... [ER1]

210 First of all it NOTEs in the database...X HAS BULLETHOLE [ER2, ER3]

211 It then CHECKs whether X is ON anything... [ER4]

213 X is ON PLATE so it will do that to PLATE... So that should keep doing that, PLATEs on ... something, so on and so on... [ER5]

A SECOND SOLUTION

Here is the solution eventually developed by subject S5, who didn't 'get it':

```
TO SHOOTUP /X/
 1 NOTE /X/  HAS  BULLETHOLE
 2 CHECK /X/  SHOOTS  ?
  2A If Present: SHOOTUP  *; EXIT
  2B If Absent: EXIT
```

Figure 5

Below are extracts from S5's protocol. Whereas S8 was able to develop the solution 'in her head', S5's solution evolved during code-writing:

46 I'm going to follow that example [* INFECT].

51 [Reads from INFECT example in SOLO primer]... NOTE..um, X HAS FLU... SANDWICH HAS BULLETHOLE....

54 SANDWICH ON PLATE, um....NOTE...um...

63 I've got to get the SHOOT in somewhere haven't I?

65 CHECK...X SHOOTS SANDWICH. IF PRESENT....SHOOTUP....

83 Well, I hope it will go all the way through the sequence and shoot the floor. The data base is in and I've copied that program [= INFECT] exactly.

S5 has a recursion schema which differs from that of S8 in several respects. First, S5's schema does not have a filled SURFACE-TEMPLATE slot, but rather (a) a pointer to the place in the SOLO primer where a typical recursive procedure, i.e. INFECT, is described, and (b) a method for filling the SURFACE-TEMPLATE slot by copying the INFECT program's structure and providing arguments from the current problem. Second, S5's schema has a restriction that the relationship between objects in the database must be 'active' for recursion to work. That is, from the original INFECT teaching problem with JOHN KISSES MARY KISSES FRED, S5 had abstracted the rule that a start-node has to 'do' something to a successor-node before a side-effect can be perpetrated on the successor-node. (S8, on the other hand, had abstracted 'transitivity' from previous study of the INFECT program-- neither view, of course, is perfectly correct).

For example, 'ON' is not an 'active' relationship between the objects (SANDWICH, PLATE, etc.) given in the problem statement. 'ON' is passive and thus does not 'support' S5's notion of recursion. 'SHOOTS' is an active relation, and S5 is convinced that somehow SHOOTS must be brought into the pattern-matching segment of the program in order to make the program work at all [segments 63 and 65 of S5's protocol]. This conviction precludes solution of the problem, unless careful re-analysis of the example program leads to reformulation of the rule about relationships between database objects. S5 never relinquishes her belief that an active relationship need exist between the nodes for recursion to work, and her reformulations of the program are all guided by this single important but wrong-headed principle. S5's protocol continues:

156 This one about the BULLETHOLE and this one with the KISSES are different. I need to say that the first X, the first parameter does something actively... to the second parameter. All I've got is BULLETHOLE. In the example it's got KISSES, which is an active thing.

Although S5 made several subsequent attempts to map the BULLETHOLE problem onto the INFECT framework, the point of view from which the mapping occurred never changed and no solution resulted.

CONCLUSION

Because our programming problems use real-world examples rather than abstract programming tasks, the subjects' knowledge of programming interacts with their real-world knowledge during the reading, coding, and debugging processes. We have indicated the way (often imperfect) knowledge of programming concepts pervades problem solving even in its earliest stages.

Our subjects develop schemas for recursion which are more or less 'adequate' for solving the problems we devise. This adequacy ranges from that of subject S5 (who can not solve any of the recursion problems we have devised) to that of subject S8 (who can solve many, but not all, of our recursion problems). When a problem maps onto an adequate set of schemas in a novice's store of knowledge, the novice can tackle the tasks of problem understanding, method finding, coding, and informal verification in a productive and efficient manner. When a problem is mapped to an inadequate set of schemas, the problem statement is often poorly understood, and becomes embedded in a program constructed as much from world knowledge as from the basic elements of the implementation language.

REFERENCES

Brooks, R. Towards a theory of the cognitive processes in computer programming. Int. J. Man-Machine Studies, 9, 1977.

Eisenstadt, M. Artificial intelligence project. Units 3/4 of Cognitive psychology: a third level course. Milton Keynes: Open University Press, 1978.

Eisenstadt, M., Laubsch, J., & Kahney, H. Creating pleasant programming environments for cognitive science students. Proceedings of the Third Annual Cognitive Science Society Conference, Berkeley, California, 1981.

Goldstein, I.P. Summary of MYCROFT: a system for understanding simple picture programs. Artificial Intelligence, 6,, 1975.

Hayes, J.R. & Simon, H.A. Understanding written problem instructions. In Gregg, L.W. (Ed.), Knowledge and Cognition. Hillsdale, N.J.: Lawrence Erlbaum Associates, 1974.

Kahney, H. An in-depth study of the behaviour of novice programmers. Technical Report No. 82-9, Human Cognition Research Group, The Open University, Milton Keynes, England, 1982.

Laubsch, J. & Eisenstadt, M. Domain specific debugging aids for novice programmers. Proceedings of the Seventh International Joint Conference on Artificial Intelligence (IJCAI-81), Vancouver, B.C. Canada, 1981.

Sussman, G.J. A computer model of skill acquisition. New York: American Elsevier,1975.

Waters, R.C. A method for analyzing loop programs. IEEE Transactions on Software Engineering, SE-5:3, 1979.

Natural Problem Solving Strategies and Programming Language Constructs {1}

Jeffrey Bonar
Computer and Information Science Department
University of Massachusetts
Amherst, Massachusetts 01003

1. Introduction

Any interesting computerized task soon involves programming. Experience with statistics packages, word processing, and even microwave ovens shows that we always want our systems to be able to follow a step-by-step specification involving decisions and repeated actions. Even with a very intelligent computerized assistant, we would like to give it detailed instructions at an appropriate level of abstraction.

This ubiquity of programming presents a problem, however. It is widely known that programming, even at a simple level, is a difficult activity to learn. {2} What is it about this cognitive skill that is so difficult? Is it inherent in programming, or directly related to the nature of the programming tools currently used for novices? In this report we will present evidence that current programming languages do not accurately reflect human problem solving strategies developed in a context of step-by-step natural language specification. This evidence was gained by studying novice computer programs collected from their terminal sessions [Bonar et al, 1982], video-taped interviews of novices programming, and written studies focusing on specific aspects of novice programming techniques. {3}

Step-by-step natural language specification provides powerful intuitions for novice programmers using a programming language. We hypothesis that these intuitions take the form of frame-like plans - regular but flexible techniques for specifying how to accomplish a task. Programming knowledge also involves frame-like plans [Soloway et al, 1982] [Waters, 1979]. While an individual programming language plan may have many lexical and syntactic similarities to a corresponding natural language plan, the two plans often have incompatible semantics and pragmatics. Many novice programmer's misconceptions derive directly from these incompatibilities.

In this brief report we will show an example of natural language and programming language plans. Using those plans we will discuss a transcripts of novice programmers using a natural language plans while attempting a programming language problem. We conclude with a brief discussion of the implications of this work.

{1} This work was supported by the National Science Foundation under NSF Grant SED-81-12403. Any opinions, findings, conclusions, or recommendations expressed in this report are those of the author, and do not necessarily reflect the views of the U.S. Government.

{2} Our own conservative estimate from several introductory programming courses is that more than 40% of the conscientious students never really understand the rudiments of programming.

{3} Du Boulay and O'Shea [1981] present an excellent overview of research into how novices learn programming.

2. A Mismatch Between a Natural Language Plan and a Program

Consider the following problem:

> Problem 1: Please write a set of explicit instructions to help a junior clerk collect payroll information for a factory. At the end of the next payday, the clerk will be sitting in front of the factory doors and has permission to look at employee pay checks. The clerk is to produce the average salary for the workers who come out of the door. This average should include only those workers who come out before a supervisor comes out, and should not include the supervisor's salary.

The following natural language specification for this problem, written by one of our subjects, is typical:

1. Identify worker, check name on list, check wages
2. Write it down
3. Wait for next worker, identify next, check name, and so on
4. When super comes out, stop
5. Add number of workers you've written down
6. Add all the wages
7. Divide the wages by the number of workers

There are several natural language specification plans used here. Note how steps 1 through 4 specify a loop: steps 1 to 3 describe the first iteration of the loop, indicating repetition with the phrase "and so on". Step 4 adds a stopping condition, assuming that this condition will act as a "demon", always watching the action of the loop for the exit condition to become true. The specification also assumes "canned procedures" for counting inputs, step 5, and for summing a series of numbers, step 6. Note however, that these two procedures are both denoted with the word "add".

Now focus on the two actions performed in steps 1 and 2. The plan to describe these actions is "get a value (step 1), and process that value (step 2)". This plan is nearly universal in this sort of description. Unfortunately, many programming languages support a far less natural plan: "process the last value, get the next value". To see why this is so, consider a problem analogous to Problem 1 but in a programming language domain:

> Problem 2: Write a program which repeatedly reads in integers until it reads they integer 99999. After seeing

99999, it should print out the correct average. That is, it should not count the final 99999.

In Pascal, a popular novice programming language, the correct solution to Problem 2 is:

```
program Problem_2_Expert;
   var Count, Total, New : integer;
   begin
   Count := 0; Total := 0;
   Read (New);
   while New <> 99999
     do begin
        Count := Count + 1;
    Total := Total + New;
    Read (New)
    end;
   if Count > 0
      then
        Writeln ('Average =',Total/Count)
      else
        Writeln ('No data.')
   end.
```

Notice the peculiar while loop construction. Because a while loop tests only at the top of the loop, it is necessary to have a Read both above the loop and at the bottom of the loop. Within the loop we see the plan "process the last value, read the next value". This plan is part of the knowledge used by experienced Pascal programmers. Do novice programmers easily acquire such a plan? Apparently, no.

First of all, novices want the while to have a demon like structure. Consider, for example, the following transcript:

S: How do I get [the while loop] to do that over again? See, I guess I don't know, I thought I had it. What happens now, how do I get it to go back? ... I say to myself, why would it do [the while test] after [the last line of the loop body]? It seems to me that it would do it as soon as the [variable tested in the while condition] changes. ...

I: So how will the while statement behave?

S: Again, total guess here, I'm saying the while statement, here's a logical guess ... everytime [the variable tested in the while condition] is assigned a new value, the machine needs to check that value ...

The subjects "logical guess" is that the while behaves like a demon and not as a specific testing step among other steps. This is consistant with English phrases like "while you are on the highway, watch for the Northfield sign". Soloway et al [1981a] report that 34% of an introductory programming course had the "while demon" misconception.

Novices also try to implement the "get a value, process that value" plan, even though they are programming in Pascal. Consider. the following novice program fragment,

```
   ...
var Count, Total, I : integer;
   begin
   Count := 0
   Total := 0
   Writeln ('Enter integer')
   Read (I)
   while I <> 99999 do
      begin
      Count := Count + 1
      Total := Total + I
      Read (I) <crossed out>
      end
   ...
```

and a transcript of the subject discussing this program:

S: If I put a number in [at the top of the loop], it comes through [the loop body]. I don't think I want [the inside Read] read again, I want it read up [at the top of the loop] ... If I read it [at the bottom of the loop body], what's that going to do for me? It's not going to do anything for me. OK, if I come out of the loop, having entered [a value], finish all [the loop body], then if I read in another one [points to Read above the while, traces a flow from that outside Read down through the loop]. I guess what I need to figure out is how do I get back up here [points to the Read above the while].

The subject wants to put the Read at the top of the loop, making the test in the middle of the loop. This allows the "get a value, process that value" plan. In a separate study Soloway, et al [1981b] show that a new Pascal looping construct supporting this plan significantly improved novice and intermediate performance with Problem 2.

3. Conclusions

The implication of these results is not simply to make syntactic fixes to programming languages. Instead, we are suggesting that the knowledge people bring from natural language has a key effect on their early programming efforts. Shneiderman and Mayer [1979] have proposed a model of programmer behavior based on language specific knowledge (which they call "syntactic") and more general programming knowledge (called "semantic"). Our results suggest that there is a third body of "natural language step-by-step specification knowledge" which strongly influences novice programming behavior.

Miller [1981], Green [1981], and others have previously looked at step-by-step natural language specifications. They concentrated on looking at the suitability of natural language for directing computers. Based on the ambiguities and complexity limitations of natural language, they concluded it would be quite difficult to "program" in natural languages. Here, we are not contradicting that result, but extending it. We are finding that novice programmers do use natural language, even when they think they are using a programming language.

There are several implications of this work for programming education. We are beginning to explain many novice programming errors through the idea of natural language step-by-step specification plans. The quality of these explanations has proved important in the development of a tutor to do intelligent computer assisted instruction of programming [Soloway et al, 1981c]. In the future, we hope to extend the tutor to understand a stylized form of these natural language plans.

Finally, what is the key to cognitively appropriate novice computing systems? Our work suggests that we need serious study of the

knowledge novices bring to a computing system. For most computerized tasks there is some model that a novice will use in his or her first attempts. We need to understand when is it appropriate to appeal to this model, and how to move a novice to some more appropriate model.

Acknowledgements - My deepest thanks to Elliot Soloway for his support and guidance. I would also like to thank John Clement for his critical comments.

4. References

Bonar, Jeffrey, Kate Ehrlich, Elliot Soloway, and Eric Rubin, (1982) "Collecting and Analyzing On-Line Protocols from Novice Programmers", in Behavioral Research Methods and Instrumentation, May 1982.

Du Boulay, B. and T. O'Shea (1981) "Teaching Novices Programming", in Computing Skills and the User Interface edited by M.J. Coombs and J.L. Alty, Academic Press, New York.

Green, Thomas (1981) "Programming As a Cognitive Activity", in Human Interaction With Computers, edited by C. Smith and T. Green, Academic Press.

Miller, Lance A. (1981) "Natural language programming: Styles, strategies, and contrasts", IBM Systems Journal, 20:2, pp. 184-215.

Shneiderman, Ben and Richard Mayer (1979) "Syntactic/Semantic Interactions in Programmer Behavior: A Model and Experimental Results", International Journal of Computer and Information Science, 8:3, pp. 219-238.

Soloway, Elliot, Jeffrey Bonar, Beverly Woolf, Paul Barth, Eric Rubin, and Kate Ehrlich (1981a) "Cognition and Programming: Why Your Students Write Those Crazy Programs", appeared in proceedings of the National Educational Computing Conference.

Soloway, Elliot, Jeffrey Bonar, and Kate Ehrlich (1981b) "Cognitive Factors in Looping Constructs". Computer and Information Science Technical Report 81-10, University of Massachusetts, Amherst, May.

Soloway, Elliot, Beverly Woolf, Eric Rubin, and Paul Barth (1981c) "Meno-II: An Intelligent Tutoring System for Novice Programmers", Proceedings of International Joint Conference in Artificial Intelligence, Vancouver, British Columbia.

Soloway, Elliot, Kate Ehrlich, Jeffrey Bonar, Judith Greenspan, (1982) "What Do Novices Know About Programming?", To appear in Directions in Human-Computer Interactions, edited by B. Shneiderman and A. Badre, Ablex Publishing Company.

Waters, Richard C., (1979) "A Method for Analyzing Loop Programs", IEEE Transactions on Software Engineering, SE-5:3, May.

Tacit Programming Knowledge

Elliot Soloway
Kate Ehrlich

Cognition and Programming Project
Computer Science Dept.
Yale University
New Haven, Ct. 06520

1. Introduction

[1] The goals of the Cognition and Programming Project at Yale University are:

- empirically explore the issues surrounding programming
  - what does an expert programmer know, and how does this compare to what a novice does (and doesn't) know [Soloway et al., 1982a, Ehrlich & Soloway, 1982],
  - what makes a programming language construct "cognitively appropriate" — and can we design such constructs [Soloway et al., 1981a]
  - what is the relationship between algebra knowledge and programming knowledge [Erhlich et al., 1982, Soloway et al., 1982]
- build AI based computer environments which can aid the novice programmer in learning to program [Soloway et al., 1981b, Soloway et al., 1982b].

In this short paper, we will describe some techniques we employ to investigate the first issue: what do programmers know.

2. Programming Plans: The Tacit Knowledge in Programming

A number of researchers have replicated the chess experiments of deGroot [deGroot, 1965] and Chase & Simon [Chase and Simon, 1973] in the domain of programming; consistent with those earlier experiments with master and non-master chess players, it appears that expert programmers also have more knowledge which is more highly chunked than novice programmers [Shneiderman, 1976, Adelson, 1981, McKeithen, Reitman, Rueter and Hirtle, 1981],

Building on this work, our goal is to identify the *specific* knowledge which expert programmers appear to have and use. The problem is that experts are often unaware of using this sort of knowledge — hence the term *tacit* knowledge. Collins [Collins, 1978], Larkin [Larkin et al., 1980], Rissland [Rissland, 1978], etc. have argued for the importance of tacit knowledge in various domains; our objective is to identify the tacit knowledge in programming.

To this end, we have developed a first order theory of the programming knowledge underlying simple looping programs which we feel experts have and use. Knowledge in this theory is encoded in terms of *plans*: stereotypic chunks of knowledge. For example, we posit that there are control flow plans and variable plans; in Figure 1, we would suggest that the body of the program is an implementation of the Running Total Loop Plan: new values are successively generated, in this case by a Read, and are added to a Running Total Variable, Sum. Also, there is Counter Variable, Count, which keeps track of the number of numbers generated. Our approach to programming plans is similar in spirit to that of Rich [Rich, 1980] and Waters [Waters, 1979].

```
Problem  Read  in  a set of integers and print out their average  Stop
reading numbers when the number 99999  is  seen.  Do  NOT  include  the
99999 in the average
```

[1] This work was supported in part by the National Science Foundation, under NSF Grant SED-81-12403.

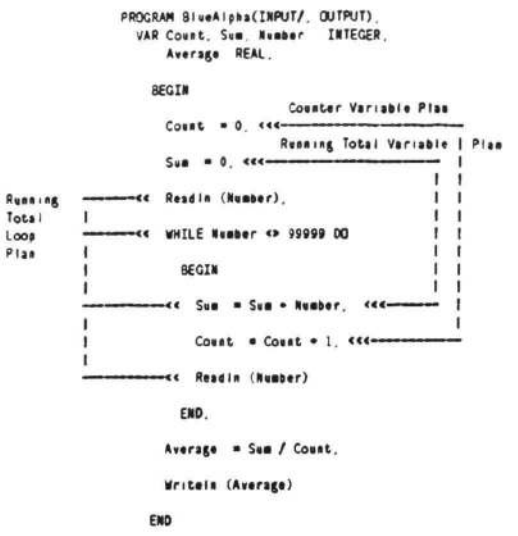


Figure 1: Examples of Plans

How does one go about testing a theory of this sort? Simply asking programmers whether or not they use the Running Total Loop Plan would not be too illuminating: the claim is that they are often unaware of having and using this type of knowledge. Below we describe techniques which we have found useful in this regard.

3. The Fill-in-the-blank Technique

The first technique we have used draws on work done in exploring the reality of scripts in text understanding. For example Bower, Black and Turner found that, in response to questions about a story, subjects would "fill in" from their "script" knowledge, information which was not explicitly given in the text. Similar in flavor, we give programmers a program in which a line of code has been left out, and ask them to fill it in. We purposely do not tell the subjects what the program is supposed to do; our objective is to have subjects use their experience with previous programming problems in order to recognize what line of code is most appropriate in the particular situation. If subjects didn't have plan structures, we would expect the answers they give to be arbitrary, and thus vary wildly from subject to subject. As we discuss below, the answers which novices give typically do vary significantly, while the answers which advanced programmers give do in fact exhibit a significant degree of consistency.

We also add an extra twist to the above design in order to more precisely home in on plan knowledge. We create two versions of the test program; in the first version, the information needed to fill in the blank line is more or less unambigious, while the second version contains *conflicting* information. For example, the programs in Figures 2 and 3 are both intended to produce the square root of N. Since N is in a loop which will repeat 10 times, 10 values will be printed out. The question is: how should N be set? The technique will be to compare the performance of programmers on the program which does not contain the plan conflict (Figure 2), with their performance on the program which does contain the conflict (Figure 3).

Please complete the program fragment given below by filling in the blank line (indicated by a box). Fill in the blank with a line of Pascal code which in your opinion best completes the program.

```
program VioletAlpha(Input/, Output).
  var N  real.
      I   integer.
  begin
  for I  = 1 to 10 do
    begin
    ------------------------------
    |                            |.
    ------------------------------
    if N < 0 then N  = -N.
    Writeln ( Sqrt(N) ).
       (* Sqrt is a built-in
         function which returns the
         square root of its argument*)
    end.
  end
```

Figure 2: Problem VioletAlpha:
The influence of a single Plan

In the program in Figure 2, N is a New Value Variable, since its function is merely to hold successive values. The plan for this type of variable does not present an overriding constraint on how it should be set in the blank line: a Read(N) or a N := N + SomeValue would both be acceptable. However, context does provide a strong constraint. Notice the If test preceding the Sqrt(N) instantiates the *"guard a portion of a program from improper data"* plan by protecting the Sqrt from negative integers (the Sqrt function can only work on positive integers). This test specifies an important constraint: N should take on values that could possibly be negative, otherwise the If test would be totally superfluous. Thus, N should

Please complete the program fragment given below by filling in the blank line (indicated by a box). Fill in the blank with a line of Pascal code which in your opinion best completes the program.

```
program VioletBeta(Input/, Output).
  var N  real.
      I   integer.
  begin
  N  = 0.0
  for I  = 1 to 10 do
    begin
    ------------------------------
    |                            |.
    ------------------------------
    if N < 0 then N  = -N.
    Writeln ( Sqrt(N) ).
       (* Sqrt is a built-in
         function which returns the
         square root of its argument*)
    end.
  end
```

Figure 3: Problem VioletBeta:
A conflict between Plans

not be set via an assignment statement to some simple function of N and/or the index variable I, e.g., N:=N+I, N:=I, N:=N+1. Rather, by setting N via a Read statement, negative values have the possibility of entering the program. This argument is based on a principle of tacit communication which states: ***include only necessary code in a program.*** By including a test for negative values, an experienced programmer is informing the reader that it is possible that such numbers could be generated; if such numbers could not possibly enter the program, then the inclusion of this test would violate this unwritten rule of communication.

The blank line in the program in Figure 2 is strategically placed: we wanted to explore the degree to which programmers are sensitive to the contextual relationship which obtains between the guard plan and the initialization aspect of New Value Variable Plan.

Program VioletBeta in Figure 3 is exactly the same as that in Figure 2 except that now N is given a value of 0 before the loop. Previously the New Value Variable Plan was neutral with respect to how N should be set. However, since N was initialized via an assignment statement to 0, the general rule of relating initialization to update should come into play, and direct that N be updated via an assignment. On the other hand, the If test, which realizes the "guard plan" and protects the square root operation, still sets up the expectation that N will be read in. If N will be set via a Read in the loop, the setting of N to 0 initially is superfluous. Thus, in Program VioletBeta we have purposely created a situation in which two plans are in conflict: the New Value Variable Plan expects N to be updated via an assignment, since it was initialized via an assignment, but the guard plan on the Sqrt operation expects that N will updated via a Read, so as to permit negative values to enter the program.

We felt that novices, with their limited experience, would be more sensitive to the constraint that obtains between a variable's initialization and update, as compared to the constraint that obtains between a guard plan and a variable's update. Hence, we predicted that the proportion of novices who Read in the value of N would decline when there was a conflict between plans. On the other hand, we felt that more advanced programmers would have had sufficient experience in both, and know when each is most appropriate, e.g., non-novices would realize that the test for a negative N should take precedence over the initialization of N to 0, since the "guarding" of the input is usually very important to the correct running of the program. Thus, we predicted that non-novices would fill in the blank with Read(N) equally often in both versions of the problem.

	NOVICES			NON-NOVICES	
	ALPHA no conflict	BETA conflict		ALPHA no conflict	BETA conflict
Category 1: Set N via Read	44	30		20	26
Category 2: Set N via assignment	7	15		4	4
	chi-squared = 5.20, p < 0.05			chi-squared < 1, N.S.	

Table 1: Fill-in-the-blank Responses

The responses of novices and non-novices on these programs, shown in Table 1, support our predictions. Non-novices chose to set N via a Read in the non-conflict case (VioletAlpha), and also chose to set N via a Read in the conflict case (Beta). This is consistent with our hypothesis that non-novices could use contextual information — the guard plan constraint — to override the variable plan constraint in the conflict case. In contrast, novices chose Read significantly less often in the conflict case than in the non-conflict case (chi-squared= 5.20, $p < 0.05$). This is consistent with our hypothesis that novices were more influenced by the familiar variable plan constraint than by the less familiar, contextual guard plan constraint.

4. Reading Time Studies

We also wanted to see how reading time was effected by the no conflict/conflict situations. Thus, we carried out studies which tracked the time a programmer started reading the program to the time he began to fill in the blank. For the programs in Figures 2 and 3, we found that novice programmers took effectively the same amount of time to respond in Program VioletAlpha as in Program VioletBeta (see Table 2). In contrast, while the advanced programmers responded quicker than the novices on Program VioletAlpha, they took significantly longer than the novices to respond to Program VioletBeta. We feel these data also support our hypothesis that Program VioletBeta contained a conflict between plans, to which only the advanced programmers were sensitive, while there was no similar conflict in Program VioletAlpha.

Alpha	Beta	
109	150	Novices n s
72	193	Non-novices p < .05

Mean Reading Times
In Seconds

Table 2: Reading Times Study

5. Concluding Remarks

Tapping into the tacit knowledge which programmers seem to have and use is a complex task. The basis for our experimental methods rests squarely on our, albeit preliminary, theory of programming knowledge. That is, we needed the theory in order to create the programs which serve as our stimulus materials. We are currently working on extending that theory to more complex programming problems and constructions.

We are also carrying out fill-in-the-blank studies and reading time studies with unplan-like programs, and programs which contain bugs. One objective is these studies is to explore the extent to which programs can be perturbed and still have people recognize the correct underlying intentions.

A longer range goal is the development of measures of program complexity based not just on features of the program text itself, but rather on the cognitive demands which the program makes on the programmer. Black and Sebrechts [Black & Sebrechts, 1981] have argued quite persuasively that measures of program complexity based on textual features (e.g. number of operations, length of variable names) cannot be effective measures, in the same way that the old measures of reading complexity, based also on textual features, were not effective measures. Such measures can capture only "surface" information. In contrast, effective measures must be based on the types and number of inferences which a programmer must make in order to understand the program. By cataloging the types of inferences which programmers do make, we have taken a first step in this enterprise.

Acknowledgements

We would like to thank Chuck Rich for his help in developing the stimulus materials used in this experiment, John Leddo for running the reading time study, and Joost Breuker and Valerie Abbott for their help in analyzing the data.

References

Adelson, B. Problem Solving and the Development of Abstract Categories in Programming Languages. *Memory and Cognition*, 1981, *9*, 422-433.

Black, J.B. & Sebrechts, M.M. Facilitating human-computer communication. *Applied Psycholinguistics*, 1981, *2*, 149-178.

Chase, W.C. and Simon, H. Perception in Chess. *Cognitive Psychology*, 1973, *4*, 55-81.

Collins, A. *Explicating the Tacit Knowledge in Teaching and Learning*. Technical Report 3889, Bolt Beranek and Newman, Cambridge, Mass., 1978.

deGroot, A.D. *Thought and Choice in Chess*. Paris: Mouton and Company 1965.

Ehrlich, K., Soloway, E. *An Empirical Investigation of the Tacit Plan Knowledge in Programming*. Technical Report 82-30, Dept. of Computer Science, Yale University, 1982.

Ehrlich, K., Soloway, E., Abbott, V. *Styles of Thinking: From Algebra Word Problems to Programming Via Procedurality*. Cognitive Science Society, Univ. of Michigan, Mich., 1982.

Larkin, J., McDermott, J., Simon, D. and Simon H. Expert and Novice Performance in Solving Physics Problems. *Science*, 1980, *208*, 140-156.

McKeithen, K.B., Reitman, J.S., Rueter, H.H., Hirtle, S.C. Knowledge Organization and Skill Differences in Computer Programmers. *Cognitive Psychology*, 1981, *13*, 307-325.

Rich, C. *A Library of Plans with Applications to Automated Analysis*. Technical Report 294, MIT AI Lab, 1980.

Rissland, E. Understanding Understanding Mathematics. *Cognitive Science*, 1978, *2(4)*, .

Shneiderman, B. Exploratory Experiments in Programmer Behavior. *International Journal of Computer and Information Sciences*, 1976, *5,2*, 123-143.

Soloway, E., Lochhead, J., Clement, J. Does Computer Programming Enhance Problem Solving Ability? Some Positive Evidence on Algebra Word Problems. In R. Seidel, R. Anderson, B. Hunter (Eds.), *Computer Literacy*, New York, NY: Academic Press, 1982.

Soloway, E., Bonar, J. and Ehrlich, K. *Cognitive Factors in Programming: An Empirical Study of Looping Constructs*. Technical Report 81-10, Department of Computer Science, University of Massachusetts , 1981.

Soloway, E., Woolf, B., Barth, P., and Rubin, E. *MENO-II: Catching Run-Time Errors in Novice's Pascal Programs*. International Joint Conference on Artificial Intelligence, Vancouver, B.C., 1981.

Soloway, E., Ehrlich, K., Bonar, J., Greenspan, J. What Do Novices Know About Programming? In Directions in Human-Computer Interactions, B. Shneiderman and A. Badre, Eds., Ablex, Inc. in press.

Soloway, E., Rubin, E., Woolf, B., Bonar, J. MENO-II: A AI-CAI Programming Tutor. Proceedings of the ADCIS Conference, Vancouver, B.C., in press.

Waters, R.C. A Method for Analyzing Loop Programs. *IEEE Trans. on Software Engineering*, May 1979, *SE-5*, 237-247.

THE ROLE OF METAPHORS IN NOVICES LEARNING PROGRAMMING

Ann Jones

The Open University
Milton Keynes, England

Abstract

Learning a complex skill such as programming requires the developments and use of conceptual models, both of the concepts in the programming language, and the 'behaviour' of the machine. The latter has been referred to as the 'notional machine' (du Boulay, B., O'Shea, T. and Monk, 1981). Such a conceptual model, however, must interact and build upon models and metaphors which students already have. It is these metaphors and some techniques for studying them which are discussed in this paper.

Introduction and Background

Behavioural studies of programming are motivated by a diversity of goals, for example a desire to understand the task better and thus how it can be performed more efficiently: or a concern about the importance of developing procedural literacy, (for example, Sheil, 1980, (b)) or an interest in programming as an applied example of a high level skill. It is the latter, mainly, which motivates the present study: the overall question, although it is far from simple, can be simply phrased 'What goes on in the mind of the learner programmer?'

There is now a substantial body of research on programming, although Sheil, (1980) argues that many empirical studies of programming have added very little to our knowledge of what it means to learn programming, partly because the methodology is fraught with difficulties but mainly because we still know so little about what the task entails: how programming knowledge is organized and how it can be represented. There is some agreement that it can be thought of as a collection of units, (or 'frames', 'paradigms' or 'schemas') organised as program fragments with a set of propositions about its behaviour and rules for combining it with others. (Rich 1978, Floyd, 1979.) There is no evidence, however, that novices have access to such structures; on the contrary, studies of the knowledge organization of experts and novices, in the programming domain, indicate not unsurprisingly, that novices lack such organizing schemas. (McKeithen and Reitman, 1981; Adelson 1981). It has been argued (du Boulay, O'Shea and Monk, 1981.) that one of the difficulties of teaching a novice programming is how to describe at the right level of detail the machine she is learning to control; and as a way of doing this they suggest teaching using the idea of a notional machine - an idealised conceptual computer whose properties are implied by the constructs in the programming language employed. The notional machine is similar to the 'transactions' which Mayer uses to describe the workings of a BASIC machine (Mayer, 1979), and also suggests as a basis for teaching BASIC. Other studies, (e.g. Miller, 1974) have also emphasized providing novices with carefully thought out metaphors and models to help them learn. The gap, however, in all this is what the learner herself brings to the situation, and that is, all her past experience and knowledge which will be used to interpret and organize the new material that is to be learnt. Although programming has many specialised terms, many words do have everyday meanings and associations which are different from their programming use and may not facilitate the learning process. Such words will not necessarily have a shared meaning among novices. McKeithen and Reitman (1981), in studying the organization of programming knowledge found that beginners' organizations show a rich variety of common language associations to these programming concepts. Botts' study of learning how to use a text editor, (Bott, 1979) suggests that such pre-associations are powerful and pervasive, and may not be easily replaced by new metaphors and models.

The Study

The study of novice programmers' initial metaphors and models is being investigated as part of a larger study of how students learn two very different programming languages: only one of which will be discussed here, a language called SOLO. The students using SOLO are taking an Open University Cognitive Psychology course; and SOLO provides an environment for them to manipulate an assertional data base, as a tool for learning and thinking about knowledge representation. It was designed to make life easy as possible for total novices by being restricted to a small number of primitives, and incorporating many user aids, such as a spelling corrector. Nevertheless, learning to program in SOLO is by no means trivial, and so it has been necessary to find out what problems the students have so that the programming environment can be tailored to suit their needs. (Eisenstadt, Laubsch and Kahrey, 1981). In order to build a really fool-proof environment it is necessary to understand precisely what the novice really thinks is going on inside the SOLO machine. Of course, trying to find out what someone 'really thinks' is going on is a tall order, and to a large extent this study has been concerned with exploring methodologies which will provide 'windows' into novices' conceptual models of the SOLO machine. Several such windows have been explored: students were asked to actually start learning SOLO in the laboratory and to talk aloud while doing the exercises in the primer, (for this study, the fact that SOLO is learnt at a distance, from a correspondence text, is a great advantage, as the teaching method and techniques used are made explicit and are consistent across students); they are interviewed and asked to talk about some of the concepts before they started; they completed Repertory grids, (Kelly, 1970) and the worked through some very simple exercises. The next section will briefly discuss the virtues and problems of some methodologies for specifically exploring the metaphors and models which students bring with them, and some examples of these.

The Repertory grid

Repertory grids are usually used to elicit constructs rather more general than those concepts used in programming. They are a way of finding out how a person categorizes the world or some part of the world. Other studies of knowledge structures, e.g. Adelson (1981) have investigated how programs

are organised. The approach used here however was to ask students to categorize the actual promitives of the SOLO language, and subjects talked aloud while they did the exercise. In this exercise they were shown three of the SOLO 'words' on cards, and asked if there was a way in which two were alike and one was dissimilar; and told that they would then categorize the rest of the cards according to this construct. This is repeated until no more constructs can be elicited. In doing this I was interested in the constructs a subject would choose, given the freedom to carve up the 'SOLO world' however she wanted, and also whether, given a construct elicited from the first triple it would be possible to categorize the remaining SOLO terms in accordance with it. For example, one subject started with NOTE (an 'Assert' function), CHECK (a 'retrieve' function) and LIST - which lists procedures. She said that NOTE was to do with 'Giving (the database) new things' whereas CHECK and LIST have a retrieval function - 'they show you what's there.' It's not clear how much sense such constructs have for the rest of the primitives. This subject's categorization for this construct was:

"Giving new things"	"Retrieval"
DESCRIBE	IF PRESENT
NOTE	IF ABSENT
To (Defining a procedure)	CHECK
CONTINUE	EDIT
EXIT	
PRINT	
PARAMETER	
VARIABLE	
FORGET (delete)	

In this case, the construct does make some sense as a way of categorizing the SOLO terms. As might be expected from other related studies (eg Adelson, 1981) many of these beginner programmers used constructs which were idiosyncratic and related to everyday knowledge as much as to programming knowledge.

Eliciting such constructs is clearly a hard activity for beginners (as indeed it is for experts), but although subjects found it demanding they also found it rewarding as it forced them to think about how they organized these concepts which they were not aware of having categorised. This thinking it through becomes explicit in their verbalization. What seems to be happening to this is that the students are learning while doing the task. Whilst this is exciting and can provide rich data it is also clearly problematic for analysis as the learner's state is changing during the exercise.

Secondly, some students seem to "chain" concepts; to lose track of the construct or category and to categorize each element or concepts (the primitives) according to its similarity or otherwise, to the previous one.

The grouping of the constructs for each subject and differences between subjects has not yet been analysed, and so, overall, it is difficult at the moment to assess how useful this technique will be.

Concept Interviews and Talking through Exercises

The most fruitful technique so far for investigating metaphors has been a combination of concept interviews and students talking aloud while working through the exercises in the promer. In the 'concept' interviews, subjects were asked to talk about some of the words used in the SOLO language before starting to learn about it. Information about how a student interprets a certain word before even starting provides a baseline for interpreting their behaviour. Some interpretations are remarkably common, yet unsurprising, (in retrospect!) "Parameter", if it elicited any reply at all, was "limits", "boundaries" "constraints", which is not particularly close to its programming use. 'Node', on the other hand, which is a word used in SOLO's semantic network, was for many the biological 'node' of a nerve network; - a not inappropriate metaphor.

Protocols of students working through exercises in the teaching test were also taken. Consider the following extract from a transcript, which conerns how procedures take paramenters:

"I think of it in relation to a sort of work processor, that if I was doing a lot of letters I would do a letter and put an X in 'Dear X' and then each one I'd just print in Fred, Mary.. so that each letter...."

This metaphor is very useful for a novice thinking about the idea of a procedure taking a parameter. It should be stressed however that this is something the learner brings with her; no matter what metaphors are offered in the teaching, the student must map what she's learning on to her own experience. Normally this is a 'hidden' activity; but part of the aim of this research is to make it 'explicit'. Such information can pay dividends later, ss such metaphors may only be useful for a fragment of time, or for learning on particular thing. In the above example, the metaphor led to expectations about the way the editor would behave, - that it would be able to delete single words within a line of the procedure, which did not match its actual behaviour, as in fact, whole lines must be deleted, and yet, such beliefs were persistent. Both (1978) cites the way in which such expectations can lead to interpretations which are false, yet very pervasive, and how complicated stories are constructed to account for the mismatch between expectation and reality, before the learner finally discard an inappropriate schema.

Conclusions

In helping novices to learn programming, it is not enough to provide metaphors and models. In addition we must study the models and metaphors students already have, and which they bring with them to the learning situation. This paper has discussed such metaphors and possible methodologies for investigating them. The next step is to examine how existing metaphors interact with experience in the development of conceptual modes of a programming language. This is a vital issue in understanding how novices learn programming.

Acknowledgements

I would like to thank Tim O'Shea and Richard Young for help and advice on this paper, and for helpful discussions.

References

Adelson, B. Knowledge Structures of Computer Programmers Proceedings of 3rd Annual Conference of Cognitive Science Society, 1981, pp 243-248.

Bott, R.A. A Study of Complex Learning. Report No. 82, University of California, Centre for

Human Information Processing, 1979.

du Boulay, B., O'Shea, T., and Monk, J. The black box inside the glass box: presenting computing concepts to novices. Int J Man Mach Studies, 1981, 14, 237-249.

Eisenstadt, M. Artificial Intelligence Project, Cognitive Psychology Course, Open University 1978.

Eisenstadt, M., Laubsch, J., and Kahney, H. Creating Pleasant Programming Environments for Cognitive Science Students. Proceedings of 3rd Annual Conference of Cognitive Science Society 1981.

Floyd, R,W, "The Paradigms of Programming" Commus. ACM 228 (August 1979) 455-460.

Kelly, G.A. A brief introduction to personal construct theory. In Perspectives in Personal Construct Theory (BANNISTER D. Ed), London Academic Press, 1970.

Mayer, R. A Psychology of Learning BASIC Communication of the ACM, Vol. 22, No. 11, Nov. 1979.

McKeithen, D.B. and Reitman, J.S. et al. Knowledge Organization and Skill Differences in Computer Programmers. Cognitive Psychology, 13, 307-325, 1981.

Miller, L.A. Programming by non programmers. Int J Man-Mach Studies, 1974, Vol. 6, 273-260.

Rich, C., and Shrobe, H. "Initial report on a Lisp programmer's apprentice", IEEE trnas. Softw. Eng. SE-4 (1978) 456-467.

Sheil, B. The Psychological Study of Programming, Computing Surveys, Vol. 13, Nol. 1, March 1980.

Sheil, B(b). "Teaching procedural literacy" in Proc. ACM Annual Conf. 1980, pp 125-126.

PROGRAMS, THEORIES, AND MODELS

Paul Thagard

Cognitive Science Center, University
of Michigan, Ann Arbor

University of Michigan-Dearborn

April, 1982

This paper makes use of the philosophical literature on theories and models to develop an account of the role of AI programs in psychological theorizing. It is often said that programs *are* theories (e.g. Winston 1977, p. 258). I argue that programs do *not* constitute theories or models in any precise sense, but that the important contribution of programs to psychological theory can be described by adopting a new conception of theories as *definitions* of kinds of systems, developing a cognate conception of model, and interpeting AI programs as simulations of models which approximate to theories.

A program - a set of instructions which a computer can follow - is clearly not a theory according to what used to be the standard philosophical view that theories are sets of sentences axiomatized in a formal system (see e.g. Hempel 1965, pp. 182-183). However, a more plausible interpretation of theories is available.

The alternative conception of scientific theories was originally proposed by P. Suppes (1957, 1967) and has been developed by various authors and applied to fields as diverse as physics, biology, and economics (see e.g. Sneed 1971, F. Suppe 1972, 1977; van Fraassen 1970, 1972; Stegmueller 1976, 1979; Beatty 1980; Hausman 1981). It has been variously referred to as the "semantic" conception and the "structuralist" view of theories. There are important differences among these various accounts, but in what follows I shall eclectically adapt whatever features of the different formulations seem best to apply to cognitive science. In order to avoid confusion, I shall simply refer to the "new" conception or account of theories.

Whereas the traditional view of theories took them to be sets of sentences in an axiomatic system, the new account takes a theory to be a kind of definition. In Suppes' original account, a theory was a definition of a set-theoretic predicate, but for present purposes I shall employ a simpler version of the new account due to Giere (1979). For Giere, a scientific theory is a definition of a kind of natural system (p. 69). He illustrates his account by applying it to the theory of Newtonian mechanics. On the traditional view, this theory might be taken as consisting essentially of Newton' three laws of motion plus the law of universal gravitation. On Giere's view, Newtonian theory is a definition of a kind of particle system: "A natural system is a classical Newtonian particle system if and only if it is a system of objects satisfying Newton's three laws of motion and the law of universal gravitation." (p. 69). As a definition, such a theory is neither true nor false: in itself, it makes no empirical claim. However, it can be used to make empirical claims, for example that the solar system is a system of the kind defined by the theory. Giere calls such claims "theoretical hypotheses", but I shall term them simply "theoretical claims". A theoretical claim has the form: real system R is a system of the kind defined by the theory T.

Whereas a program is clearly not a set of sentences comprising a theory on the traditional view, it is very tempting to think of a program as specifying a kind of cognitive system and hence as qualifying as a theory on the new conception. For example, Kosslyn's imagery programs might be understood as specifying a kind of system for processing information using mental images. John Anderson's programs define a different sort of processing system, oriented around propositions. In either case, we might make the claim that the real human information processing system is a kind of system specified by the program. Such a claim can be empirically evaluated.

A program implicitly characterizes a processing system by specifying what knowledge structures are to be used and what procedures are to operate on them. Although this makes it appealing to say that a program can be a theory according to the new conception, there are two important reasons for resisting the appeal. First, although a program can loosely be said to "characterize" a processing system, it can not be said to *define* a system in the way required by the new conception of theories. Second, we would never want to make the theoretical claim that any real system is just like the system produced by the program, since any program contains a host of implementation-dependent characteristics which we know to be extraneous to real human cognition.

To handle the latter difficulty, I want to develop the concept of a *model*. This is a dangerous choice of term, since "model" has been used with even more ambiguity and vagueness than has "theory". However, the term "model" is often used in cognitive science in much the way I want to define it, and I hope to give a definition sufficiently precise to distinguish models from theories.

As Giere and others have pointed out, "model" and "theory" are sometimes used synonymously, but I think we can outline two features which generally distinguish models from theories. (Cf. Kosslyn 1980, Pylyshyn 1978.) First, models are intended only to have analogies with real systems; they are not expected to characterize them with complete accuracy (cf. Hesse, 1963). Second, models are often intended to have a relatively narrow range of application: we can have models for specific phenomena, whereas theories are usually intended to have wide generality. I shall now show how these features of models can be characterized within the general framework of the new conception of theories. We will still not be able to say that a program *is* a model, but the account of models will bring us closer to describing the role of programs in model building and theory construction.

On my interpretation models are like theories in being definitions of a kind of system, and so are in themselves neither true nor false. However, as indicated above, we expect models to include in the definition of a kind of system features which we would not attribute to real systems. Models define systems which we know not to be exactly like

real world systems. Accordingly, the claims which models are used to make must be different from the claims which theories are used to make. Recall that theories are used to make theoretical claims that a real system *is* a system of the kind defined by the theory. Since a model contains specifications which are known to be false of the target real systems, it can not successfully be used to generate such theoretical claims. For example, a processing model based on the computer metaphor may define a kind of system in which processing is serial, even though the theorizer believes that processing in the brain is parallel. That discrepancy would be enough to defeat any theoretical claim which said that the brain is a processing system of the kind described in the model. We need to be able to use the model to make a weaker claim.

As Hesse (1963) and Kosslyn (1980) have pointed out, the relation between a model and what it models is one of *analogy*. We do not assume that a model exactly describes the target phenomena, only that the phenomena are in important respects *like* what is described in the model. Under the new conception, we can say that a model defines a kind of system, but that we only expect the systems so defined to be analogous to real systems. Hence instead of a theoretical claim we use a model to make what I shall call a "modelling claim", which has the form: a given real system R is very much like the kind of systems defined by model M. This is clearly less precise than the identity claim made in a theoretical claim.

Models are thus less ambitious than theories. Not only do they include in their definitions characteristics which real systems are not expected to have, they are likely to define a narrower set of characteristics than would a theory, which would be expected to give a more complete account of the behavior of a system. Theories are also expected to apply generally to a number of different kinds of systems, whereas models can be either general or specific (Kosslyn 1980). A general model of cognitive processing is one which would be like a theory in having numerous applications, generating numerous modelling claims. But models, unlike theories, can be specific in that they are intended to apply only to a particular sort of system, and a modelling claim is made only about that kind of system. Construing models as definitions of kinds of systems is clearly compatible with both their general and specific uses.

All this has been preparatory to asking the central question: are computer programs psychological *models*? Since models differ from theories in admitting unrealistic characteristics as part of their system definitions, it is tempting to construe programs at least as models of human information processing. But the second impediment remains: a computer program may exemplify a system, but it does not define a kind of system, and therefore can not qualify as a model in the precise sense developed above. Still, we continue to get closer to being able to specify the role of programs in the construction of psychological theories and models.

Zeigler (1976) usefully distinguishes between a real system, a model, and a computer, and says that whereas the relation between the model and the real system is one of *modelling*, the relation between the computer and the model is one of *simulation*. A computer simulates a model which models a real system. Indirectly, then, we can say that a computer is a simulation of a real system. Zeigler's notion of model is different from the one discussed here, but his basic distinctions can be translated into the terms of the current discussion.

When a program is run on a computer, the computer is a simulation of a system. In particular, the system simulated is intended to be a system of the kind defined by the model. A model defines a kind of system, and the program, when executed, performs like a system of the sort defined. The program thus embodies many important features of the model. Hence a program can be used indirectly to make claims about the real system about which a modelling claim is made. Since the program simulates a system of the kind defined by the model, and since the model can be used to make the claim that the real system is a system of the kind defined by the model, we can use the program to make a *simulation claim*: the real system R is analogous to the system S simulated by execution of program P. In short, a simulation claim can have the form "program P simulates R". However, it must be kept in mind that the claim in both these forms is shorthand for a description of a much more complex relation involving models as definitions of systems. In sum, a program can not be said to be a theory or a model, but provides, when executed, a simulation of a system of a kind defined by a model which approximates to a theory.

REFERENCES

Anderson, J.R. (1976), *Language, Memory and Thought*, Hillsdale, New Jersey: Erlbaum Associates.

Beatty, J. (1980), "Optimal-Design Models and the Strategy of Model Building in Evolutionary Biology," *Philosophy of Science 47*: 532-561.

Giere, R. (1979), *Understanding Scientific Reasoning*, New York: Holt, Rinehart and Winston.

Hausman, D. (1981), *Capital, Profits, and Prices: An Essay in the Philosophy of Economics*, New York: Columbia University Press.

Hempel, C.G. (1965), *Aspects of Scientific Explanations*, New York: The Free Press.

Hesse, M. (1966), *Models and Analogies in Science*, Notre Dame: Notre Dame University Press.

Kosslyn, S. (1980), *Image and Mind*, Cambridge: Harvard University Press.

Moor, J. (1978), "Three Myths of Computer Science," *British Journal for Philosophy of Science 29*: 213-222.

Pylyshyn, Z. (1978), "Computational Models and Empirical Constraints," *Behavioral and Brain Sciences 1*: 93-99.

Suppe, F. (1972), "What's Wrong with the Received View on the Structure of Scientific Theories?" *Philosophy of Science 39*: 1-19.

Suppe, F. (1977), *The Structure of Scientific Theories* (second edn.), Urbana: University of Illinois Press.

Suppes, P. (1957), *Introduction to Logic*, New York: Van Nostrand.

Suppes, P. (1967). "What is a Scientific Theory?" in S. Morgenbesser (ed.), *Philosophy of Science Today*, New York: Basic Books, 55-67.

van Fraassen, B. (1970), "On the Extension of Beth's Semantics of Physical Theories," *Philosophy of Science 37*: 325-339.

van Fraassen, B. (1972), "A Formal Approach to the Philosophy of Science," in Colodny (ed.), *Paradigms and Paradoxes*, Pittsburgh: Pa.: University of Pittsburgh Press, 303-366.

Winston, P. (1977), *Artificial Intelligence*, Reading, Mass.: Addison Wesley.

Zeigler, B. (1976), *Theory of Modelling and Simulation*, New York: Wiley.

On Changing the "Logic" of Proposed Logics of Scientific Discovery

S.C. Grover
University of Calgary

Critics of the concept of a logic of discovery generally hold that discovery involves irrational, aesthetic and metaphoric components which preclude systematic description or reduction to an algorithmizable procedure (e.g. 1, 2). This paper reconsiders certain of the issues involved in this philosophical controversy and discusses the possibilities for computer simulation of inventive scientific thinking.

It has become increasingly clear via philosophical analysis and recent work in artificial intelligence, that traditional forms of logic fall short of providing an adequate description of the thinking underlying scientific discovery (3). For instance, Cohen (4) has shown that: "Newton derives his inverse square law of gravitation by a precise mathematical derivation from, among other things, Kepler's Third Law for planets. . .We can show logically that Newton's system contradicts Kepler's Third Law, while Newton coolly derives one from the other" (5, p. 260). Deductive logic does not seem to be the basis then for Newton's creativity in this instance. Inductive logic seems often to fare no better as an explanation for inventiveness: ". . .most of us cannot conceive that there might be rules that would lead us from laboratory data to theories as complex as quantum theory, general relativity, and the structure of DNA. Our shared archetypes of significant science virtually all involve theoretical entities and processes which are inferentially far removed from the data which they explain" (2, p. 178).

Inductive and deductive logic are incomplete models for a logic of discovery also in that often scientists do not begin with "valid" premises or "sound" data. Yet, they frequently arrive at theories and findings which are deemed highly significant and legitimate. So it was, for example, with Darwin who arrived at the theory of evolution -- based on the concept of natural selection -- from his monad theory which posited individual primitive life forms that arose spontaneously on a continual basis (6).

An overreliance on traditional logic may account for some of the limitations in contemporary computer simulations of scientific discovery processes (which are quite impressive nonetheless). Thus the Bacon ·1 and ·3 programmes must be given data free of noise to manipulate; lest the programmes' inductive processing be led astray. The ultimate consequence of such an approach is that these programmes can rediscover certain known empirical laws, such as the ideal gas law, but cannot generate new discoveries (7).

What other forms of logic might then be relevant to the problem of scientific discovery? The sort of logic required to account for, for instance, reformulations of problems into useful researchable ones is what Achinstein terms an "evaluative logic" (8). Such a logic would include rules for deciding on the plausibility and importance of research problems and "solutions". A theory might be considered more plausible if it accounts for more data or for puzzling empirical findings. Achinstein uses as an example Bohr's notion that the hydrogen atom consists of a nucleus around which a single electron revolves and sometimes jumps from one stable orbit to another. Achinstein contends that Bohr's hypothesis was considered plausible since it was useful in explaining the spectral lines present when hydrogen is excited by heat or electricity and emits light. Another example is Pauli's "discovery" of the neutrino. The concept of the neutrino was initially reluctantly accepted as plausible -- despite the absence of empirical evidence for a "neutrino event" -- because it could explain the failure of energy equations to balance before and after beta decay (9).

As the aforementioned examples illustrate, evaluative logic differs in important ways from deductive or inductive logic. It may lead to a concept or model in the absence of direct empirical support as in the case of Pauli's neutrino. In addition, evaluative logic is a flexible system which does not lead inexorably to any particular conclusions(s) as is the case with deductive logic. Thus Bohr's theory may have been a plausible one or the most plausible theory advanced at the time, however, the "logic" of the argument did not inherently preclude other possibilities.

Does this discussion not simply beg the question of how new ideas are generated in the first place, and substitute for that question the issue of theory justification? Gutting (10) holds that a logic of hypothesis generation is intimately linked to an evaluative logic which assesses ideas or models. As Gutting points out, the so-called truism that one can think of almost anything is false. He gives the following example: "Most people. . .even ones with sufficient intelligence and imagination, could not have thought of the hypothesis of electron spin. Only a scientist thinking of the atom in terms of a planetary model could have thought of such a hypothesis. On the other hand, the hypothesis is implicit in the model and so likely to occur to anyone who is seriously concerned with developing this model. So if the question is raised: Why did Goudsmit and Uhlenbeck think of the spin hypothesis? at least a significant part of the answer lies in a conceptual analysis of the nature of Bohr's model of the atom" (10, p. 224-225).

Thus discoveries occur given a particular historical and theoretical context. Such a context or background knowledge is not currently a significant feature of programmes such as the Bacon simulation attempts. It is as if the programme is largely expected to operate in a theoretical vacuum detecting regularities in the data which, as the programme's namesake Francis Bacon held, would "leap out" at the observer (7). However, in providing only "sound" data devoid of anomalies only a low level theoretical bias of a sort is built into the system. It seems that many attempts at simulating scientific discovery are, perhaps unwittingly, designed so as to be consistent with the notion that "science begins in the nothingness of ignorance" (11, p. 12). However, as Gould points out, theories always abound with the result that "science advances primarily by replacement not by addition" (11, p. 12).

Consider for instance Lavoisier's discovery of oxygen. It was his rejection of phlogiston chemi-

cal theory which was a prerequisite for development of the notion of combustion as due to a combination effect rather than a dissociation reaction. His contemporary, Priestly, did not reject phlogiston theory in the light of Lavoisier's evidence that combustion led to an increase in the weight of a burned compound and not a decrease as phlogiston theory necessitated. Priestly simply postulated that phlogiston has a "negative weight". This case illustrates Curd's point that: "The factors that justify our inferences to theories in the first place are the same as those that we use to decide which theory to pursue after they have been generated." (12, p. 215).

What is needed then are programmable rules which capture something of the logic of data and problem assessment given a particular theoretical framework. Also, required are higher order sets of rules that reflect on the theoretical assumptions upon which the programme operates. To accomplish this might be akin to equipping the programme with a metacognitive competency. Programmes such as Internist-I (13) come closer than others to operating on data given certain background knowledge e.g. a classification scheme for all possible diseases, and thus are more similar to the scientist who also comes to his research problem with a particular frame of reference. However, the Internist programmes, like the Bacon programmes, cannot make new discoveries e.g. a new disease is not generatable by Internist I or II. Perhaps in part this is because the metacognitive feature (for a lack of a better term) is absent. Fortunately, progress is being made in human research in the understanding of various aspects of metacognitive competencies (e.g. 14, 15, 16). Perhaps, the addition of a metacognitive component in computer simulations of scientific discovery processes will allow for more flexible programmes that make new discoveries, of a sort. Should the latter occur, a logic of discovery would not, as Wartofsky now claims, "dissolve the notion of creativity altogether" (1, p. 8).

References

1. Wartofsky, M.W. Scientific Judgment: Creativity and Discovery in Scientific Thought. In Nickles, T. (ed.) Scientific Discovery: Case Studies. Boston Studies in the Philosophy of Science, Vol. 60. Boston: D. Reidel Publishing, 1980, 1-20.

2. Laudan, L. Why was the Logic of Discovery Abandoned? In Nickles, T. (ed.) Scientific Discovery, Logic, and Rationality. Boston Studies in the Philosophy of Science, Vol. 56. Boston: D. Reidel Publishing, 1980, 173-183.

3. Grover, S.C. Toward a Psychology of the Scientist: Implications of Psychological Research for Contemporary Philosophy of Science. Washington: University Press of America, 1981.

4. Cohen, I.B. Newton's Theory versus Kepler's Theory and Galileo's Theory: An Example of a Difference Between a Philosophical and a Historical Analysis of Science. In Elkana, Y. (ed) The Interaction Between Science and Philosophy. Atlantic Highlands: Humanities Press, 1974, 299-338.

5. Hattiangadi, J.N. The Vanishing Context of Discovery: Newton's Discovery of Gravity. In Nickles, T. (ed.) Scientific Discovery, Logic and Rationality. Boston Studies in the Philosophy of Science. Vol. 56. Boston: D. Reidel Publishing, 1980, 257-265.

6. Perkins, D.N. The Mind's Best Work. Cambridge: Harvard University Press, 1981.

7. Langley, P. Data-driven discovery of physical laws. Cognitive Science, 1981, 5, 31-54.

8. Achinstein, P. Discovery and Rule-Books. In Nickles, T. (ed.) Scientific Discovery, Logic and Rationality. Boston Studies in the Philosophy of Science. Vol. 56. Boston: D. Reidel Publishing, 1980, 117-137.

9. Gale, G. Theory of Science. New York: McGraw-Hill, 1979.

10. Gutting, G. The Logic of Invention. In Nickles, T. (ed.) Scientific Discovery, Logic and Rationality. Boston Studies in the Philosophy of Science. Vol. 56. Boston: D. Reidel Publishing, 1980, 221-234.

11. Gould, S.J. The Mismeasure of Man. New York: W.W. Norton, 1981.

12. Curd, M.V. The Logic of Discovery: An Analysis of Three Approaches. In Nickles, T. (ed.) Scientific Discovery, Logic and Rationality. Boston Studies in the Philosophy of Science. Vol. 56. Boston: D. Reidel Publishing, 1980, 201-219.

13. Schaffner, K.F. Discovery in the Biomedical Sciences: Logic or Irrational Intuition? In Nickles, T. (ed.) Scientific Discovery: Case Studies. Boston Studies in the Philosophy of Science. Vol. 60. Boston: D. Reidel Publishing, 1980, 171-205.

14. Loper, A.B. Metacognitive Development: Implications for Cognitive Training. Exceptional Educational Quarterly 1980, Vol. 1, No. 1-8.

15. Kendall, C.R., Borkowski, J.G., Cavanaugh, J. C. Metamemory and the transfer of an interrogative strategy by EMR children. Intelligence, 1980, 4, 255-270.

16. Campione, J.C. and Brown, A.L. Memory and Metamemory Development in Educable Retarded Children. In Kail, R.V. Jr. and Hagen, J.W. (eds.) Perspectives on the Development of Memory and Cognition. New York: John Wiley and Sons, 1977.

A General Model for Simulating Information Processing Experiments

Earl Hunt and Pollyanna Pixton
The University of Washington

Psychologists who study cognition have followed two approaches. One is to isolate elementary processes of thought and study them in laboratory settings. Most of experimental psychology follows this tradition. Alternatively, one can study complex thinking directly, by developing descriptions of the processes of chess playing, mathematical problem solving, medical diagnosis, and the like. This tradition is dominant in Cognitive Science. The experimental psychology approach has produced a set of reasonably concise concepts applicable in restricted settings, but it is not clear how these concepts are to be combined during complex reasoning. The descriptive approach has produced concepts that describe thought, but the concepts are so flexible that it is often difficult to test them. The widely used production notation, for instance, is a way of thinking about thinking rather than a testable model of thought processes.

Our research attempts to unify the two approaches. Instead of trying to build up from elementary processes to complex acts, we have taken a "top down" approach. We assume that the production notation is an appropriate language for describing thought, and then use it to construct a unified model of information processing that is applicable to several laboratory paradigms.

If production notation programs ("production systems") can be written to model any thought process, then the mind, in general, must be an interpreter for such programs. We have written such interpreter, using concepts derived from experimental psychology in its construction. The interpreter contains sections dealing with the input of information over multiple "sensory" channels (Broadbent, 1971), the manipulation of information in working and long term memory (Baddeley, 1976), the activation of distinct coding systems within long term memory (Posner, 1978), and the execution of information processing steps by a cascade rather than a linear process (McClelland, 1979).

Overview of the Model

The basic programming construct, a production, is a two part rule,

pattern ⟶ action

where "pattern" refers to a set of conditions that must be met for a production to be activated, and "action" refers to the steps to be taken when the production's pattern conditions are met.

Time in the interpreter is divided into cycles. Within each cycle the following events take place, functionally in parallel. Assume that stimuli are present in the sensory channels and in working memory and that each production in long term memory has associated with it a number indicating its "level of activation". The interpreter compares the stimuli to the pattern half of each production. The comparison produces a numerical value that will be called the "strength" of the match. A production is considered "active" if the strength of the match exceeds a threshold associated with the pattern. A new activation level is then calculated, which is monotonically increasing function related to the difference between the strength of the match and the threshold value. (In most of our work, we simply use the difference). The activation level is then either increased or decreased by the activation level of other productions linked to it. This process, which constitutes "spreading activation" is referred to as priming. Finally, at the end of each cycle all activation levels are reduced ("decayed") to a proportion of their previous values.

When the activation level of one production exceeds the activation level of all competing productions by a preset criterion, the action half of the production is initiated. The action may be an external response, alteration of an internal parameter in the model, or generation of a stimulus in working memory. If no external response is made, "time" is incremented and the program continues cycling through the set of productions, now using the new stimulus or the old stimulus with new parameters, if they have been altered. Firing and cycling continues until an action terminates the program or the program exceeds the allowed processing time.

The program imposes psychologically justifiable constraints upon production execution. This is done in such a way that production processing will produce the phenomena observed in laboratory studies of mechanistic information processing. These constraints are described in more detail in the next section.

Details of the Program and Model

The program, which we call MIND, is written in standard Pascal and contains about 1000 lines of code. Production pattern-action pairs are currently represented in an internal symbol code rather than brief English statements.

A. Initialization

The input to the program consists of a set of productions, the threshold levels for each production and an association matrix which links the productions to each other in a negative, positive or null manner. Other program parameters read during initialization are the decay rate, the decision criterion (DR), an internal noise scale factor and a maximum processing time.

Information (a stimulus) is presented over two external classes of sensory channels: visual and auditory. Associated with each of these external classes of channels is a special channel which is referred to as an immediate memory for information of that class. In addition there is a special class of channels referred to as "semantic" channels. The semantic channels and the immediate memory channels are collectively referred to as "working memory" (WM). Each production is associated with a channel or channel class. Only stimuli from external sources can be placed in the external channels. The working memory channels can be written to only by the action side of a production.

Each pattern in a production is an ordered string of features. A stimulus consists of one or more patterns. The initial stimulus (patterns

and pattern features) is read into the program along with stimulus feature noise levels. Noise levels are used when comparing the stimulus features with the production pattern features. The initial stimulus is placed in specified external channels.

B. Response Queue

The response queue contains all the actions which have been initiated during the previous program cycle. At the beginning of each cycle, the queue is examined and the appropriate action is executed. Details of possible actions are explained in section E.

C. Production Activation

A match is computed between the pattern part of each production and the stimulus on the appropriate channels. A pattern always specifies that it is to be matched to a channel class, and may specify a particular channel within that class. The matching function uses the confusion matrix to weight heavily the most likely pattern matches and to weight lightly the least likely patterns. "1", a most likely pattern would be:

see "1" ⟶ recognize "1"

and a least likely pattern would be:

see "2" ⟶ recognize "2"

Stimulus "7", being somewhat similar to "1", would be an intermediate case. Within the pattern part of each production, stimulus features are weighted by their importance for that pattern.

The strength of the match plus a noise term (a random number that is scaled by the internal noise input parameter) determines the activation level, y[i], of the ith production. If the activation level is greater than the threshold level, the production is considered active and is placed in a set of active productions, {act(y[i])}. If it is not greater than the threshold, the activation level is set to zero.

In this process, all stimuli are compared to all productions in the appropriate channel class.

D. Decision Rule

When production activation processing has been completed, all the active productions are searched to identify the highest activation level. If this most active production exceeds all the other productions by some decision rule variable, DR (an input parameter), the action half of the production is placed in the response queue.

E. Actions in the Response Queue

Actions either:

1. Make an external response. A response will terminate production processing.
2. Place an effective stimuli in one of the channels in working memory.
3. Alter an internal parameter of the program (such as altering the threshold level of a production).

Actions are seen as taking place in stages that extend over time. Once the action is initiated, it proceeds, one stage during each time cycle, in parallel with any other actions that may be being executed at the same time. Actions can not contain any branches or decision points.

F. Priming

After the decision rule has been processed, the activation levels of all the productions are "primed". Using the link formed between the productions by the association matrix, y[i] is either increased (when the link is positive), decreased (when the link is negative) or not affected (if there is no link). Essentially, a weighted sum of the activation levels of the other productions is added to the ith production's activation level, y[i].

G. Decaying

The activation level is also reduced by a delta value, D, another input parameter. Delta is always greater than zero but never greater than one. The decay rule is:

$$y[i] = D * y[i]$$

H. Time Cycling

After the priming and decaying of the activation levels has occurred, time is incremented. If the time then exceeds the maximum processing time specified during initialization, Production processing halts. Otherwise the program checks the response queue and continues processing the productions.

Preliminary Results

The MIND program has been used to recreate several of the most reliable findings observed in laboratory studies. As the purpose of the simulation experiments was to evaluate the psychological reasonableness of the interpreter, we sought situations in which the production systems to be interpreted were, at the program level, as simple a psychological model as possible. The logic of this approach is similar to the logic behind use of very simple programs to test the arithmetic capabilities of computer hardware. As is well known, there are probably no situations that dictate the use of one and only one possible model for human behavior. We do feel that the laboratory situations we have studied approach this ideal in varying degrees.

The approach will be illustrated by a study of the "choice reaction time" (CRT) paradigm. A participant's view is shown in Figure 1. The task is to press the button whose number matches the number appearing on the screen. The task as shown is a two choice task, four and eight choice tasks are constructed on the same principle. It is well known that the time to make a choice in a CRT experiment is a logarithmic function of the number of alternative stimuli that may appear (Hick's law).

Figure 2(a) shows a production system for executing a two choice CRT task. Figure 2(b) shows the associated production activation network. The figure illustrates an important principle that is used in constructing our networks. If two productions, A and B, are in the same channel class and are mutually exclusive alternative interpretations of a stimulus, then the productions inhibit each other. However, if production A produces, as its action, a stimulus that might trigger production B, then A primes B. The priming relation may hold for productions in the same or in different channel classes.

Figure 3 presents the results of a simulation of CRT experiments with varying numbers of

choices. Two results are shown, for two different values of the DR parameter. Data from an actual experiment (Taylor, 1982) are also shown. The number of cycles required by the MIND program was approximately a linear function of the logarithm of the number of choices, but departed from linearity slightly at the 8 choice point. Reaction times from the psychological study showed the same pattern.

Another characteristic of CRT experiments is the "speed-accuracy trade off". For a given individual and condition, the faster a response is made the more likely an error is to occur. When accuracy is plotted against reaction time the function is almost invariably negatively accellerated (Pachella, 1974). Figure 4 shows a speed-accuracy curve obtained from MIND by varying the value of the DR parameter, while keeping all other parameters and the number of choices constant. The program clearly matched the function found in human data.

The speed-accuracy tradeoff describes the relation between accuracy and latency for a given individual and condition. When one changes either the individuals being tested or the experimental conditions, speed and accuracy are often positively correlated. For instance, older people tend to perform more slowly in CRT tasks, and to make more errors (Welford, 1977). This result was simulated by holding DR constant constant, and varying the internal noise parameter. The results are shown in Figure 5. Again the pattern is similar to that obtained in the laboratory.

MIND has been used to simulate a number of other results from the literature in experimental psychology. These include the Stroop phenomenon (Stroop, 1935), the effects of repetition of the same stimulus over trials in CRT paradigms, and interference effects when two tasks are done simultaneously. These results will be reported in a larger paper. While we do not claim to have modeled the microstructure of all these phenomena perfectly, the initial results are encouraging. The ranges of parameter values that are adequate to simulate one task overlap considerably with those required in other tasks. This is a particularly encouraging finding. It appears that the values of the parameters of this model must be held to a rather tight range if the model is to work at all, but that within this range reasonable results can be obtained.

ACKNOWLEDGEMENT: The research reported here was supported by the Office of Naval Research, Contract N00014-80-C-0631, to the University of Washington. We are glad to thank Dr. Marcy Lansman for her constructive comments and criticisms.

REFERENCES

Baddeley, A. D. The Psychology of Memory. New York: Basic Books, 1976.

Broadbent, D. E. Decision and Stress. New York: Academic Press, 1971.

McClelland, J. L. On the time relations of mental processes: An examination of systems of processes in cascade. Psychol. Rev., 1979, 86, 187-330.

Pachella, R. G. The interpretation of reaction time in information processing research. In B. H. Kantowitz (ed.) Human Information Processing: Tutorials in Performance and Cognition. Hillsdale, N. J. Lawrence Erlbaum Associates, 1974.

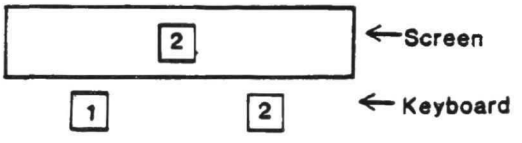
Posner, M. I. Chronometric explorations of mind. Hillsdale, N. J.: Lawrence Erlbaum Associates, 1978.

Stroop, J. R. Studies of interference in serial verbal reactions. J. Experimental Psychology. 1935, 18, 643-662.

Taylor, J. The effects of benzodiazepines on cognition and performance. U. of Washington Ph. D. thesis (Psychology). 1982.

Welford, A. T. Motor performance. In J. E. Birren and K. W. Schaie (ed.) Handbook of the Psychology of Aging. New York: Van Nostran Reinhold, 1977, pg. 450-477.

FIGURE 1



CHOICE REACTION TASK

Production Rules

Visual	**Channel**	**1 = 1**	**⟶ Put S1 in Semantics**
	Channel	**1 = 2**	**⟶ Put S2 in Semantics**
Semantic	**S1**		**⟶ Make response 1**
	S2		**⟶ Make response 2**

Figure 2 (a) Simulation of CRT experiment

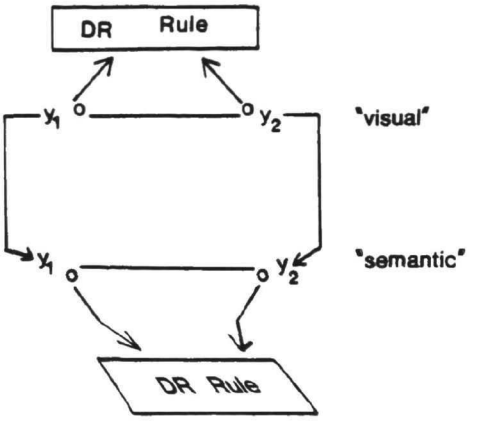
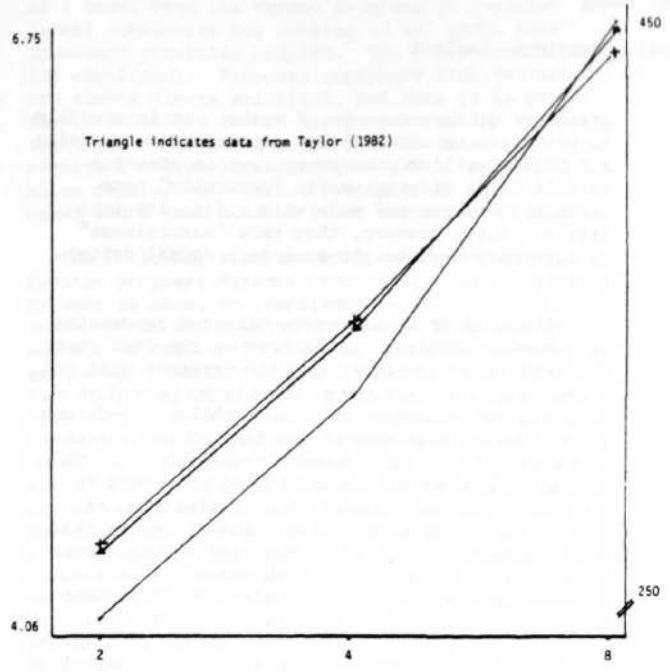


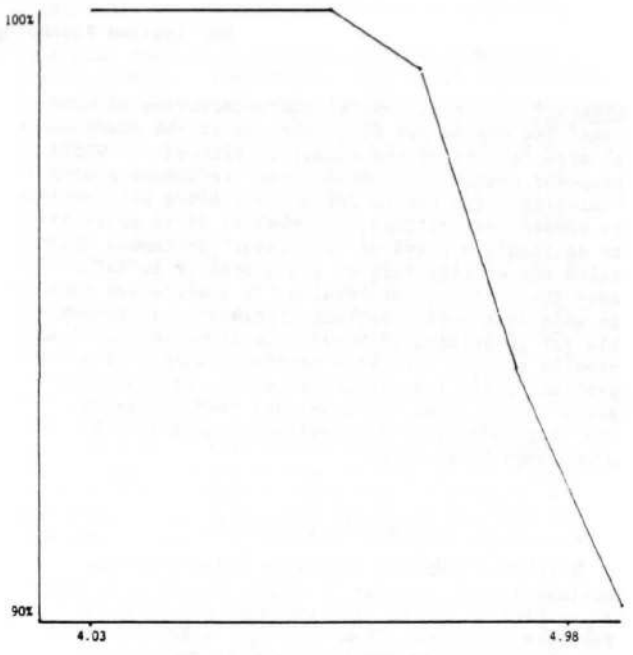
Figure 2(b) Association Network For 2 choice Task

Figure 3



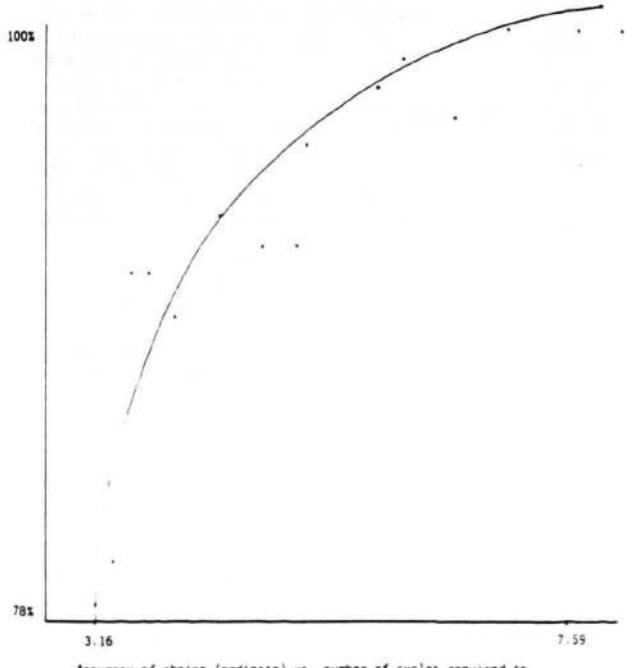
Number of cycles to decision point (left ordinate) or reaction time (right ordinate, in milliseconds) as a function of the number of choices

Figure 5



Accuracy of choice (ordinate) vs. number of cycles (Abelissa). Each point represents a different value of the internal noise parameter. Two choice task.

Figure 4



Accuracy of choice (ordinate) vs. number of cycles required to reach a decision. Each point represents a different value of the DR parameter.

ARCHITECTURE-DIRECTED PROCESSING

Richard M. Young

MRC Applied Psychology Unit, Cambridge, England

Abstract. Certain general characteristics of human cognition may be due to properties of the functional architecture of the cognitive processor. While proposed cognitive architectures are almost always "universal" and can be forced to execute arbitrarily chosen computations, nonetheless it is possible to delineate a class of "compliant" processes that allow the architecture of the processor to influence the course of processing. A speculative case is made that such compliant processing is responsible for invariants of human cognition, such as that problem solving occurs as heuristic search in a problem space, that long-term memory search takes place in cycles of retrieval and re-description, and that uncertain information is dealt with by prominence heuristics.

Compliant processes

A central theme in Cognitive Science is the explanation of features of human cognition in terms of properties of the programs that generate and regulate behaviour. The paradigm is to account for the empirical phenomena observed in some domain, e.g. the time taken to decide the truth or falsity of simple propositions, by showing that they derive from properties of the processes responsible for the behaviour. For all its undoubted merits, there is a gap at the heart of this approach. Although such computational explanations have genuine scientific value, for example by offering a single coherent account for a range of apparently diverse phenomena, there is a need also to try to understand why those particular programs are found but not conceivable others.

The idea explored in this paper is that the functional architecture of the processor itself influences and constrains the kind of programs it can execute, and hence leads to invariants in the resulting behaviour. There is little novelty in this idea: all I hope to do here is to draw together a number of threads from various places. The idea derives mainly from the work of Pylyshyn (1980) and especially Newell (1973, 1980). Pylyshyn (1980) discusses the notion of the functional cognitive architecture, i.e. the fixed structural properties of the human cognitive system. Building on that notion, we extend it to the properties of the processes that the architecture supports. The argument is inspired by, and is closely similar to, that of Moore & Newell (1974). In describing a system called Merlin built round a single processing mechanism, that of assimilation by analogy, they suggest that certain general problem solving methods (Generate and Test, Heuristic Search, etc.) arise within Merlin as "natural methods". In other words, Merlin exhibits these methods not because it runs a program directing it to do so, but because they arise as consequences of its single processing technique. In a similar way, this paper is proposing that certain general characteristics of human cognition arise as "natural methods" from the functional architecture of the cognitive processor.

It may be helpful to consider an analogy, both to understand the idea better and also to highlight its idiosyncracies. Most of us are familiar with the idea that different programming languages lend themselves selectively to different sorts of programs for different sorts of tasks. It is possible in principle to use LISP for commercial programming and COBOL for list processing, but in practice certain kinds of program fit "naturally" into certain languages and only with difficulty into others. Note, however, that this "naturalness" is extremely hard to pin down in a formal definition.

The notion of architecture-directed processing is somewhat similar. It centres on the idea that for a given architecture certain programs will run "naturally", while others can only be coaxed on with a sledgehammer. However, architecture-directed processing goes beyond the idea of naturalness, since it allows the architecture to influence the actual selection and sequencing of the steps to be taken. To some extent this is also true of programming languages. With ordinary sequential flow languages, such as FORTRAN and PASCAL, there is a kind of "default" control structure (i.e. execute the next statement) which the programmer can override when she wants to (by iterations, jumps, subroutines, and so on). However, in normal practice programmers use this sequential control in order deliberately to specify the order of execution, so to regard it as a default is a little misleading. With architecture-directed processing the influence is more pervasive, since at least for certain production system architectures (PSAs) (Newell, 1973, 1980; Anderson, 1976; Waterman & Hayes-Roth, 1978) the program does not have to specify an order of execution at all. Once the repertoire of possible steps has been supplied, the selection and sequencing can be left to the architecture, which appropriate PSAs can perform in a highly flexible manner responsive to the particulars of the task (Young, 1977, 1979). The program can still, of course, specify the control structure where it needs to. This freedom leads to the idea that responsibility for the flow of control has been split between the program and the architecture. It follows that programs will differ in the extent to which they insist upon a particular control regime. Programs that allow the architecture to have largely its own way we will call compliant. Not to be taken too seriously, but as a starting point, we can offer a tentative

> Definition. A program is "compliant" to the extent that it allows the selection and sequencing of steps to be determined by the architecture it runs on.

The examples given below will try to demonstrate that, given an architecture, compliancy leads to the appearance of certain invariants in the generated behaviour.

It's hard to be precise...

In one important respect this notion of "compliancy" is very similar to the idea of "naturalness" in programming languages, and that is in the difficulty of making it more precise. Despite our recognition of the selective suitabilities of different languages for different kinds of programs, it remains the case that the languages are almost always computationally "universal", and therefore formally equivalent in power. It follows that any program can, in principle, be written in any of the languages, and that it is

hard or impossible to capture the idea of "naturalness" in a formally precise way. So far as I know, even the recent progress in computational complexity has nothing to say about this important practical problem. The story is similar for compliancy. Proposed cognitive architectures are almost always universal, and thus it is possible in principle to run any program on any architecture. In most cases, of course, this will require a non-compliant program which imposes an alien control structure. Our interest is in the cases where this kind of brute force is not needed.

The following examples will make clear that further progress depends upon being able to specify what is meant by compliancy in more precise terms. I am not totally optimistic that we will succeed in this, but I can see two avenues worth exploring. The first is to take advantage of the fact that compliancy has to do specifically with flow of control. There is a sense in which the steps of compliant programs execute at "base level", whereas non-compliant programs require an extra level of interpretation. If this difference can be captured reasonably precisely, there is some hope of deriving the consequences of compliancy in a more rigorous way. The second possibility depends on achieving some understanding of the mechanisms by which new programs are acquired. This might provide a much stronger basis for placing constraints on the kinds of program the cognitive processor will run: not that non-compliant programs are "unnatural", but by showing that only compliant programs could ever be learned.

Examples

There follow three examples to illustrate how compliant programs lead to the appearance in behaviour of certain invariants dictated by the underlining architecture. Two warnings need to be given. One is that the difficulty of making the notion of compliancy even moderately rigorous makes it impossible in any strict sense to derive the invariants from the architecture. The arguments given, though intended to be plausible, have to be regarded as hand-waving. The other is that these examples are speculative. I would not wish to give the impression that the arguments are summaries of a more complete story already worked out. Rather they should be regarded as the goals for a programme of work still to be undertaken.

Ex.1: Problem solving is carried out by heuristic search in a problem space. That assertion can reasonably be taken as the one-sentence conclusion of Newell & Simon's (1972) study of human problem solving. It arises as a consequence of compliant programs running on PSAs of certain types. This is the clearest of the three examples, and the argument is essentially due to Newell (personal communication).

Consider a PSA which is like OPS (Forgy & McDermott, 1977) in the following respect. Whenever more than one production rule is applicable, the one to fire is determined by the following principles ("conflict resolution"). (1) Recency: rules whose conditions are sensitive to more recent information take priority over those matching only older information. (2) Special case: rules which are special cases of other rules take priority over them. (For further details see McDermott & Forgy, 1978; Forgy & McDermott, 1977). Suppose that knowledge of the problem domain is coded by specifying the possible moves that can be taken in circumstances C as rules like:

Rule 1: C & <? side conditions> = <action1>
Rule 2: C & <? side conditions> = <action2>,
etc.

Note that such rules provide a highly compliant representation. They impose only local constraints on how they are used, and thus have individually, as it were, no opinion about the more global flow of control. Suppose that C is known. Then one of the rules shown will fire, Rule1 say. If the action taken leads to some new information and there exist rules responsive to that information, then by the recency principle it will be one of those rules that fires next. And so it continues, as long as there is new information and rules to respond to it. Once that is no longer so, processing falls back, say to the rules shown, and one of the alternative rules at that level will fire; in this case, Rule2. In other words, a depth-first search is performed. On the other hand, if at any time a rule which is sensitive to a particular configuration of information becomes satisfied, then by special case it will be the one to fire. In other words, specific knowledge is brought to bear when appropriate. The upshot of all this is that the principle of recency generates depth-first search, while special case adds heuristic guidance.

It is worth emphasising the contrast between this explanation and virtually all earlier accounts in the cognitive modelling literature (including, for example, Newell & Simon, 1972). We have just argued that people solve problems by heuristic search, not because they run a "heuristic search program", but because, in the absence of guidance to the contrary -- i.e. with a compliant program -- heuristic search is the natural thing for the PSA to do.

Ex.2: Indirect recall from long term memory. When the cues presented are insufficient to elicit some target information from long term memory directly, both theory (Norman & Bobrow, 1979) and the experiment (Williams & Hollan, 1981) suggest that recall occurs in a series of cycles of alternating retrieval and re-description.

Again, this behaviour is a consequence of the conflict resolution principles of a PSA. Suppose that the target information is on the action side of a rule. Then by supposition, not all the information on its condition side is yet present (the point is to gather it so that the rule does fire). Whatever information is present, constituting a partial description of the item being sought, will trigger some rule or other. This in turn will add to the description. Special case ensures that each item retrieved is relevant to the current description; if there is no relevant information, then general procedural heuristics will fire. As in problem solving, the recency principle ensures that newly retrieved information is followed up first.

Ex.3: Uncertain information is dealt with by "prominence" heuristics (Fox, 1980a), such as representativeness and availability (Tversky & Kahneman, 1974). The implied contrast is with rational, non-heuristic techniques such as the use of Bayes' theorem and the maximisation of expected value. For this example we have to move beyond the OPS architecture, to a PSA which assigns different strengths to different items, and thereby recognises a degree of match between the data and a rule. Examples are HPSA (Newell, 1980) and the PSYCO architecture used for simulating medical

diagnosis (Fox, 1979, 1980b). The argument essentially follows those two authors.

The key issue is the representation of the degree of certainty. If it is coded explictly as simply another component of the data,
e.g. (DISEASE-IS GASTRIC-ULCER CF = 0.7),
then it will be treated as part of the information content by whatever rules happen to process it, and no consequences follow from the architecture. If, on the other hand, certainty is coded as the strength of the item,
(DISEASE-IS GASTRIC-ULCER) [0.7],
then the certainty has effects at the level of the architecture (i.e. it appears as an aspect of the form rather than the content of the item), and influences processing at this level. What happens of course is that certainty enters as a factor in conflict resolution, with stronger items, other things being equal, being processed before weaker ones. The outcome is that processing of uncertain information is dominated by the data that for whatever reason are more "prominent" in memory (Fox, 1980a). Items which are highly familiar, already in working memory, or more closely linked to other relevant items will be the first to come to mind and will carry more than their fair share of responsibility for guiding behaviour.

References

Anderson, J. R. (1976) Language, Memory and Thought. Erlbaum.

Forgy, C. L. & McDermott, J. (1977a) OPS, a domain-independent production system language. Proceedings of the 5th International Joint Conference on Artificial Intelligence, 933-939.

Forgy, C. L. & McDermott, J. (1977b) The OPS2 reference manual. Technical Report, Department of Computer Science, Carnegie-Mellon University.

Fox, J. (1979) Medical diagnosis: Inference, recall and a theory of skill. Unpublished ms.

Fox, J. (1980a) Making decisions under the influence of memory. Psychological Review, 87, 190-211.

Fox, J. (1980b) The PSYCO manual. MRC Social and Applied Psychology Unit, University of Sheffield,, England.

McDermott, J. & Forgy, L. (1978) Production system conflict resolution strategies. In Waterman and Hayes-Roth (1978), 177-179.

Moore, J. & Newell, A. (1974) How can Merlin understand? In L. W. Gregg (Ed.), Knowledge and Cognition, 201-252. Erlbaum.

Newell, A. (1973) You can't play 20 questions with Nature and win: Projective comments on the papers of this symposium. In W. G. Chase (Ed.), Visual Information Processing, 283-308. Academic Press.

Newell, A. (1980) HARPY, production systems and human cognition. In R. Cole (Ed.), Perception and Production of Fluent Speech. Erlbaum.

Newell, A. & Simon, H. A. (1972) Human Problem Solving. Prentice-Hall.

Norman D. A. & Bobrow, D. G. (1979) Descriptions: An intermediate stage in memory retrieval. Cognitive Psychology, 11, 107-123.

Pylyshyn, Z. (1980) Computation and cognition: Issues in the foundation of cognitive science. Behavioural and Brain Sciences, 3, 111-169.

Tversky, A. & Kahneman D. (1974) Judgement under uncertainty: heuristics and biases. Science, 185, 1124-1131.

Waterman, D. A. & Hayes-Roth, F. (1978) Pattern-Directed Inference Systems. Academic Press.

Williams, M. D. & Hollan, J. D. (1981) The process of retrieval from very long-term memory. Cognitive Science, 5, 87-119.

Young, R. M. (1977) Mixtures of strategies in structurally adaptive production systems: Examples from seriation and subtraction. Proceedings of workshop on pattern-directed inference systems. SIGART Newsletter No. 63, June, 65-71.

Young, R. M. (1979) Production systems for modelling human cognition. In D. Michie (Ed.), Expert Systems in the Microelectronic Age, 35-45. Edinburgh University Press.

Question Answering: Two Separate Processes *

Marc Luria

Division of Computer Science
Department of EECS
University of California, Berkeley
Berkeley, Ca. 94720

1. Introduction

I have developed a question answering program that will answer questions about simple stories. In my program, question-answering is divided up into two separate processes: 1) answering formation and 2) answer expression. The program first looks down a causal chain which is formed by the story-understanding program and figures out in what part of the chain the answer lies. The answer can also be a subset of the chain, sometimes a quite long one. The second part of the program takes this long chain and decides what things are important to express to the questioner. This answer expresser uses general rules of expression to figure out what it needs to include to make the answer understandable, informative and interesting.

This solution is different from other question-answering algorithms (e.g. Winograd 1972, Lehnert 1977) which view question answering as one process. These programs gather possible answers, and then choose the 'best' answer from among them. My system first gets the chain which I consider to be the answer to the question, and then figures out which parts of the chain should be generated into English as the answer. The advantage of my approach is that it allows one to treat the answer as one entity and use the answer expression mechanism to express what people are interested in. The resulting answers are generally more informative and conversationally appropriate than those generated by other algorithms.

The program works in conjunction with PAMELA, a story understanding program that specialises in goal-based stories. (Wilensky 1977, Wilensky 1981, Norvig 1982) After a story is initially 'read' by PHRAN, a parser, (Arens 1981) it is then passed to this PAMELA and 'understood'. The question answering program is passed a database which consists of events, inferences, and most importantly, for my purpose, causal chains which instantiate events in the story as steps of particular plans and plans for particular goals. Contained in this causal chain is the actual 'understanding' of the sequence of events in the story, what caused what, and what goal actors had in mind when they performed a particular act or plan. After a question is asked, this question is parsed by the same parser that parsed the story, and then the answer is formulated by looking at the database. Finally, the answer is passed to the answer expresser which sends the answer to a natural language generator.

2. Program Examples*

The following examples were processed by my program.

Story: Susan saved her money from her allowance. One day she rode her bike to the bookstore and bought the book that her teacher had recommended. Susan did very well on her math test the following week.

Q: Why did Susan buy the book?
A: So that she could study from it and do well on her exam.

Q: How did Susan do so well on her math exam.
A: She bought a book that her teacher had recommended and studied from it.

Q: How did she get the book?
A: By riding her bike to the bookstore.

3. Finding the Best Answer

A difficult and important part of answering a question is not in finding an answer to the question, but finding the best answer. In a database of causal chains, if one can find an event in the database then there may be many possible answers to a given question. Consider the previous story.

If we ask:

Question2: Why did Susan buy the book?

The following answers are obtained by stepping at different on the causal chain.

Answer2a: Because she wanted to have it.
Answer2b: Because she wanted to read it.
Answer2c: Because she wanted to know math.
Answer2d: Because she wanted to do well on her exam.

Note that the items nearer the top of the goal structure constitute better answers although the best answer would be something like:

Answer2e: So that she could study it and do well on her math test.

However, in a more complicated story, merely looking to the end of chain might not work quite as well. For example, if in the previous story we added:

She put the book on her head and learned the material through osmosis. Susan did very well on her math test the following week.

Clearly, Answer2d is no longer a good answer.

One possible solution including only 'important' answers. Important inferences might include abnormal plans, natural disasters, etc. The problem with this was that even though these 'important' inferences definitely should be included in the answer, one should not necessarily stop at that point in the chain and say that this is the answer. For example, just stopping at 'important' events in response to question2 one would get:

Answer2f: So that she could put it put it on her head.
Answer2g: So that she could learn by osmosis.

which is less desirable than:
Answer2h: So that she could learn from the math book by osmosis and do well on her exam.

4. Dividing up the Question Answering Process

My program is able to find these better answers because of the separation of finding the answer (the subset of the chain) from expressing the answer to the user. Instead I use the two programs:

Answer-Formulator: looks down a causal chain, figures out where what parts of the chain are relevant to an answer and returns a chain.

Intelligent-Expresser: takes this causal chain as input, figures out from its general rules of expression what

*This research was sponsored in part by the Office of Naval Research under contract N00014-80-C-0732 and the National Science Foundation under grant MCS79-08543.

*At this point the program is not connected to the natural language parser at Berkeley called PHRAN or the generator PHRED (Wilensky and Arens, 1980). The questions and answers are therefore translated from the conceptual form I now use.

is important to say so that the questioner will a) understand the answer and b) get the kind of information that people are generally interested in, and outputs to a natural language generator, some intermediate form from which it could generate an answer.

For example, my program would produce answer2e above by the following process. First it would find Susan buying the book in the database and then follow the chain, in this case, to where it finds that she did well on her exam. This whole part of the chain would be passed to the expression mechanism which would notice that studying the book and doing well on her exam were important parts of the answer. In this case, the Intelligent-Expresser uses the general conversational rule of not informming someone of something they already know. Having the book and reading the book are thereby eliminated because they are stored in the data base as normative purposes for buying and for having a book, respectively.

This approach also allows one to generate answers that were otherwise problematic to represent in a conceptual form. For example, the simple question:

Question3: Did Susan go to the bookstore?
Answer3: Yes, she rode her bike there.

The answer is obviously yes, because this event appears in the database. However, 'yes' is something that is difficult to represent in conceptual form. 'Yes' is not really a concept but rather a word that is almost exclusively used in a conversation. The answer formation part of my system looks in the database for concepts similar to going to the bookstore. Realizing that riding to the bookstore was similar to going there it would answer:

```
(ride
  (actor (person (object susan1)))
  (object (bicycle (object bicycle1)))
  (destination (bookstore (object bookstore1))))
```

This part of the chain and the context in which the question was asked is passed to the answer expression part of the program, that would a) see that this is a simple verify question, b) realize that the concept to be verified was in fact found in the database in a slightly different form and c) figure out that it should answer 'yes' plus some intermediate form that represents that it should include the ride concept.

This same method can be extended to other types of verify questions. For example,

Question4: Did Susan ride her bike to the bookstore so that she could do well on her math test?
Answer4: Yes, she bought a book at the bookstore which she used to study for her exam.

Question5: Did Susan buy the math book so that she could do well on her math test?
Answer5: Yes, she used it to study for her exam.

The answer formation part looks to see if a chain with the starting place of 'riding to the bookstore' and ends with 'doing well on her math test', exists in the database. This whole chain does exist and includes, she rode to the bookstore was a plan for being at the bookstore, which was a precondition for buying a book, which was a plan for having the book which was a step of reading the book, which was a plan for knowing the math material, which was a goal from doing well on her exam.

The answer expression part of the program gets this chain, realizes it should answer 'yes' and decides how much in addition to the 'yes' it would need to include in the answer. Notice how in Answer4 it had to include more information from this chain than it had to include in Answer5.

5. Conclusion

This intelligent expression part of the program is not something that is designed to be used exclusively in question-answering but would be a system that would be valuable in any context where an interactive natural language system would be important. It differs from a generator in that it does not merely generate something from a conceptual form into English, but rather decides what kinds of things are important to be said, which is then passed to a generator. Hopefully, this kind of system could be expanded to work on other conversational tasks as well.

References

Lehnert, W., 1978. *The Process of Question Answering: A Computer Simulation of Cognition*. Hillsdale, N.J. Lawrence Erlbaum Associates, Inc.

Wilensky, R. , 1978. *Understanding Goal-Based Stories*. Technical Report 140, Computer Science Department, Yale University, New Haven, CT.

Wilensky, R. and Arens, Y. 1980 *PHRAN - a Knowledge Based Approach to Natural Language Analysis*. University of California at Berkeley. Electronic Research Laboratory Memorandum No. UCB/ERL M80/34.

Wilensky, R. 1981. Meta-planning: Representing and using knowledge about planning in problem solving and natural language understanding. *Cognitive Science*, Vol. 5, No. 3. 1981.

Winograd, T. 1972 *Understanding Natural Language*. New York. Academic Press.

Exploded Connections:
Unchunking Schematic Knowledge

Steven L. Small
Department of Computer Science
The University of Rochester
Rochester, New York 14627

Background

It has been understood for some time that the organization of knowledge into event schemata and visual schemata can aid significantly in the inference-making process. If we know that we are in a typical room, to use Minsky's example [1974], then we expect to see windows, walls that are perpendicular to a ceiling, etc. If we know that we are at a restaurant, to use Schank's example [1975], then we can expect to be seated by a maitre d'hotel, to be given menus, etc. By classifying situations according to a small collection of schematic situations, a wide variety of inferences become immmediately clear and simple.

This same kind of schematic reasoning constitutes the heart of several well-known theories of low-level comprehension, especially by Schank [1972], Wilks [1973], etc. By classifying linguistic clauses into a small number of semantic categories, such as physical transfers of location (PTRANS), propelling of objects (PROPEL), etc., a number of inferences are straightforward. Certain kinds of paraphrase are simple: "buying" and "selling" are represented in almost the same way; "running," "walking," and "biking" have much in common in their semantic representations. The schema for abstract transfers of possession (ATRANS) leads us to expect exchange of one thing for another from one person to another. If any of these are not specifically specified, they can be inferred easily. Further, such an abstract transfer probably took place because one person wanted to own something that had been owned by someone else, the other person probably didn't want it so much anymore, and similar kinds of simple inferences. The MARGIE system [Schank et al, 1973] exhibited very impressive behavior without using much more than schematic inferences based on the semantic representation scheme of Conceptual Dependency (CD).

Unchunking Schemata

In this short paper, we suggest a framework for the study of schematic aspects of natural language comprehension. Specifically, we pursue the fact of Feldman [1976] in preferring the dynamic rather than static chunking of knowledge; the use of parallel [illegible] knowledge representation facilitates that goal. The approach draws from previous work in schematic representation and reasoning [Minsky, 1974; Schank, 1975], spreading activation [Quillian, 1968], parsing [Small, 1980; Marcus, 1979], speech recognition [Lowerre, 1976], psycholinguistics [Dell, 1980; McClelland and Rumelhart, 1980], and computer vision [Ballard, 1981; Marr, 1978]. By decomposing schematic knowledge into diffuse units and by studying the way these facets of knowledge are connected (inferentially), we expect to show important results in several areas:

- how a language comprehension system can maintain diffuse loci of control (hypotheses) simultaneously, but still come to a decision when required;
- how to obtain schematic reasoning (and expectations) from distributed units not a priori committed to representing unique static situations;
- how to merge schematic inference mechanisms from the top-down (e.g., scripts) and from the bottom-up (e.g., case frames); and
- how to relate experimental psychological data (e.g., reaction times on normals and aphasics) to computer models.

The modelling effort employs an architecture significantly different from the typical computer and closer to that of the human brain. We use a particular spreading activation or active semantic network scheme, called *connectionism*, which consists of a massive number of appropriately connected computing units that communicate through weighted levels of excitation and inhibition [Feldman and Ballard, 1982]. While such an architecture does not solve any problems per se, we believe that a number of questions become easier to set forth and more straightforward to solve. This paper intends only to suggest the directions of our current research in addressing several language comprehension issues from the new perspective.

Some Main Issues

In particular, we show how a number of classical problems in the theory of schemata might be approached in a new way. Three principal issues are discussed: (1) Comprehension takes place on a number of interacting levels of processing; (2) multiple hypotheses are simultaneously maintained at a number of diffuse processing loci; and (3) context affects processing in both top-down and bottom-up directions. Experimental psychologists are beginning to understand these issues through reaction time data. McClelland and Rumelhart [1980] have identified two processing levels; Dell [1980] presents data suggesting interactions within the phonemic level; Swinney [1979] shows ways in which context *does not* affect processing; Seidenberg et al [1980] illustrate an entire time course for processing at the lexical level; and Samuel et al [1982] present data suggesting the mechanisms of letter processing and the word superiority effect.

Multiple Levels of Comprehension

While it is sometimes the case that a language understander needs to know the primitive schematic actions that compose a more complex action, often he does not. The relevant information carried by particular words and expressions is precisely that information that aids the hearer to understand the intended meaning of the speaker. This always takes place in some context and cannot be separated from it. In a dialogue, a hearer must interpret the words and expressions in light of the communicative goals of the speaker; in a story, a reader must serve to connect new fragments of text with the existing interpretation of story structure. Further, general knowledge about the world must be applied where needed

to the comprehension process (even at the level of individual words and phrases), and the story or goal structures constructed must be constrained by it.

It seems unusual to consider certain actions in terms of their decompositions (in the sense of CD) into structures of primitive units. There are very few contexts in which the sentence "Rick kissed Joanie" would be best understood by focusing on the (nonetheless valid) fact that "Rick moved his head to in front of the head of Joanie so that they were both facing each other, and then puckered his lips and touched them to Joanie." This long description must be represented as the algorithmic (functional) concept underlying kissing. This description would be required to understand the sentence "Joanie caught Rick's cold" occurring next. It would certainly not help in understanding the sentence "Joanie bought herself a new blouse." The understanding of this second conceivable utterance requires an entirely different set of relationships concerning kissing.

Multiple Simultaneous Loci of Control

Thus, we need at least two different kinds of associations of kissing to understand sentences in which it is a central action. When hearing such a sentence, both of these kinds of associations are activated, and either one can be relevant to understanding what comes next. Furthermore, the context previous to the kissing action could serve to make one or the other of these associations the prominent one [illegible] understanding. The sentence "Rick didn't care about the flu" would facilitate understanding the next sentence in a way more heavily weighed toward the algorithmic association of kissing than the emotional one. Likewise, the preceding sentence "Rick felt strongly affected" should facilitate the other associations. An active processing network works through simultanous activity in many processing locations, permitting a cognitive model to avoid irrevocable all-or-none decisions in favor of a more continuous approach. This leads to plausible explanations of subsequent context effects, including puns.

Context Effects on Comprehension

In building a computer model of language comprehension, we must consider these phenomena. The context preceding an utterance must serve to favor certain interpretations over others. As in the example presented here, the competing interpretations need not be incompatible; the context must simply facilitate the comprehension of subsequent utterances by focusing on one level of interpretation over others. It should take longer for a hearer to understand how Rick could get the flu from Joanie if conditioned to focus on their emotional involvement, than to understand the same utterance after contextual conditioning to focus on the mechanics of kissing. The results of Seidenberg et al [1980] suggest that analogous contextual effects hold with respect to lexical access.

The computer model should make accessible all levels of interpretation of utterances, but should not make everything as easily accessible as everything else. When knowledge of the mechanics of kissing are required to understand a fragment of text, it must be available. If the text is about some romantic relationship, this knowledge would usually be an obstacle, rather than a help, to understanding. In such a case, it should be available if needed (though perhaps slowly), but mostly it should not be involved.

The Exploded Connection Scheme

We propose a uniform representation scheme for both high- and low-level language processing that shares some of the flavor of the schematic approaches, but which incorporates flexibility through three methods:

(1) The use of incredibly large numbers (and wide variety) of schematic situations (*units*);

(2) A focus on the relationships among these situations rather than on the situations themselves (*connections*), and

(3) The use of numerical potentials to weigh (comparatively) the relevance of any particular schematic situation to the data (*activation levels*).

We call the approach an *Exploded Connection Scheme* (ECS), and are using it to build a unified theory of low- and high-level language comprehension. The elemental units of ECS encompass the gamut of traditional elements of comprehension models, from phonemes and morphemes to cases and semantic primitives to concepts and event sequences. We believe that there are large numbers of each kind of unit, and that reasoning takes place through the richness of the unit vocabulary and the connections among the individuals. Some of our current ideas on the organization of these exploded units can be found in [Cottrell, 1982].

What we are arguing for is a highly diffuse active representation of knowledge and its processing. Traditional models [Schank et al, 1973; Small, 1980] represent the meaning of utterances in single, large, complex structures of some small number of primitive elements. Large processes then manipulate this knowledge, encoding and decoding the large symbol structures. Alternatively, we are suggesting representing meaning in a very large number of (exploded) active processing units, which compute activation as a function of incoming weighted excitation and inhibition. The scheme focusses on the complex interactions among diffuse knowledge units and reduces the complexity of individual processes. Such processing units that do not manipulate complex symbol structures can interact frequently and tightly.

The Pair Principle

Each unit of a particular type triggers activation in other similar units that are likely to come next in meaningful speech. This happens at every level of processing, within the level itself, and is called the *pair principle*. The principle states that every element of knowledge triggers other elements (of the same kind) that are likely to succeed the given one (temporally or inferentially) in meaningful speech. Dell [1980] shows a model for speech production in which connections according to this pair principle lead to plausible explanations of experimental results in production of speech errors. By adding connections between levels, activity spreads through the network in a way that leads to predictions at every level of processing about what is coming next. The spreading activation model of McClelland and Rumelhart [1980] shows such predictions through grapheme/lexeme interactions. The results of Samuel [1980] on word superiority also support this view.

Examples of the pair principle can be seen at every conceivable level of language processing. A phonemic unit activates those other phonemic units that can follow the given one under the rules of the phonological system of a particular language (or the phonologic rules known to the hearer). A high-level activity unit activates the units for other high-level activities that can reasonably come next under the rules of cultural behavior of a society (or analogously, those rules known to the understander). An

action unit increases the potentials of type units that represent the kinds of things that could be the case fillers of that particular action. The pair principle underlies the way we go about connecting units together within particular levels of the comprehension model.

Schemata

At the level of high-level activities the pairwise connections of units might seem less obvious than at the level of phonemes. The sound pattern of languages are well-known, but not so the cultural regularities. Further, the differences among individuals may seem greater when it comes to their expectations about events in the world as opposed to their use of sounds. We contend that this is not so. The restaurant script of Schank [1975] constitutes a good example of the expectations of people from our culture about the high-level activities involved in eating at a restaurant. Two fundamental problems are known to exist with scripts: (1) how do you know when one is relevant; and (2) how do you use information from one script in understanding activities in another?

Schank [1979] has begun to address these questions in his recent work on MOPs, or memory organization packets. In our way of viewing language comprehension, the Yale group has shifted slightly its emphasis: they are increasing the number and nature of the schematic situations they recognize and they are focusing a bit more on the connections among situations. Their restaurant script is now connected to other schemata representing the general notions of visiting a business establishment, preparing and eating a meal, out-of-house social activities, etc. We agree with this shift, and push it to its logical conclusion, as enumerated above in our three representation methods.

A conceivable pair matrix for several example high-level activities of restaurant-going is shown in Figure 1. We can envision additional pair matrices for each activity found on the right-hand side of the one shown--the entire set of connections being quite large. During the comprehension process, the activation of one activity unit causes concomitant model activity in those that follow it. Further, this pairwise triggering does not stop at the units that are only one connection away, but continues (at a smaller-valued potential) for a good distance, activating a large number of units until the ever-decreasing value is no longer significant.

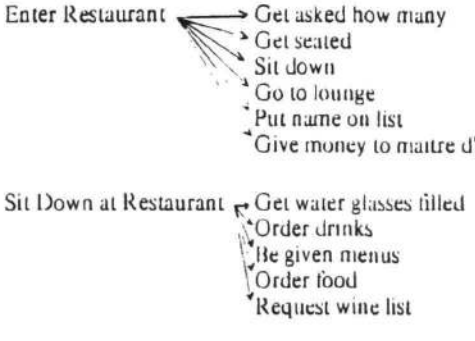


Figure 1. *Activity Pair Relations*

Hierarchy

One kind of inference has always been a problem for schema-based models of comprehension based on static chunks of knowledge and a single processing focus. In the birthday party scenario of Charniak [1972], what does the model do if something happens not directly related to parties? Likewise, in his later exploration of the world of painting [1977], how does a schema driven model understand an event in a story that has nothing to do with painting? An attempted solution within the single process approach has been to use some sort of stack mechanism for the schemata, such that one context gets pushed (to use the computer metaphor) and another takes over. Again, the problem arises: which schema to activate when the previous one is pushed? And when is it popped?

The solution to this classic problem within the Exploded Connection Scheme involves connections among hierarchical levels of active knowledge units in the model. Research in semantic networks has led to interesting epistemologies and computer representations. The work of Fahlman [1979] in particular shows how the different levels of description might be related to each other in our own scheme. The main difference between his NETL approach and ECS centers on the elimination of a central controller in favor of multiple competing processing loci, each with a dynamic activation (confidence) level.

The pair matrix that shows some of the activities involved in going to a restaurant and the activities likely to follow them in everyday circumstances in our culture (Fig. 1) seems specific to that overall activity, i.e., restaurant going. The events listed are exploded, in the sense that they describe "entering a restaurant" and "being seated at a restaurant," rather than "walking" and "sitting down." This explosion means that there are a large number of different units all representing the same kind of activity in different contexts. If we leave things as such, there are many problems of schema-based models that will cause trouble in our scheme. How can we reason that "sitting down in a restaurant" could lead next to a "knee spasm," for example? Whether we represent the knowledge as "the sitting down" action of the "restaurant schema" or as the "sitting down in a restaurant" unit in a connected network, the inference is not possible (unless "knee spasms" are explicitly linked to the "sitting down in a restaurant" unit, an unacceptable solution).

Our solution to this problem is to connect every unit in the exploded network with a number of units that represent the same event less specifically. The method we use, called the *hierarchy principle,* involves representing events in an ever more specific hierarchy, from completely context-free actions, such as ingesting, to very particular ones, such as "eating squid at a Spanish restaurant in Georgetown." A small set containing a few intermediate kinds of eating is illustrated in Figure 2. While it might seem like there are far too many units in this hierarchy to be plausible as a representation of knowledge for comprehension, it is important to realize that: (1) most of the possible units do not exist in each individual; (2) the hierarchy is a tangled one; and (3) only the units at the highest levels are fairly fixed in the nature of what they represent.

Figure 2. *Hierarchical Activities*

Connections and Spreading Activation

Let us refer to the pairwise connections that are based on the temporal order of processing (e.g., phonemes, events) as *temporal connections*, those based on mundane inference as (*mundane*) *inference connections*, and those in the hierarchy as *hierarchical connections*. Sometimes it is convenient to call connections of the first two kinds *follow connections*. The combination of spreading activation along these different pathways provides an answer to the problem with schema-based comprehension mentioned previously. The temporal and inferential order of events now triggers activation in event units in two dimensions, leading in the horizontal direction to expectation of specific schematic events, and in the vertical direction to non-schematic events of a less-specific nature. Note that the vertical activation causes activation horizontally among these more general activity units.

The restaurant-going example can be used to illustrate the nature of this activation. When the "being seated at a restaurant" unit becomes active, a number of event units along the follow connection pathways are also activated. These include such things as "being given a menu," "asking for a wine list," etc., as shown in the pair matrix of Figure 1. That activation in turn causes additional activation at a lower level along the next set of follow connections, and so on, until the ever decreasing activation has become essentially zero.

Simultaneously, however, activation proceeds along the hierarchical connections as well, likewise decreasing for each new radius of connection. Of course, each event unit in the hierarchy has both hierarchical and temporal connections, and activation from hierarchical paths proceeds out along all connections, regardless of type, thus creating a new set of event pair expectations. The way that "being seated in a restaurant" can naturally lead to the comprehension of a "knee spasm" through a combination of hierarchical and temporal pathways is shown in Figure 3. By the decreasing activation idea, the potential (activation level) of "knee spasm" should be significantly less than that for things like "looking at the menu," but that is perfectly consistent with our thoughts on how context strongly affects perception of new inputs.

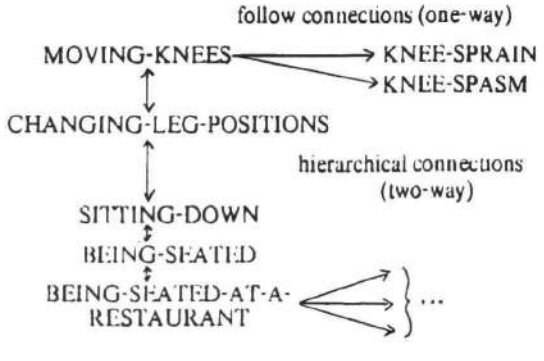


Figure 3. *Schema Interaction*

Furthermore, in cases when stimuli could be interpreted in more than one way, this scheme leads to hypothesis (experimentally testable) about the preferred interpretation. The role of perception is also important in the scheme, since the activation level of units depends on inputs along all dimensions of connectivity in the network. The potential of a unit can be changed by direct perceptual stimulation, stimulation from above or below in the isa hierarchy, from previous events along the follow pathways, or from other sources yet to be identified. The potential represents in a uniform way the stimulation provided by a combination of all incoming connections. The construction of our model involves identifying the nature of connections and units (which we are now doing) and the nature of the combination rules for each kind of unit.

Summary and Conclusions

The research program we are commencing -- the construction of a computer model of human language comprehension -- represents an interdisciplinary effort in cognitive science. The plan involves connecting up a large number of neuron-level computing units to process cohesive text. Empirical constraints on the organization of the active network consist of processing evidence from psychology, physiological evidence about the brain, and computational plausibility. Thus far, we have made some preliminary studies in parsing and schematic reasoning, and have a working network simulator that has been applied successfully to some simple problems in high level constraint relaxation.

In this paper, several issues regarding the organization of schematic knowledge for language comprehension have been described within the connectionist framework. We have suggested mechanisms for (1) obtaining schematic reasoning from diffuse computing units; (2) merging top down and bottom-up control in schematic reasoning; (3) maintaining diffuse loci of control yet coordinating global behaviors; and (4) directly relating psychological evidence to computational models in cognitive science. The results presented are certainly in a preliminary state, but are leading to interesting simulations and valuable collaborative work.

References

Ballard, D.H., "Parameter networks: Towards a theory of low-level vision," *Proc*, 7th IJCAI, Vancouver, B.C., August 1981.

Charniak, E., "Ms. Malaprop, a language comprehension program," *Proc.*, 5th IJCAI, 1977.

Charniak, E., "Toward a model of children's story comprehension," AI Memo 266 AI Lab. MIT 1972.

Cottrel, G., "Toward Connectionist Parsing", Technical Report, Department of Computer Science, University of Rochester, 1982 (to appear).

Dell, G.S., "Phonological and Lexical Encoding in Speech Production", Ph.D. dissertation, Department of Psychology, University of Toronto, 1980.

Fahlman, S.E. *NETL, A System for Representing and Using Real Knowledge.* Boston, MA: MIT Press, 1979.

Fahlman, S.E., D.S. Touretzky, and W. van Roggen, "Cancellation in a parallel semantic network," TR, Computer Science Dept., Carnegie-Mellon U., 1981.

Feldman, J.A., "Bad-mouthing frames," *Proc.*, TINLAP, 1976.

Feldman, J.A. and D.H. Ballard, "Connectionist models and their properties," to appear in *Cognitive Science 6*, 1982.

Hinton, G.E., Review of S.E. Fahlman, *NETL: A System for Representing and Using Real-World Knowledge. AISB Quarterly*, 42-43, Winter/Spring 1981/82.

Lowerre, B.T., "The Harpy Speech Recognition System", Ph.D. dissertation, Department of Computer Science, Carnegie-Mellon University, 1976.

Marcus, M.P., "An Overview of a Theory of Syntactic

Recognition for Natural Language." AI Memo 531, MIT Artificial Inteligence Laboratory, 1979.

Marr, D., "Representing visual information," in A.R. Hanson and E.M. Riseman (Eds). *Computer Vision Systems*. NY: Academic Press, 1978.

McClelland, J.L. and D.E. Rumelhart. "An interactive activation model of the effect of context in perception: Part 1," Report 8002, Center for Human Information Processing, U. California, San Diego, May 1980.

Minsky, M., "A framework for representing knowledge," in Winston (Ed). *The Psychology of Computer Vision*. McGraw-Hill, 1974.

Minsky, M., "K-Lines: A theory of memory," *Cognitive Science 4*, 2, 117-133, 1980.

Norman, D.A., "A psychologist views human processing: Human errors and other phenomena suggest processing mechanisms," *Proc.*, 7th IJCAI, 1097-1101, Vancouver, B.C., August 1981.

Quillian, M.R., "Semantic memory," in Minsky (Ed). *Semantic Information Processing*. Boston, MA: MIT Press, 1968.

Rieger, C.J., "The Importance of Multiple Choice", *Proceedings TINLAP-2*, Urbana, Illinois, 1978.

Rosenfeld, A., R.A. Hummel and S.W. Zucker "Scene labelling by relaxation operations," *IEEE Trans. SMC 6*, 1976.

Samuel, A.G., J.P.H. van Santen, and J.C. Johnston, "Length effects in word perception: We is better than I but worse than you or them," *J Experimental Psychology: Human Perception and Performance 8*, 1, 91-105, 1982.

Schank, R.C., "Conceptual dependency: A theory of natural language understanding," *Cognitive Psychology 3*, 4, 1972.

Schank, R.C., N. Goldman, C. Rieger, and C. Riesbeck, "MARGIE: Memory, analysis, response, generation, and inference on English," *Proc.*, 3rd IJCAI, 1973.

Schank, R.C., and R.P. Abelson, "Scripts, plans, and knowledge," *Proc.*, 4th IJCAI, 1975.

Schank, Roger C., "Reminding and memory organization: An introduction to MOPs," Research Report 170, Dept. of Computer Science, Yale U., 1979.

Seidenberg, M.S., M.K. Tanenhaus, and J.M. Leiman, "The Time Course of Lexical Ambiguity Resolution in Context", Technical Report #164, Center for the Study of Reading, University of Illinois, 1980.

Small, S.L., "Word Expert Parsing: A Theory of Distributed Word-Based Natural Language Understanding," Ph.D. dissertation and TR 954, Department of Computer Science, U. Maryland, 1980.

Swinney, D.A., "Lexical access during sentence comprehension: (Re)consideration of context effects," *J Verbal Learning and Verbal Behavior 18*, 645-659, 1979.

Tanenhaus, M.K. and J.M. Leiman, "Evidence for multiple stages in the procesing of ambiguous words in syntactic contexts," *J Verbal Learning and Verbal Behavior 18*, 427-440, 1979.

Wilks, Y., "Preference semantics," AI Memo 206, AI Lab, Stanford U, 1973.

Wilks, Y., "Some thoughts on procedural semantics," Cognitive Studies Centre Report 1, U. Essex, 1980.

The Context Model: Language Understanding in Context*

Yigal Arens

Division of Computer Science
Department of EECS
University of California at Berkeley
Berkeley, CA 94720

1. Introduction

This paper describes the language understanding component of the Unix Consultant (UC) system being developed at the Berkeley Artificial Intelligence Research project. The purpose of UC is to hold a conversation with a naive user of the Unix operating system while he or she is working on the computer, answering questions and solving problems for the user. The system has several other components, including the common sense planner PANDORA (Faletti, 1982), and the plan understander PAMELA (Norvig, 1982).

Our natural language understanding system contains as a subpart the PHRAN phrasal analysis program (Wilensky and Arens, 1980a) (Wilensky and Arens, 1980b) (Arens, 1981). PHRAN's knowledge base consists of **Pattern-Concept Pairs** -- pairings of language structures with a conceptual representation of their meaning. It operates by matching the pattern parts of the pairs against the input and using the corresponding concept to describe its meaning.

The current system attempts to deal with the fact that PHRAN by itself unable to deal with reference, and cannot disambiguate unless the linguistic patterns used require a particular semantic interpretation of the words. In addition, we wish to account for the fact that the same utterance may be interpreted differently in different contexts. These inabilities on the part of PHRAN originate in the fact that PHRAN's knowledge is almost entirely of the *language*, as opposed to knowledge about the entire conversation, more general world knowledge, etc. Of course, in order to specify the patterns, PHRAN needs at least some information about the semantics of the words appearing in the sentences it analyzes, but this is limited to the semantic categories the objects described by the language belong to (e.g. Person, Vehicle) and a Conceptual Dependency representation (Schank, 1975) of the actions. In order to hold a meaningful and useful conversation, however, it is clear that such a system must go beyond the (almost) purely linguistic analysis of the sentence to include the effect and the interaction this analysis has on our model of the conversation and on our knowledge as a whole.

The system we are currently constructing has a single mechanism which addresses many of these problems, which we call the **Context Model**. The Context Model contains a record of knowledge relevant to the interpretation of the discourse, with associated levels of activation. There are rules governing how elements introduced into the Context Model are to influence it and the system's behavior.

PHRAN and the Context Model interact continually. PHRAN passes its limited interpretation of the input to the Context Model, and it in turn determines the focus of the conversation and uses it to resolve the meaning of ambiguous terms, of references, etc., and passes these back to PHRAN.

Although it too involves the use of spreading activation and associations among semantic structures for the purpose of understanding text, the Context Model differs substantially in scope from Quillian's work in TLC as described in (Quillian, 1969). TLC was concerned mainly with the determination of the conceptual representation of the input sentence, a task which is handled here mostly by the phrasal analyzer. The Context Model groups related entries in it and arrives at a notion of the situation being discussed. Alternative situations in which a concept may appear can be ignored, thus enabling the system to have a more directed spreading of activation.

(Grosz, 1980) develops in great detail a scheme for determining focus of a task oriented dialog and using it to resolve references. Grosz's system relies heavily on the inherent temporal structuring of the task -- whereas we are trying to develop a more general approach, independent of the type of subject matter discussed. Our system must have the ability to shift focus freely according to the user's input, including the ability to store and recall previous contexts into focus.

The resulting system is able to converse and answer questions, while allowing the user to move in a relatively free manner from one topic to another, as the next example illustrates.

1.1. Example

The exchange described below takes place with the UNIX Consultant (UC) system being constructed at Berkeley. The purpose of the system is to answer the questions of naive users of the UNIX operating system while they are using the computer. See (Wilensky, 1982).

[1] User: How do I print the file fetch.l on the line printer?
[2] UC: To print the file fetch.l on the line printer type 'lpr fetch.l'.

.
. (intervening commands and questions)
.

[3] User: Has the file fetch.l been printed yet?
[4] UC: The file fetch.l is in the line printer queue.
[5] User: How can I cancel it?
[6] UC: To remove the file fetch.l from the line printer queue you must type 'lprm arens'.

In this example the user first asks a question [1] and receives a reply from the system. Then come several other questions and answers, and then the second part of the example. The user asks another relatively straightforward question and then a more problematic one. In order to reply to the last question the system must find the referent of 'it'. The language used implies that this must be a command, but the command in question was issued long ago. The system is able to determine the meaning of [5] only because the context of [1] and [2] had been stored and so could be recalled upon the seeing of [3]. This example will be discussed in more detail in section 3.

2. The Context Model and Its Manipulation

The Context Model is in a constant state of flux. Entries representing the state of the conversation and the system's related knowledge and 'intentions' are continually being added, deleted, or are having their activation levels modified. As a result the same utterance may be interpreted in a different manner at different times. Following are short descriptions of the different elements of the system.

2.1. Entries

The Context Model consists of a collection of entries with associated levels of activation. These entries represent the system's interpretation of the ongoing conversation and its knowledge of related information. The activation level is an indication of the prominence of the information in the current conversational context, so that when interested in an entry of a certain type the

*This research was sponsored in part by the Office of Naval Research under contract N00014-80-C-0732 and the National Science Foundation under grant MCS79-06543.

system will prefer a more highly activated one among all those that are appropriate.

There are various types of entries, and these are grouped into three general categories:

1) **Assertions** – statements of facts known to the system.
2) **Objects** – objects or events which the system has encountered and that may be referred to in the future.
3) **Intentions** –
   a) Entries representing information the system intends to transmit to the user (i.e. output) or other components of an understanding system (e.g. goal tracker, planner).
   b) Entries representing information the system intends to determine from its knowledge base.

2.2. Clusters

The entries in the Context Model are grouped into **clusters** representing situations, or associated pieces of knowledge. If any one member of a cluster is reenforced it will cause the rest of the members of the cluster to be reenforced too. In this manner inputs concerning a certain situation will continue reenforcing the same cluster of entries -- those corresponding to that particular situation. Thus the system arrives at a notion of the topic of the conversation which it uses to help it choose the appropriate interpretation of further inputs.

2.3. Reenforcement

When the parse of a new input is received from PHRAN the system inserts an appropriate entry into the Context Model. If there already exists an entry matching the one the system is adding then the activation levels of all entries in its cluster(s) are increased. The level of activation decays over time without reenforcement, and when it falls below a given threshold the item is removed.

2.4. Stored Clusters

Upon inserting a new item in the Context Model the system retrieves from a database of clusters all those that are indexed by the new item. Unification is done during retrieval and the entries in the additional clusters are also inserted into the Model, following the same procedure described here except that they are given a lesser activation. We thus both avoid loops and accommodate the intuition that the more intermediate steps are needed to associate one piece of knowledge with another the less the mention of one will remind the system of the other.

The system begins operation with a given indexed database of clusters, but clusters representing various stages of the conversation are continually added to it. In principle, this should be performed automatically when the system is cued by the conversation as to the shifting of topic, but currently the system user must instruct it do so. Upon receiving such an instruction, then, all but the least activated entries in the Context Model are stored as a cluster indexed by the most highly activated among them. This enables the system to 'recall' a situation later when presented with a related input.

2.5. Operations on Entries in the Context Model

After a new entry is made in the Context Model the process described above takes place and eventually the activation levels stabilize, with some of the items being deleted, perhaps. Then the system looks over each of the remaining entries and, if it is activated highly enough, performs the operation appropriate for its type. The allowed operations consist of the following:

1) Deleting an entry.
2) Adding another entry.
3) Transmitting a message to another component of the system (i.e. output to the user or data to another program, e.g. PANDORA (Faletti, 1982), for more processing)
4) As part of the UC system, getting information from the UNIX system directly (and inserting an entry corresponding to the result).

3. Details of the Example

In [1] the user asks a simple question. PHRAN analyzes the question and sends the Context Model a stream of entries to be inserted. Among them are the fact that 'fetch.l' is the name of a file, and that the user asked what is the plan for printing it on the line printer. The system records these facts in the Context Model. Indexed under the entry representing the user's desire to obtain a goal there is a cluster containing entries representing the system's intent to find a plan for the goal the user has and instructing the system to tell the user of this plan. This cluster is instantiated here with the goal being the particular goal expressed in the question. The entry expressing the system's need for a plan for the user's goal leads to the plan in question being introduced also. This happens because the system happens to already have this association stored. When the system looks over the entries in the Context Model and comes to the one concerning the need to find the plan in question it will check to see if an entry for such a plan already exists, and in our case it does. But if no plan were found, the system would insert a new entry into the Context representing its intent to pass the information about the user's request to the planner PANDORA (Faletti, 1982). PANDORA will in turn return the plan to be inserted in the Context Model.

So the system finds the plan (issuing the command above) and inserts a new entry instructing the system to output it to the user. And eventually that is done – hence [2].

The topic shifts and the previous context is stored (with the operator's aid, as mentioned above), indexed by the most highly activated entries, including the file name, the mention of the line printer, the event of printing the file, and the command issued.

In [3] and [4] we have an exchange similar to the previous one except that the system actually has to consult the operating system in order to find the answer to the question. There is one major addition however -- as a result of the existence of the new cluster described above, the system has all this extra information triggered and loaded into the Context Model. And this is what makes it possible for the system to determine the referent of 'it' in [5]. Several other commands were mentioned and executed more recently, but in the new cluster just loaded many entries match already existing ones causing all – including the command intended for cancellation – to be more highly activated.

4. Shortcomings

The system is not currently able to determine on its own that the topic has changed and that it must store the current context. In addition to linguistic cues, we should be able to use the Context Model too in order to help in such a determination, but this work has not been done yet.

When it is instructed to, the current system stores essentially a copy of the more highly activated elements of the Context Model when creating a new cluster. They are not assumed to have any particular structure or relations among them other than all being highly activated at the same time. This causes two problems:

1) As a result it is very difficult to generalize over such clusters (cf. Lebowitz, 1980). The system may at some point determine a plan for changing the ownership of a particular file, and store a cluster containing it. If it is faced with the need to change the ownership of another file, however, the system will not be able to use this information. In the example above this problem was not encountered because the clusters used were preprogrammed to include variables in place of particular files.
2) There is no way to compare two clusters and determine that in fact they are similar. Thus we may have many clusters indexed by a certain entry all of which actually describe essentially the same situation.

Another element missing from the system is a model of the user. Certain assumptions are made as to the knowledge the user has of the Unix operating system, but these are built in and cannot be modified according to past interactions. Constructing such a

model will probably require work beyond the scope of this project.

5. References

Arens, Y. (1981). Using Language and Context in the Analysis of Text. In *Proceedings of the Seventh International Joint Conference on Artificial Intelligence*, Vancouver, B.C.

Faletti, J. (1982). PANDORA – A Program for Doing Commonsense Planning in Complex Situations. Submitted to *The Second Annual National Conference on Artificial Intelligence*, Pittsburgh.

Grosz, B., J. (1980). Focusing and Description in Natural Language Dialogues. In *Elements of Discourse Understanding: Proc. of a Workshop on Computational Aspects of Linguistic Structure and Discourse Setting*. A. K. Joshi, I. A. Sag, and B. L. Webber, eds. Cambridge University Press.

Lebowitz, M. (1980). Generalization and Memory in an Integrated Understanding System. Tech. Report 186, Yale University Department of Computer Science. Ph.D. Thesis.

Norvig, P. (1982). Integrating Frame-Based and Goal-Based Processing in a Story Understanding Program. Submitted to *The Second Annual National Conference on Artificial Intelligence*, Pittsburgh.

Quillian, M., R. (1969). The Teachable Language Comprehender: A Simulation Program and a Theory of Language. In *Communications of the ACM*, v.12, no.8.

Schank, R. C. (1975). *Conceptual Information Processing*. American Elsevier Publishing Company, Inc., New York.

Wilensky, R. (1982). Talking to UNIX in English: An Overview of UC. Submitted to *The Second Annual National Conference on Artificial Intelligence*, Pittsburgh.

Wilensky, R., and Arens, Y. (1980). PHRAN – a Knowledge-Based Natural Language Understander. In *Proceedings of the 18th Annual Meeting of the Association for Computational Linguistics*, Philadelphia.

Wilensky, R., and Arens, Y. (1980). PHRAN – a Knowledge Based Approach to Natural Language Analysis. University of California at Berkeley, Electronic Research Laboratory Memorandum No. UCB/ERL M80/34.

Wilensky, R., and Morgan, M. (1981). One Analyzer for Three Languages. University of California at Berkeley, Electronic Research Laboratory Memorandum No. UCB/ERL M81/67.

Judgmental Inference:
A Theory of
Inferential Decision-Making During Understanding

Richard H. Granger

Artificial Intelligence Project
Computer Science Department
University of California
Irvine, California 92717

ABSTRACT

In the course of understanding a text, a succession of decision points arise at which readers are faced with the task of choosing among alternative possible interpretations of what they're reading. Careful analysis of a wide range of sample texts reveals that such decisions are often based on complex evaluations of the interpretation being constructed, and sometimes cause the reader to construct and discard a number of intermediate inferences before settling on a final interpretation for a text.

This paper describes Judgmental Inference theory as a proposed scheme of evaluation metrics and mechanisms, derived from examination of inference decisions arising during text understanding. A series of programs, ARTHUR, MACARTHUR and JUDGE are briefly described, which incorporate some of the metrics and mechanisms of Judgmental Inference, enabling them to understand texts more complex than those that can be handled by other understanding systems.

1.0 Introduction

Many national newspapers carried front-page versions of the following story early this year:

[1] A Nicaraguan soldier, who last year made a public statement alleging Cuban, Ethiopian and Nicaraguan military aid to Salvadorian leftist guerrillas, today publicly retracted his story at a State Department news conference.

Why did the Nicaraguan soldier make the statements he made, a year ago and now? Why did the State Department hold these two news conferences? It is possible that the State Department had some reason for holding the news conference, intending the Nicaraguan soldier to recant; but most readers assume that the State Department had different intentions that were not fulfilled, for reasons out of their control. Indeed, most readers don't even consciously think of the former interpretation, even though it is a logical possible alternative explanation of the events.

Our analysis of examples like this has led to the identification of decision points at which human understanders are faced with the task of choosing particular inferential paths from among an array of possible alternatives. These inference decisions are based on complex evaluation metrics for judging the appropriateness of a particular inference, and on mechanisms for constructing and revising interpretations during understanding. Judgmental Inference theory (Granger [1982]) consists of a set of evaluation metrics and mechanisms derived from examination of inference decisions arising during text understanding. This paper describes how some of these judgmental metrics and mechanisms are applied during understanding.

We view this work as compatible with and complementary to research that focuses primarily on representational issues in text understanding, such as Schank and Abelson [1977], Wilensky [1980], Charniak [1980]. By examining the occurrences of inference decisions during understanding, we intend to provide a look at the mechanisms by which such representations are chosen, constructed, judged, confirmed and/or discarded during the processing of a text.

2.0 Illustration of understanders' decisions

2.1 Evaluating and supplanting inferences

Consider the following example:

[2] Kathy and Chris were playing golf. Kathy hit a shot into the rough. She wanted to let her good friend Chris win the game.

Most readers assume that the reason Kathy hit her shot into the rough was to increase her opponent's chances of winning, out of friendship. However, consider the following:

[3] Ken and Carl were playing golf. Ken hit a shot into the rough.

From reading just this two-sentence version, people infer that Ken and Carl both were playing to win, and that Ken's bad shot therefore was accidental, and will hinder his goal of winning the game. However, after readers have read the third sentence that appears in version [2], they appear to have changed this initial interpretation a great deal. It is not just that Kathy doesn't want to win the game, but also that she probably made her bad shot on purpose, not accidentally. Virtually all readers arrive at this interpretation by the end of this example, by supplanting some of their initial inferences with new ones (see Granger [1980]).

2.2 Evaluation metrics of cohesion and parsimony

Why do people arrive at this different interpretation about Kathy's action in this example? The answer is far from obvious. In particular, there is no question of logical consistency here; the interpretation that Kathy hoped to lose the game but that her bad shot was nonetheless accidental is just as logically consistent as the one that people actually infer, namely that her bad shot was intentional, not accidental.

It turns out that the scope of this phenomenon is very wide: people often arrive at interpretations that appear to involve the supplanting of initial inferences, even when that extra work is not necessary on grounds of logical consistency.

This research was supported in part by the Naval Ocean Systems Center under contract N00123-81-C-1078.

(A large number of additional text examples of this phenomenon are given in Granger [1980] and [1982].)

The decision to reject an initial inference, then, must depend on an evaluation of the representation based on some metric other than logical consistency. One such evaluation metric that was (implicitly) incorporated into previous theories of inference generation (e.g. Rumelhart [1981], Crothers [1978], Bower, Black and Turner [1979], Schank [1973]) we have termed the "cohesion metric". The cohesion metric requires that every statement in a text be connected to at least one other, resulting in all the pieces of the text representation being tied together via either referential, causal or intentional connective inferences.

Cohesion by itself is not sufficient to evaluate the goodness of a text representation, however. Another evaluation metric, identified in our previous work (Granger [1980]), measures the parsimony of a representation, with respect to the goals that motivate the events in the text. For instance, consider the following example:

[4] Doug went to a gas station. He robbed it and got away with $50.

(a) Doug went to the gas station intending to get gas, and then he changed his mind and decided to rob the station instead;

(b) Doug went to the gas station intending to rob it.

Just as in the "golf" example [3], this example can be interpreted in two different ways, both of which are not only logically consistent, but also referentially and causally cohesive, since Doug had to get to the gas station before he could rob it, regardless of his intentions in performing those actions.

Therefore, the cohesion metric does not differentiate between these two alternative interpretations, but people do: they universally seem to generate interpretation (b), which consists of a single goal (getting money from the gas station), and in fact they rarely even consciously notice the possibility of (a), which consists of two separate goals each explaining one of Doug's actions. The evaluation metric of parsimony essentially tests that an interpretation be maximally parsimonious with respect to the number of goals used to explain the events in the story; i.e., the fewer separate motives inferred to account for the story events, the better.

(Note: an evaluation of an unparsimonious interpretation will not always result in the decision to supplant inferences; sometimes readers leave "loose ends" in their interpretation, to be resolved later. See Granger [1982] for a discussion of loose ends.)

2.3 Shaping interpretations of behavior

We have identified some further evaluations that understanders perform, beyond cohesion and parsimony, which arise when a reader is led to "doubt" any part of his interpretation of a text. Such doubts can be instilled either by information presented in the text, or by "extra-textual" factors (see Granger [1981]) which may steer the reader away from an otherwise plausible interpretation. Examples of such "doubt-factors" include the reader's knowledge of the reliability of the text source (e.g., the difference between the New York Times and the National Enquirer); knowledge of an actor's deviousness (e.g., a car salesman vs. a priest); relative boredom or interest, i.e., the reader's desire to pursue possible alternative interpretations vs. just settling on a default interpretation that's "good enough". An easy way to induce a doubt factor in a reader is to simply tell him that his initial interpretation is incorrect; i.e., explicitly ask for a new and different interpretation of a text.

It turns out that readers are very capable of producing a series of such alternative interpretations of texts when they're continually told their initial interpretation is incorrect. For instance, following is a story adapted from a newspaper text, along with a series of interpretations informally elicited from a subject:

[5] The Pakistani ambassador to the United States made an unscheduled stop in Albania on his way home to what an aide of the ambassador described as "a working vacation".

Q1) Why did the ambassador go to Albania?
A1) It looks like he was on vacation — he went to Albania first and then to home, I guess in Pakistan.

Q2) No, that's not the real reason. Why did he go to Albania?
A2) Well, maybe there was some emergency reason ... it said it was unscheduled, so maybe it was that something went wrong and they had to stop there, and then they went on.

Q3) Still not it, but try again; why did he go to Albania?
A3) Ok, maybe, well he's an ambassador, so he could have been supposed to go to Albania ... so it could have been a meeting, like "shuttle diplomacy" ... but it was supposed to be a secret, so that's why they said it was unscheduled.

These different interpretations of [5] are each based on different interpretations of the actor's reasons for doing what he did. It is natural that different behavior interpretations should give rise to different text interpretations; most current theories of text representation focus primarily on representation of the events described in the text, rather than on a more "syntactic" analysis of the structure of the text itself.

Our analysis of this and similar examples has revealed a large class of inference evaluations people perform based on their attempts to decide what kind of behavior an actor has performed, for instance:

1. "simple" goal pursuit, e.g., "John was hungry, so he ate a hamburger";

2. "complex" goal pursuit, (i.e., goal interactions; see Wilensky [1979]) — e.g., "John wanted to see the football game but he also had a paper due the next day" (goal conflict);

3. deceptive or intentionally misleading behavior, e.g., "Clark wanted Lois to think he was drunk, so he smiled and fell off the barstool onto the ground";

4. accidental (non-goal-directed) behavior, e.g., "Jack smiled and fell off the barstool onto the ground"(!);

5. impromptu reactions to unplanned-for contingencies, e.g., "Bill threw himself under the jeep when he saw the man pull a gun".

Our classification scheme for dividing up the gamut of possible interpretations of behavior (e.g., intentional vs. unintentional at the top level, subdividing intentional behaviors into simple, deceptive, pre-planned, impromptu, etc., and unintentional behavior into various types of failures such as skill failure, information failure, etc.) is described in detail in Granger [1982]. We call each of these subdivisions an interpretation-"shape", since categorizing an actor's behavior into one of these classes will result in a particular shape of the representation graph constructed, and because re-interpreting an actor's behavior results in re-shaping the representation.

We have implemented two computer programs, ARTHUR and MACARTHUR, which incorporate the evaluation metrics of cohesion, parsimony, and shapes to produce interpretations of texts that cannot be handled by other text-understanding systems. Granger [1982] gives sample output of the operation of these programs on some of the text examples discussed above.

3.0 Additional categories of inference decisions

3.1 "Suspicious" understanding

It is often impossible for an understander to identify the "correct" interpretation shape for an actor's behavior. For instance, consider the following version of a story that was on the front page of a number of national newspapers earlier this year:

[6] A report by the New York State Racing and Wagering Board released today states unequivocally that leading jockeys conspired to "fix" at least 13 races in the mid-1970's, and that the jockeys have been "patently unbelievable" in denying their involvement in the scheme.

Understanding [6] requires the recognition that the observed behavior of jockeys can be very difficult to classify as either "accidental" or "deceptive". Hence, a jockey (or a jai-alai player, boxer, etc.) may lose a competition without an observer's being able to tell whether he did it intentionally or accidentally.

These are special cases of the general problem of detecting deceptive behavior by using knowledge of "cover stories". Some recent work in AI (e.g., Bruce and Newman [1978]) has pointed out that a method of maintaining separate "belief spaces" for different actors is crucial for understanding deception. However, understanding deception can also require a great deal more than this; in particular, a more subtle deceiver will typically try to cause observers to infer for themselves some false interpretation of his actions, thereby covering up the real reasons. Political propaganda, advertisements for products, and face-saving "white lies" are all examples of this kind of deception. The ability to understand (and generate) complex deceptive behavior such as this depends not only on separate belief spaces, but also on the ability to construct plausible alternative explanations for events. The more plausible the alternative explanation, the more likely the deception is to succeed in misleading understanders.

A "suspicious" understander is one who can (at least) construct alternative interpretations of events, and then can attempt to decide among them, typically by gathering additional information. Such information-gathering is based on finding a possible motive, i.e., finding a plausible explanation that the "obvious" explanation is intended to cover. The JUDGE program, currently under construction, is being designed to make use of knowledge of the shapes of alternative interpretations to detect plausible cover stories in the domain of criminal investigation. For more descriptions of cover stories and JUDGE, see Granger [1982], and Granger and Eiselt [1982].

3.2 Understanding accidents

We have also investigated the types of accidental behavior that can be described in texts, and the relations between accidental and goal-directed behavior. For example recall Ken, who accidentally hit his golf shot into the rough. Although his action of of striking the ball was intentional, the causal outcome of the ball ending up in the rough was unintended. We have classified Ken's problem as a "skill failure", i.e., an intentionally-performed physical action which results in a non-intended outcome as a result of some physical lack. There are a number of other types of intention-accident pairs like this, such as "information failure", "too-shallow planning", etc. For a further discussion of accidents and how to understand them, see Granger [1982], and Meehan [1981].

4.0 Conclusions and future research directions

4.1 What we're proposing

We have observed that people's understanding behavior is marked by an ongoing process of making inference decisions. Among the decisions understanders implicitly make are:

1. Is the interpretation referentially and causally cohesive?
2. Is the interpretation parsimonious with respect to the actors' intentions?
3. Is there reason to doubt or be suspicious of the shape of the initial interpretation?
4. Is there reason enough to revise the interpretation (supplant, re-shape, etc) or should it be left with "loose ends"?

The evaluation metrics and the construction and revision processes of Judgmental Inference theory are derived directly from our observations and analyses of some of the classes of inference decisions that readers are faced with during the task of text understanding.

We view these theories as compatible with and complementary to theories of text representations, since we intend to describe the mechanisms by which such representations are chosen, constructed, judged, confirmed and/or discarded in the process of understanding. Our theories have so far been incorporated into two working computer programs, ARTHUR and MACARTHUR, and are currently being used as the design impetus for a new computer system called JUDGE, and for a series of psychological and neurophysiological experiments, briefly described below, to test the correspondence of our theories to people's actual understanding behavior.

4.2 Minds, brains and processes

A number of researchers in the neurosciences (e.g., Arbib [1979], Geschwind [1980]) have pointed out that brain research might help guide parts of cognitive science and AI research, and vice versa. One particular issue that has been pointed out frequently is that "there is no evidence for the existence of any all-purpose computer [in the brain]. Instead, there seems to be a multiplicity of systems for highly special tasks." (Geschwind [1980], p.191). Our research on inference decisions has indeed led us away from viewing human understanding behavior as arising from a "general purpose computer"; we have ended up instead deriving a number of special-purpose mechanisms, e.g., inference pursuit, evaluation, supplanting, re-shaping, which comprise our "judgmental inference" model of understanding.

We are currently designing a number of psychological and neurological experiments on inference decisions, based on the predictions of our model (see Granger [1982]); as well as attempting to re-interpret some existing results (e.g., Rumelhart [1981], Crothers [1978], Hillyard and Kutas [1980], Black [1981]), in light of the model.

For instance, we are investigating the issue of when people evaluate their interpretations consciously vs unconsciously; our model currently fails to account for such individual differences. We hope to use the data from such experiments to find problems with our theories, and to refine the model, thereby working eventually towards some small amount of "neurological validity" in our process models of cognition.

5.0 References

Arbib, M.A. and Caplan, D. Neurolinguistics must be computational. Behavioral and Brain Sciences, 2:449-483, 1979.

Black, John. The effects of reading purpose on memory for text. Cognitive Science Technical Report 7, Yale University, 1980.

Bower, Gordon H., John B. Black, and T. J. Turner. Scripts in text comprehension and memory. Cognitive Psychology 11:177-220, 1979.

Bruce, Bertram and Denis Newman. Interacting plans. Cognitive Science 2:195-233, 1978.

Charniak, Eugene. On the use of framed knowledge in language comprehension. Yale Computer Science Research Report 137, 1978.

Crothers, Edward J. Inference and Coherence. Discourse Processes, 1:51-71, 1978.

DeJong, G.F. Skimming Stories in Real Time: An Experiment in Integrated Understanding. Yale Computer Science Research Report 158, 1979.

Doyle, J. A truth maintenance system. Artificial Intelligence 12(3), 1979.

Geschwind, N. Neurological Knowledge and Complex Behaviors. Cognitive Science, 4:185-193, 1980.

Granger, R.H. When expectation fails: Towards a self-correcting inference system. In Proceedings of the First National Conference on Artificial Intelligence, Stanford University, 1980.

Granger, R.H. Directing and re-directing inference pursuit: Extra-textual influences on text interpretation. In Proceedings of the Seventh International Joint Conference on Artificial Intelligence (IJCAI), Vancouver, British Columbia, 1981.

Granger, R.H. Judgmental Inference: Inferential Decision-Making During Understanding. Computer Science Technical Report 182, University of California, Irvine, 1982.

Granger, R.H. and K. Eiselt. 'Suspicious' Understanding: Detecting Possible Deception by Inferring Alternative Explanations. Computer Science Technical Report 185, University of California, Irvine, 1982.

Kutas, M. and S.A. Hillyard. Reading Senseless Sentences: Brain Potentials Reflect Semantic Incongruity. Science, 207:203-5, 11 Jan 1980.

Lebowitz, M. Generalization and Memory in an Integrated Understanding System. Computer Scinece Research Report 186, Yale University, 1981.

McDermott, D.V. and Jon Doyle. Non-monotonic logic I. Artificial Intelligence 13(1,2):41-72, 1980.

Meehan, J.R. Boy meets goal, boy loses goal, boy gets goal: the nature of feedback between goal-based simulation and understanding systems. Computer Science Technical Report 170, University of California, Irvine, 1981.

Rumelhart, D.E. Understanding understanding. Technical Report 100, Center for Human Information Processing, University of California, San Diego, January 1981.

Schank, R.C. Causality and Reasoning. Technical Report 1, Istituto per gli studi Semantici e Cognitivi, Castagnola, Switzerland, 1973.

Schank, R.C. and Robert P. Abelson. Scripts, Plans, Goals, and Understanding. Lawrence Erlbaum, Hillsdale, New Jersey, 1977.

Schulenburg, D. Interpreting Intentional and Unintentional Behavior. Technical Report, University of California, Irvine, 1982.

Wilensky, R. Understanding Goal-Based Stories. Research Report 140, Yale Computer Science Department, 1978. Garland Press, New York, 1980.

STRUCTURE-MAPPING: A THEORETICAL FRAMEWORK FOR ANALOGY AND SIMILARITY

Dedre Gentner

Bolt Beranek and Newman Inc.
50 Moulton Street
Cambridge, Massachusetts 02238

This paper describes a theoretical framework in which analogies and other comparisons are defined in terms of structure-mappings between domains (Gentner, 1979, 1980). Different kinds of mappings correspond to analogies, metaphors, literal similarity statements, applications of general laws, and simple chronologies. The chief focus is on explanatory analogies, such as are used in scientific modelling (Gentner, 1981, 1982; Gentner & Gentner, 1982). Such analogies are fundamentally assertions that partly identical relational structures apply to dissimilar objects across different domains.

It is generally accepted that the degree of literal similarity perceived between two objects depends on the degree of overlap among their components. In Tversky's (1977) elegant contrast model, the similarity between A and B is greater the greater the size of the intersection ($A \cap B$) and the less the size of the two complement sets ($A - B$) and ($B - A$).[1] This account works well for literal similarity, but the mere relative number of shared and non-shared predicates appears to be an inadequate basis for a general account of relatedness.

For example, consider a simple arithmetic analogy. The analogy 3:6::2:4 is no better than the analogy 3:6::200:400, even though 3 has more features in common with 2 than with 200. It is not the overall number of shared versus nonshared features that counts here, but only the relationship "twice as great as." I will argue that a general theory of relatedness between domains must be based on the relational structure of the overlapping information. The structure of the shared versus nonshared predicates determines whether a given comparison is thought of as analogy, as literal similarity, or as the application of a general law.

In this paper I first lay out some representational preliminaries; second, provide definitions and examples of each kind of relatedness; and finally, discuss some psychological implications of the framework. To give a brief preview: If both the relationships and the object descriptions correspond, the comparison is one of literal similarity; if the relationships correspond, but the objects do not, the comparison is analogical. The third possibility, that the objects correspond but the relationships do not, represents neither literal nor analogical similarity. Such comparisons arise chiefly in chronologies, in which the same entities pass from one configuration into another over time. The place of general laws in this framework will also be discussed.

Preliminary Assumptions

1. Domains and situations are psychologically viewed as systems of objects, object-attributes and relations between objects. These "objects" may be coherent conceptual bundles or component parts of a larger object, rather than separate concrete objects; the important point is that they function as wholes at a given level of organization.

2. Domains and situations are represented propositionally. The format used here is a propositional network of nodes and predicates (cf. Miller & Johnson-Laird, 1979; Rumelhart & Norman, 1975; Rumelhart & Ortony, 1977; Schank & Abelson, 1977). The nodes represent concepts treated as wholes and the predicates express propositions about the nodes.

3. The distinction between object attributes and relationships is important. In a propositional representation, the distinction can be made explicit in the predicate structure: attributes are predicates taking one argument, and relations are predicates taking two or more arguments. For example, COLLIDE (x,y) is a relation, while RED (x) is an attribute.

4. The distinction between first-order predicates (taking objects as arguments) and second- and higher-order predicates (taking propositions as arguments) is important. For example, if COLLIDE (x,y) and FALL (y) are first-order predicates, CAUSE [COLLIDE(x,y), FALL(y)] is a second-order predicate.

5. These representations, including the distinctions between different kinds of predicates, are intended to reflect the way people construe a situation, rather than what is logically possible.[2]

---

[1] The negative effects of the two complement sets are not equal: if we are asked "How similar is A to B?", the set (B - A)--features of B not shared by A--counts much more than the set (A - B).

6. Finally, it is assumed that a comparison "An X is (like a) Y." conveys that knowledge is to be mapped from Y to X. X will be called the target, since it is the domain being explicated. Y will be called the base, since it is the (presumably more familiar) domain that serves as the source of knowledge.

Structure-mapping: Interpretation Rules

Assume that the hearer's representation of the base domain B can be stated in terms of object nodes b_1, $b_2,\ldots,b_n$ and predicates such as A, R, R'. The hearer knows, or is told, that the target domain has object nodes t_1, $t_2,\ldots,t_m$. A structure-mapping comparison maps the nodes of B onto the nodes of T:

---

[2] Logically, a relation R(a,b,c,) can perfectly well be represented as Q(x), where Q(x) is true just in case R(a,b,c) is true. Psychologically, the representation must be chosen to model the way people think.

$$M: b_i \dashrightarrow t_i$$

The hearer derives inferences about T by applying predicates valid in the base domain B, using the node substitutions dictated by the mapping:

$$M: [R(b_i, b_j)] \dashrightarrow [R(t_i, t_j)]$$

Here $R(b_i, b_j)$ is a relation that holds in the base domain B. Attributes (one-place predicates) from B can also be mapped into T:

$$[A(b_i)] \dashrightarrow [A(t_i)].$$

Finally, higher-order relations, such as $R'(R_1, R_2)$, can also be mapped:

$$M: [R'(R_1(b_i, b_j), R_2(b_k, b_l)] \dashrightarrow [R'(R_1(t_i, t_j), R_2(t_k, t_l)]$$

Kinds of Structure-Mappings

(1) A literal similarity statement is a comparison in which a large number of predicates is mapped from base to target, relative to the number of nonmapped predicates (Tversky, 1979). The mapped predicates include both object-attributes and relational predicates.

EXAMPLE(1): The X12 star system in the Andromeda nebula is like our solar system.

INTERPRETATION: Intended inferences include both object characteristics--e.g., "The X12 star is YELLOW, MEDIUM-SIZED, etc., like our sun." and relational characteristics, such as "The X12 planets REVOLVE AROUND the X12 star, as in our system." Figure 1 shows a representation of our solar system; most or all of the predicates shown would be mapped in a literal similarity comparison.

(2) An analogy is a comparison in which relational predicates, but not many object attributes, can be mapped from base to target.

EXAMPLE(2): The hydrogen atom is like our solar system.

INTERPRETATION: Intended inferences concern chiefly the relational structure: e.g., "The electron REVOLVES AROUND the nucleus, just as the planets REVOLVE AROUND the sun." but not "The nucleus is YELLOW, MASSIVE, etc., like the sun." (see Figure 1). If higher-order relations are present on the base they can be mapped as well: e.g., The hearer might map "The fact that the nucleus ATTRACTS the electron CAUSES the electron to REVOLVE around the nucleus." from "The fact that the sun ATTRACTS the planets CAUSES the planets to REVOLVE AROUND the sun." (This relation is not shown in Figure 1.)

(3) A general law is a comparison in which the base domain is a named abstract relational structure. Such a structure would resemble Figure 1, except that the object nodes would be generalized physical entities, rather than particular objects like "sun" and "planet". Predicates from the abstract base domain are mapped into the target domain; there are no nonmapped predicates.

EXAMPLE(3): The hydrogen atom is an example of a central force system.

INTERPRETATION: Intended inferences include "The nucleus ATTRACTS the electron."; "The electron REVOLVES AROUND the nucleus." These are mapped from base propositions such as "The central object ATTRACTS the peripheral object."; or "The less massive object REVOLVES AROUND the more massive object."

(4) A chronology is a comparison between two time-states of the same domain. The objects at time 1 map onto the objects at time 2. This is the only interesting case in which objects are shared but relational structure need not be. The two time-states share object-attributes, but in general not relational predicates.

EXAMPLE(4): Two hydrogen atoms and an oxygen atom will combine to form water.

INTERPRETATION: The intended inferences that can be mapped from time state 1 to time state 2 concern enduring characteristics of the component objects: "Oxygen HAS ATOMIC WEIGHT 16."

Neither configurational relations nor dynamic relations of the initial system can be mapped into the final system. Note that overlap among component objects is not sufficient to produce similarity between systems: Two isolated hydrogen atoms and an oxygen atom do not resemble water, either literally or analogically.

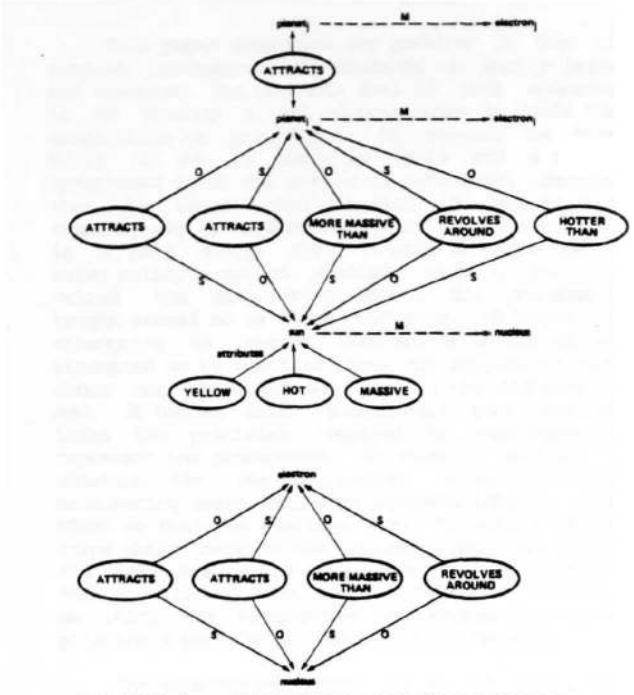


Figure 1. Structure-mapping between solar system and hydrogen atom.

To summarize, overlap in relations is necessary for any strong perception of similarity between two domains. Overlap in both object attributes and inter-object relationships is seen as literal similarity, and overlap in relationships but not objects is seen as analogical relatedness. Overlap in objects but not relationships may be seen as temporal relatedness, but not as similarity.

According to this analysis, the contrast between analogy and literal similarity is a continuum, not a dichotomy. Given that two domains overlap in relationships, they are more literally similar to the extent that their object-attributes also overlap. A different sort of continuum applies between analogies and general laws: In both cases, a relational structure is mapped from base to target. If the base representation includes concrete objects that must be left behind, the comparison is an analogy. As the object nodes of the base domain become more abstract and variable-like the comparison is seen as a general law.

Psychological speculation: The Analogical Shift Conjecture. People learning a new domain often make spontaneous comparisons with other domains. The speculation is that the earliest comparisons are chiefly literal-similarity matches, followed by analogies, followed by general laws. For example, Ken Forbus and I have observed a subject trying to understand the behavior of water flowing through a constricted pipe. His first comparisons were similarity matches, e.g., water coming through a constricted hose. Later, he produced analogies such as a train speeding up or slowing down, and iron balls banging into one another and transferring momentum. Finally, he was able to state a version of the Bernoulli principle, that velocity increases and pressure decreases in a constriction.

Literal similarity matches are highly accessible but not very useful in deriving causal principles, because there is too much overlap. Analogies are harder to generate, since they require searching the data base for relational matches, not object matches. However, once found, an analogy should be more useful in deriving the key principles, especially if the set of overlapping predicates includes higher-order relations such as CAUSE (see Winston, 1981). Finally, by comparing two or more analogies, the common subparts of the relational structure can be isolated and a general law derived. [See Gick and Holyoak (in press) for relevant studies.]

In summary, no treatment of domain relatedness can be complete without distinguishing between object features and relational features: that is, between relational predicates and one-place attributive predicates. Careful analysis of the predicate structures being mapped is central to modelling the inferences people make in different kinds of comparisons.

References

Gentner, D. The structure of analogical models in science (BBN Report No. 4451). Cambridge, Mass.: Bolt Beranek and Newman Inc., 1980.

Gentner, D. Metaphor as structure-mapping. Paper presented at the meeting of the American Psychological Association, Montreal, September 1980.

Gentner, D. Are scientific analogies metaphors? In D. Miall (Ed.), Metaphor: Problems and perspectives. Brighton, England: Harvester Press Ltd., in press, 1982.

Gentner, D., & Gentner D. R. Flowing waters or teeming crowds: Mental models of electricity. In D. Gentner & A. L. Stevens (Eds.), Mental models. Hillsdale, N.J.: Erlbaum, in press, 1982.

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

Gick, M. L., & Holyoak, K. J. Schema induction and analogical transfer. Cognitive Psychology, in press.

Miller, G. A., & Johnson-Laird, P. N. Language and perception. Cambridge, Mass.: Harvard University Press, 1976.

Rumelhart, D. E., & Norman, D. A. The active structural network. In D. A. Norman, D. E. Rumelhart, & the LNR Research Group, Explorations in cognition. San Francisco: W. H. Freeman & Co., 1975.

Rumelhart, D. E., & Ortony, A. Representation of knowledge. In R. C. Anderson, R. J. Spiro, & W. E. Montague (Eds.), Schooling and the acquisition of knowledge. Hillsdale, N.J.: Erlbaum, 1977.

Schank, R., & Abelson, R. Scripts, plans, goals, and understanding. Hillsdale, N.J.: Erlbaum, 1977.

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

Winston, P. Learning new principles from precedents and exercises. MIT Artificial Intelligence Memo No. 632, Massachusetts Institute of Technology, Cambridge, Mass., May 1981.

Principles of Procedures Composition

Christopher K. Riesbeck
Yale University

Edwin L. Hutchins
Navy Personnel Research and Development Center

This paper addresses the problem of how to compose procedures that students can easily learn and remember. The ultimate goal of this endeavor is to develop a set of principles to guide the composition of procedures. At present we have built a set of analytic tools and a set of hypotheses about the nature of procedural learning that can be empirically tested. We came to this topic by way of an examination of the instruction in a navy school that teaches students how to solve relative motion problems with a job aid called the maneuvering board. The procedures taught seemed to us to be confusing. We began by attempting to rewrite them and as we did so, we attempted to be specific about our complaints, and about our attempted solutions to the problems we saw. It became clear immediately that English lacks the precision required to unambiguously represent the procedures. In order to provide a notation for the procedures, we developed the Maneuvering Board Emulation Language (MABEL). With MABEL we could be specific about the nature of the steps which comprise the procedure and also about the relations among the steps in the procedure. This specificity permitted us to propose measures on which the alternative procedures for accomplishing a particular task could be compared.

The maneuvering board is a job aid that represents the motions of ships relative to each other in a way that supports computations that predict the consequences of possible future actions (including no action at all) to be taken by the ships. The maneuvering board itself is a sheet of paper printed with a polar coordinate plot (azimuth grid) and various scales that can be used in plotting ranges and bearings. Problems are solved on the maneuvering board by plotting points, and drawing lines and vectors which represent aspects of ships' motions (DMA 1975).

In this paper we will deal with a portion of only one of the many problems that are solved on the maneuvering board, the Closest Point of Approach (CPA) problem. In this procedure, the relative motion of an observed ship is plotted, and the bearing, range, and time of the closest point of approach between the two ships is determined. If it is determined that the ships will pass closer to each other than is desired, actions will have to be taken to ensure a safe separation. Those actions will be based on other computations performed on the maneuvering board.

Creating a representation language

The main issue in designing a language is finding the right "grain" (Moore and Newell 1974), i.e., the right level of detail. A representation language for the maneuvering board that included the pencil coming in contact with paper fiber and depositing carbon granules would be cumbersome and unenlightening, while one at the same level of abstraction as English fails to capture important distinctions.

The language we have designed was built according to the following constraints:

1) it would not include any appeal to the real world or to the goals to be achieved. Thus, there is no operator for "Find closest point of approach." The operators are all within the world of the maneuvering board itself.

2) it would not include any mention of the actual physical tools involved. Thus, there is no mention of pencils, parallel rules or dividers.

We call this language MABEL, for MAneuvering Board Emulation Language. The objects in MABEL include points, several types of lines (scales, segments, rays, vectors), circles, numbers (speeds, distances, times and angles), and turns (left and right). MABEL has only geometric operators. Although some operators involve fairly complex geometric activity (e.g. INTERSECT(line circle), TRANSLATE(line, point)), not all geometric constructions are included.

Task analysis

Dependency analysis

A dependency analysis constructs a graph representing what steps of a procedure depend on other steps. As a trivial example, we can't find the distance from the reference ship to the closest point of approach (CPA) until we first find the location of the CPA point. Hence we say that the distance determining step depends on the CPA plotting step.

The dependency analysis reveals the constraints on step ordering that are imposed by the nature of the task itself. It defines the set of procedures composed of the given steps that can actually produce the desired result. Among the members of this set, some procedures feel more natural or meaningful than others. One property of procedures that makes them meaningful is the organization of goals and actions.

Goal-action analysis

To make the goal structure of a procedure explicit, we do a goal-action analysis. A goal-action analysis creates a tree whose top node is the goal to be satisfied. Under this node are other nodes representing the goals that have to be achieved in order to satisfy the top goal. Finally, attached to each goal are the actions to be done once the subgoals are achieved. A goal analysis is typically more specific and therefore more constraining than the dependency analysis.

Below is a goal-action tree for finding the bearing of the CPA.

Goal: bearing of CPA (BC)
  Goal: Direction of Relative Movement (DRM)

```
Goal: Line of Movement (LOM)
    Goal: M1
    Action: PLOT(B1, R1, GRID)
    Goal: M2
    Action: PLOT(B2, R2, GRID)
  Action: RAY(M1, M2)
Action: TRANSLATE(LOM, P:GC) => L:DRM
        INTERSECT(L:DRM, P:GE) => P:DRM
        READVALUE(P:DRM) => DRM
Action: ADD(DRM, +/- 90)
```

Measuring Goal-Action Sequences

A goal-action sequence is a linearization of a goal-action forest. The sequence specifies when each goal is initiated (i.e., when work on the goal begins) and when each action is executed (i.e., when the action is performed). To generate a sequence from a forest, we select some goal node in some tree to be the first one in the sequence. After that, we can go to any node in the forest and select its goal or action component, subject only to the following constraints:

-The goal of a node must be initiated before the action of that node can be done.

-Lower actions in one tree must be executed before higher actions.

If the goal-action sequences are to be converted into computer programs, then the order in which things are done really doesn't matter, as long as the constraints given are satisfied. But if the sequences are to become instructions for people to read, follow, learn, and so on, then the constraints fail to take into account the limits of the short-term memory or the organization of long-term memory. Intuitively, we can feel that a sequence of instructions that hopped randomly from one subgoal to another would be very confusing and hard to learn.

In the following paragraphs, we will describe a number of measures for sequences. Each measure is concerned with something that we believe makes sequences easy or hard to learn. For the moment, it is just assumed that these measures are the significant ones. By making each measure explicit, we hope to simplify the problems of actually testing the learnability of instructions.

Number of Top-level Goals (NTG), counts how many goals are initiated in the sequence without any higher-level goal preceding them. We assume that the more top-level goals an instruction text presents, the harder that text is to learn. Hence, NTG should be minimized.

Distance From Goal (DFG), counts for each action how many other actions separate it from its goal. For a sequence, we define the overall DFG to be the maximum of the DFGs for its actions. Distance From Goal should be minimized in sequences. The more actions are delayed, the more likely they are to be forgotten or used incorrectly.

Goal Stack Depth (GSD) counts for each goal in a sequence how many unfinished goals precede it. An unfinished goal is one whose action has not been done yet. The Goal Stack Depth for a sequence is defined to be the maximum GSD of the goals in the sequence. Goal Stack Depth should be minimized in sequences. It is related to Distance From Goal in that a sequence of unfinished goals causes the actions that are eventually done to be far away from their goals. A large GSD is even worse than a large DFG because the actions that are pending have to be done in the right order and this order is opposite to the order in which the goals appeared.

Distance To Usage (DTU) is a measure of the distance between the calculation of a result and the first use of that result. For a sequence, the Distance To Usage is defined to be the maximum of the DTUs for its actions. Distance To Usage should be minimized in sequences. The longer usage is put off, the more intervening results there are, and the more likely that the wrong result will be used.

To illustrate the application of these measures we present excerpts from two variants of the CPA procedure. The first comes from the instruction manual used in a navy training course (FCTCPAC, 1980), and the second is one of several alternatives we ahve investigated. In the procedure taught in the school the steps which accomplish the parts of the overall soultion are mixed together. Below is the portion of the goal-action tree for finding the bearing of the CPA according to the school procedure.

	DFG	GSD	DTU
Goal: M1	-	0	-
Act: PLOT(B1, R1, GRID) => M1	0	1	1
Goal: M2	-	0	-
Act: PLOT(B2, R2, GRID) => M2	0	1	0
Goal: Line of Movement (LOM)	-	0	-
Act: RAY(M1, M2) => LOM	0	1	0
Goal: (DRM)	-	0	-
Act:TRANSLATE(LOM, P:GC) => L:DRM	1	1	0
INTERSECT(L:DRM, P:GE) => P:DRM	1	1	0
READVALUE(P:DRM) => DRM	1	1	3
Goal: Relative Distance	-	0	-
Act: READVALUE(COPY(SEGMENT ....))	0	1	1
Goal: Elapsed Time	-	0	-
Act: SUBTRACT(M2-time M1-time)	0	1	0
Goal: Relative Speed	-	0	-
Act: READVALUE(INTERSECT(RAY(PLOT.)))	0	1	4
Goal: Bearing of CPA (BC)	-	0	-
Act: ADD(DRM, +/- 90) => BC	1	1	-

This procedure does well on keeping goal stack depth low and keeping actions near the the goals they satisfy, but it does so at the expense of having a large number of top level goals making it difficult to remember. The problem is actually worse than shown here since the complete solution to the problem has 12 top level goals.

Here is the procedure rewritten with a more top-down organization:

	DFG	GSD	DTU
Goal: Bearing of CPA (BC)	-	0	-
Goal: (DRM)	-	1	-
Goal: Line of Movement (LOM)	-	2	-
Goal: M1	-	3	-
Act: PLOT(B1, R1, GRID) => M1	0	4	1
Goal: M2	-	3	-
Act: PLOT(B2, R2, GRID) => M2	0	4	0
Act: RAY(M1, M2) => LOM	2	3	0
Act:TRANSLATE(LOM, P:GC) => L:DRM	3	2	0
INTERSECT(L:DRM, P:GE) => P:DRM	3	2	0
READVALUE(P:DRM) => DRM	3	2	0
Act: ADD(DRM, +/- 90) => BC	4	1	-

This procedure has greater DFG and a greater GSD, but has only one top level goal. Expanding it to the whole CPA problem, it has only 3 top level goals and the maxima of DFG and GSD do not increase with the wider scope of the problem.

Optimizing Goal-action Sequences

Based on the measures given above we suggest the following techniques for producing goal-action sequences

-Generate from only one tree in a forest at a time to minimize Distances From Goals and Goal Stack Depths.

-Reorder subsequences to minimize Distances to Usages.

-Uproot certain subtrees and generate from them first to minimize Goal Stack Depths.

-Merge trees to reduce the Number of Top-Level Goals.

The degree to which these measures predict the ease or difficulty of procedure learning and use is, of course, an empirical question. There are certainly limits on the ranges of applicability of some measures, and tradeoffs to be maximized among them. Never-the-less an approach of this type promises to be a significant improvement over the current hit-or-miss approach to procedures composition.

References:

DMA (Defense Mapping Agency, Hydrographic Office), H.O. Publication 217, Maneuvering Board Manual. Washington D.C.: Defense Mapping Agency, 1975.

FCTCPAC (Fleet Combat Training Center, Pacific), Maneuvering Board Manual, 1980.

Moore, J. and A. Newell "How can MERLIN understand," in L. Greg [ed.] Knowledge and Cognition Erlbaum Associates, 1974, pp. 253-285.

The views expressed in this paper are those of the authors and do not necessarily represent the position of the Department of the Navy

A computer simulation approach to the study of emotional behavior[1]

Rolf Pfeifer
Department of Psychology
Carnegie-Mellon University
Pittsburgh, Pennsylvania 15213

Although the importance of emotion in human behavior has long been recognized, only recently has there been serious interest in the problem among cognitive scientists (Abelson, 1981; Bower & Cohen, 1982; Dyer, 1982; Lehnert, 1981; Norman, 1980; Mandler, 1975; Pfeifer & Nicholas, 1982; Simon, 1967; Sloman & Croucher, 1981). The present work is an effort to demonstrate that problems of emotion can be approached in an information processing framework. A first step in this direction has been taken by developing a computer simulation model capable of exhibiting certain kinds of emotional behavior. The model, dubbed FEELER (Framework for Evaluation of Events and Linkage into Emotional Responses), is used to illustrate three basic areas that a theory of emotion must deal with, namely (a) how emotions are generated, (b) what is meant by an occurrent emotion, and (c) how emotions influence our behavior. It is suggested that models or frameworks like the one to be presented will help to make the theory of emotions more accessible to cognitive psychologists, and that it provides new ways of thinking about emotional processes.

Underlying assumptions and related work

Even though the Schachter & Singer (1962) experiments have been criticized on a number of grounds (see e.g. Izard, 1977, for a summary of the criticisms), their hypothesis that emotional processes employ two separate but interacting systems, seems to be accepted by many theorists in the field (see e.g. Lyons, 1980). Stated briefly, the systems are a physiological one, the *autonomic arousal system*, and a *cognitive-evaluative* one. An *occurrent* emotion consists of two parts, a pattern of physiological arousal, *and* a cognitive-evaluative component which, in the individual's belief system, causally links this pattern to an event. A physiological pattern alone does not constitute an occurrent emotion.

The design of FEELER has been influenced by the related work of Abelson (1981), Bower & Cohen (1982), Dyer (1982), Lehnert (1981), and by Mandler's hypothesis that the psychological events that influence arousal are the ones which *interrupt* well-organized behaviors (Mandler, 1975). It is assumed that arousal is an important factor in determining the intensity of an emotion (Clark, 1982; Fiske, 1981; Mandler, 1975).

There have been a number of efforts to include emotions into computer simulation models (Colby, 1981, for example) but in most of them emotion has not been the primary focus.

General description of the model

Basic architecture: FEELER has a production system architecture which is similar to John R. Anderson's ACT model (Anderson, Kline, & Beasley, 1979), but some features have been added. As shown in Figure 1 there is a long term memory (LTM, consisting of two parts, namely a network for declarative knowledge (declarative memory) and a memory for procedural knowledge (production memory)), a cognitive working memory and a physiological working memory. Two working memories are introduced separately to account for the relative independence of the physiological and the cognitive system and their distinct characteristics (e.g. different decay rates). Whenever the term "working memory," or simply "WM" is used without further qualification, it refers to *cognitive* working memory. Similarly when just LTM is used it designates declarative memory.

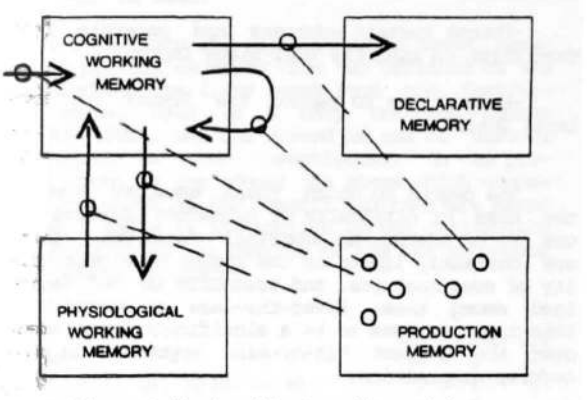


Figure 1: Basic architecture of the model

The arrows in Figure 1 depict the rules which are activated from production memory, as indicated by the circles. The tails designate which working memory they match against, the heads which memory they act upon. The action can consist of adding something to the memory, or in the case of LTM, it can be a process of spreading activation. If an element in LTM exceeds a certain activation threshold, it is automatically added to WM, where it is subject to a decay mechanism. For a discussion of spreading activation see e.g. Ratcliff & McKoon (1981). The arrow pointing into physiological working memory designates the generation of an arousal pattern.

Representation of emotional information: Since emotional experiences can be memorized and the corresponding emotions reexperienced the respective memory structures have to be defined in LTM. Emotional information which is connected to episodic memory structures includes links to the events that are responsible for the occurrent emotion, magnitudes for emotions, and a so-called *arousal image* (Clark, 1982; Mandler, 1975).

Examples of emotional behavior

Emotions generated after interrupt: Consider an example in which the model is executing a plan to take a plane trip.[2] The interrupt occurs on the way to the airport when the taxi develops a flat tire. Arousal is increased by using surprise and importance of the interrupt as multiplicative factors: if either one is small, the increase will be small, if both are large the increase will be large (see Pfeifer, 1982, for details on surprise, importance, and arousal).

Emotions are generated in this situation by emotion generation rules such as R1. R1 is adapted from Weiner's (1982) taxonomy.

R1: IF current state is negative for self and
current state was caused by person$_1$ and
person$_1$ was in control and
the emotional target is person$_1$
THEN generate anger at person$_1$

Since productions only fire if all of their conditions are present in WM, there must be a set of auxiliary productions providing the

[1] This research was supported by scholarship number 81.796.0.80 of the Swiss National Science Foundation to the author and by a grant from the Alfred P. Sloan Foundation.

[2] A model of the current environment is constantly maintained in WM.

conditions, such as R2:

R2: IF an interrupt has occurred and
emotion is to be determined
THEN determine target for emotion

Rules like R2 have to do their work for every condition before R1 can apply. The phrase "generate anger at person$_1$" means that an emotion node is created in WM which is linked to the current event structure, to the interrupting event, and to the target of the emotion. When LTM is updated, which is typically the case shortly after an interrupt has occurred, the intensity of the emotion, which is determined from the level of arousal, is attached to the emotion node, and an arousal image, consisting in the current version of a simple level indicator, is added to the current event structure.

Emotions generated after plan completion: If no interrupt had occurred on the way to the airport but instead the model had "arrived" at the airport, rule R3 might have applied:

R3: IF a subplan has been completed
THEN generate satisfaction about subplan completion

Emotions generated from emotions by rules: If anger has been generated, the emotional state of anger as such can lead to the generation of anger again by means of a rule similar to R4:

R4: IF angry and
person$_1$ is entered through perceptual system
THEN generate anger at person$_1$

R4 tries to capture the fact that if a person is angry he or she may generate anger at people who have nothing to do with the original anger-producing situation.

Emotions generated through memory activation: So far the emotion generation processes have been based on rules. Another way in which emotions can be generated is through activation processes in LTM. If elements are entered and encoded into WM through perceptual processes, activation is automatically spread through LTM, i.e. through the perceptual process itself, parts of LTM are activated and added to WM. If emotional information is attached to these elements the earlier emotions may be reexperienced: they can become an occurrent emotion. Moreover, since events in LTM are interconnected via emotion nodes, events with similar emotional qualities can be activated from the current emotional state.

Goal generation influenced by emotions: Emotions may cause certain behaviors which would not otherwise occur. Rule R5, for example, sets up the goal to harm the person (e.g. to insult, hit, yell at) who is held responsible for the individual's current negative state, which lead to the emotion of anger.

R5: IF angry and
emotional target is person$_1$
THEN generate the goal to harm person$_1$

R6: IF angry and
emotional target is person$_1$
THEN generate the goal to reassess the anger reaction

Rule R5 corresponds to a more aggressive reaction, R6 to a cautious one. R7 is a strategy to get rid of the emotion of anger by setting up a goal which diverts attention from the anger-producing situation and thus gives the anger time to decay.

R7: IF angry
THEN generate the goal to count to ten

It should be noted that the goals thus generated do not necessarily have to be pursued. This decision is up to a high-level conflict resolution mechanism.

Interpretations biased by emotions: If the action side of Rule R6 were not to set up a goal but simply to make an assumption about the world, for example "THEN assert that person$_1$ has goal to harm self," we may talk about an inference biased by an emotional state.

Summary and discussion

Table 1 is a systematic account of the possible kinds of rules involved in emotional behavior in FEELER as illustrated by the examples in the last section. The classification is based only on whether the rules directly influence emotions (i.e. they include emotions in their action side) or whether they are influenced by emotions (i.e. they include emotions in their condition side).

Cell (1) contains general inference rules which are typically used as auxiliary rules in the emotion generation process, but they are not particular to a specific emotion. Rules in cell (2) are not influenced by the current emotional state but they result in an

		ACTION SIDE: COGNITIVE	ACTION SIDE: EMOTIONAL
CONDITION SIDE	COGNITIVE	R2 (1)	R1, R3 (2)
	EMOTIONAL	R7 (3)	(4)
	COGNITIVE & EMOTIONAL	R5, R6 (5)	R4 (6)

Table 1: Summary of rules

occurrent emotion. Rules in cell (3) represent behavior which is purely motivated by an emotional state. In cell (4) are the rules defining direct interactions between emotions. So far interactions between emotions have only been modeled indirectly via the decay mechanism. Cells (5) and (6) contain rules representing interpretations or action tendencies influenced by an emotion. The rules in cell (6) lead to an emotional state which would not have been caused by the cognitive components alone.

In summary, a number of ways in which emotions can be generated and influence behavior have been modeled and analyzed. The focus in this report was on behavior based on production rules, but it was also seen that network processes participate through spreading activation mechanisms. A comprehensive concept of an occurrent emotion must include both rule-based and network-based processes, as well as their relationship to the physiological patterns of activation.

The current implementation of FEELER shows a variety of interesting kinds of emotional behaviors which have been described above. However, the representational and inference structure needs to be enriched for all aspects of the model and they have to be incorporated in a more coherent system. In addition, some issues have been only marginally addressed or not at all (e.g. learning processes, emotional expression, and high-level conflict resolution mechanisms). Despite its very real limitations FEELER provides a framework for the study of emotion in a cognitive science methodology capable of capturing a wide range of phenomena. Applications to research on mood and to the theory of defense mechanisms are briefly pointed out elsewhere (Pfeifer, 1982).

Acknowledgments

Many discussions with Peggy Clark, Susan Fiske, Matt Lewis, David Nicholas, and Herb Simon, have been invaluable to the development of the ideas in this paper. I would like to thank in particular Bill Jones, Matt Lewis, Peter Pirolli, Barbara Riehle, and Herb Simon for comments on an earlier draft, and Pat Langley for his assistance with the implementation of the model.

References

Abelson, R.P. Constraint, construal, and Cognitive Science. In

Third Annual Conference of the Cognitive Science Society. Cognitive Science Society, 1981.

Anderson, J.R., Kline, P.J., Beasley, C.M. A general learning theory and its application to schema abstraction. In G. Bower (Ed.), *Advances in learning and motivation*. New York: Academic Press, 1979.

Bower, G.H. & Cohen, P.R. Emotional influences in memory and thinking: Data and theory. In M.S. Clark & S.T. Fiske (Ed.), *Affect and cognition: The 17th Annual Carnegie Symposium on Cognition*. Hillsdale, N.J.: Erlbaum, 1982.

Clark, M.S. A role for arousal in the link between feeling states, judgments and behavior. In M.S. Clark & S.T. Fiske (Ed.), *Affect and cognition: The 17th Annual Carnegie Symposium on cognition*. Hillsdale, N.J.: Erlbaum, 1982.

Colby, K.M. Modeling a paranoid mind. *The Behavioral and Brain Sciences*, 1981, 4, 515-560.

Dyer, M.G. *In-depth understanding. A computer model of integrated processing for narrative comprehension*. Doctoral dissertation, Yale University, 1982.

Fiske, S.T. Social cognition and affect. In J. Harvey (Ed.), *Cognition, social behavior and the environment*. Hillsdale, N.J.: Erlbaum, 1981.

Izard, C.E. *Human Emotions*. New York: Plenum Press, 1977.

Lehnert, W.G. Affect and memory representation. In *Third annual conference of the Cognitive Science Society*. Cognitive Science Society, 1981.

Lyons, W. *Emotion*. Cambridge, UK: Cambridge University Press, 1980.

Mandler, G. *Mind and emotion*. New York: Wiley, 1975.

Norman, D.A. Twelve Issues for Cognitive Science. *Cognitive Science*, January-March 1980, 4(1), 1-32.

Pfeifer, R. *Cognition and emotion: an information processing approach* (Tech. Rep. 436). Dept. of Psychology, Carnegie-Mellon University, May 1982. C.I.P. Working Paper.

Pfeifer, R. & Nicholas, D.W. Toward computational models of emotion. In *Proceedings of the European Conference on Artificial Intelligence*. AISB, 1982.

Ratcliff, R. & McKoon, G. Does activation really spread? *Psychological Review*, 1981, *88*(5), 454-462.

Schachter, S. & Singer, J.E. Cognitive, social and physiological determinants of emotional state. *Psychological Review*, 1962, 69, 379-399.

Simon, H.A. Motivational and emotional control of cognition. *Psychological Review*, 1967, 1, 29-39.

Sloman, A. & Croucher, M. Why robots will have emotions. In *Seventh International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence, 1981.

Weiner, B. The emotional consequences of causal ascriptions. In M.S. Clark & S.T. Fiske (Ed.), *Affect and cognition: The 17th Annual Carnegie Symposium on Cognition*. Hillsdale, N.J.: Erlbaum, 1982.

Where Do Goals Come From?

Jaime G. Carbonell
Carnegie-Mellon University
Pittsburgh, PA 15213

Abstract

Theories of rational behavior embodied in cognitive models of problem solving, planning, and plan interpretation typically presuppose that the planning agent is given *a priori* one or more goals to pursue. Thereupon, rational behavior consists of planning and carrying out a sequence of actions in order to achieve the most important active goals. This paper argues that a complete cognitive model must necessarily incorporate *the process of acquiring goals* whether in reaction to perceptions of external events, in response to internal physiological or psychological states, or by other less direct means. An initial categorization is made of various mechanisms that can give rise to goals in an individual planner.

1. Introduction

The AI literature abounds with models of problem solving, planning and plan interpretation (e.g., GPS [11], STRIPS [6], NOAH [14], PAM [18], BELIEVER [16], TALESPIN [10], POLITICS [5, 3]). Although these models differ in terms of the specific cognitive phenomena simulated, in terms of their internal structure, in terms of their representation formalisms, and in terms of their theoretical motivations, it is striking that they all share one central hypothesis: Each and every system is heavily dependent upon the presence of one or more goals attributed to the active problem solving agents or planners. In essence, each planner or problem solver incorporates an implicit theory of rational behavior based upon the assumption that all actions are preformed in service of explicit, realizable goals. Therefore, rational behavior for a planning system consists of formulating a sequence of planned action to achieve a set of goals. In the case of story interpretation, the assumption of rationality applies to the characters, and the task of the understander becomes one of divining their goals by reconstructing corresponding plans from sequences of observed events.

Hence, under these models of planning and plan interpretation, rationality becomes synonymous with intelligence. Or, as Newell defines it: Intelligence is the ability to bring knowledge to bear in the pursuit of goals [12]. Wilensky [19] also articulates the notion that all intelligent action ensues from the pursuit of multiple goals, including the resolution of internal goal conflicts by the spontaneous creation and subsequent pursuit of *metagoals*. The implicit centrality of goals becomes more evident when one considers some attempts at modeling affect or idiosyncratic behavior. For instance, Lehnert's affect states [9] in story interpretation, and recent work on modeling emotions [1, 13] rely on mechanisms to detect goal frustration or goal achievement. My earlier work on modeling ideological belief and certain aspects of human personality traits relies even more heavily on the presence, pursuit and attribution of different types of goals to planning agents [2, 5].[1]

2. Goal Generators in Integrated Cognitive Models

If goals are central to all effective AI theories of intelligence, the natural question arises: *Where do goals come from?* Whereas taxonomies of goals [15], relations among the goals of an individual [5, 19], and methods of planning to achieve goals are all significant aspects of the study of goals, the key notion of *what cognitive, physiological or social mechanisms give rise to goals* has been largely glossed over by AI researchers. An AI program, whether planner or problem solver, does nothing until an external entity (such as the programmer) provides it with a goal to pursue, whereupon the program single-mindedly strives to find an effective plan for that goal, and regardless of success or failure, resumes idling indefinitely after the solution attempt. Clearly, any complete cognitive model must generate its own goals. Philosophical debate on issues of free-will vs determinism notwithstanding, all intelligent beings exhibit some measure of internal motivation and ability to respond to unexpected situations in the external environment.

The type of integrated cognitive model I envision would contain a goal generator that would monitor continuously the external environment and its internal state as a background process, and hence it would *notice* if it is getting hungry or tired, or that an external threat is imminent, bringing these issues (perhaps as interrupts) to the attention of the conscious "rational" processor, which then may decide to generate new goals, reprioritize existing goals, or ignore the interrupts. Essentially, the continuous monitoring of possible sources of goals necessarily forces one to face the issue of focus of attention, an issue that can be safely ignored only as long as an external entity provides all goals and thereby limits distracting factors. In fact, the single-minded pursuit of a small set of externally imposed goals determined *a priori* obviates the need to refocus attention dynamically as no unforeseen happenings will be noticed. Consider a present-day AI planning system deciding, for example, how to stack blocks. When faced with an external threat or a greater need, it will not have the sense to abandon or postpone its present task, generate and pursue a more appropriate goal, and thereby change the current focus of attention.

Rather than attempting the formidable task of characterizing the space of plausible cognitive models capable of directing their own attention, and responding to changing events by generating their own set of appropriate goals, let us focus on the more tractable subproblem of exploring various mechanisms capable of generating goals dynamically.[2] From a psychological standpoint, an obvious source of goals is the internal physiological state of the planning agent: Hunger leads to the goal of satiation of hunger; physical exhaustion leads to a desire for rest. From an AI standpoint, an equally obvious source of goals is the planning system itself generating subproblems, with the associated goal of solving the subproblem. For instance, an AI planner may decide that, given the externally imposed goal of "satiate hunger", it should first locate food, then transport itself to that location, then ingest the food. Each of these steps, if not immediately executable in the external world, generates a subgoal requiring additional planning (e.g., locating food generates the subgoal of knowing the location of the food, which then may lead to searching or asking, etc.) There are, however, more complex sources of goals. Schank and Abelson postulate a set of *themes* as goal generators whose internal structure remains a virtual black box. For instance, the *love theme* generates the goal of protecting one's loved ones. Unlike other aspects of Schank and Abelson's theory of representation and understanding, their treatment of themes does not provide a very satisfying analysis, in that it neither postulates a computational mechanism for how these themes operate or are

---

[1] In this argument I do not mean to imply that all theories of emotion or even theories of human intelligence interesting to AI practitioners are necessarily based on goals and their unrelenting pursuit. I am merely noting that theories precise enough to result in operational process models (e.g. AI programs) incorporating significant aspects of human cognition have *thus far* been dependent on goals and the implicit principle of rational behavior.

acquired, nor does it attempt exhaustive coverage or broad sampling of cognitively plausible goal generators. Here, we pursue the latter goal with the longer range objective of eventually developing computational mechanisms that give rise to goals in the context of a complete cognitive model.

3. Towards a Taxonomy of Goal Generators

Let us again pose the central question: *Where do goals come from?* However, rather than examining the literature for possible answers as I attempted above, let us enumerate and categorize possible goal generators in humans. It appears that the following general categories cover a large range, if not the entire space of goal generators:

1. Internal physiological state changes
2. Mental (e.g., emotional or attitudinal) state changes, possibly accompanied by, or resulting from physiological state changes
3. Knowledge state changes
4. Perceptions of changes in the external world
5. Socially imposed goals or constraints on the individual
6. Instrumentality (i.e., goals generated purely in service of other goals)

Examining this list, several observations become readily apparent:

- General coverage is indeed attained, in the sense that goals typically attributed to people can be coerced into a combination of one or more of the categories above.
- This list is of very little use in developing a process model, as it lacks commitment to any *fine-structure detail*.[3] Generality is not the only metric one should apply in judging the utility of a theoretical concept.
- The classification itself does not necessarily suggest that a uniform mechanism operates within each category giving rise to the set of goals thus grouped together. Therefore, if the analysis is to be useful in constructing a predictive, psychologically plausible, process model, the categorization must be motivated more strongly by the *processes* that operate in generating the classes of goals grouped together.

Bearing these concerns in mind, let us construct a more detailed categorization motivated by commitment to finer-structure detail of the processes that generate goals, and let us place less emphasis on global generality at this stage of the investigation. In the taxonomy of goal generators presented below, the hierarchical structure is meaningful, as are the suggested mechanisms, but the order in which the categories are listed is quite arbitrary.

1. **INSTRUMENTALITY**

   a. **Direct instrumentality** -- Given a higher level goal, subgoals are generated by the planning or problem solving process whenever a step in the plan to achieve the higher level goal is not directly realizable, and hence requires additional directed planning. These goals correspond to Schank and Abelson's "delta goals" [15].

   b. **Derived or indirect instrumentality** -- Secondary goals instrumental to the achievement primary goals arise through several mechanisms in addition of strict subgoal instrumentality, to wit:

      i. In the process of planning to achieve more than one primary goal, conflicts may arise among active goals of the planner giving rise to *metagoals* [19] of resolving the internal goal conflict in order for the planner to achieve all (or the most crucial subset) of his primary goals. Typically these conflicts are based on resource limitations, including limitations on the time that the active planner can devote to a particular set of tasks.

      ii. In the *counterplanning process* [4, 5], instrumental goals of assuring that an adversary cannot (or will not) thwart an otherwise viable plan arise frequently. These are not true subgoals, in that they may play no role in achieving the primary goal, but rather may be directed at misleading, diverting or negotiating with potential adversaries.

      iii. *Goal subsumption states* [19] arise when a primary goal recurs frequently, or many primary goals share a common instrumental subgoal. In essence, a subsumption state facilitates the achievement of many instances of primary or instrumental goals. Hence, the achievement of a desired subsumption state becomes a goal in itself. An instance of a subsumption state is having a steady income, thus facilitating any goals requiring money, and aiding social-status goals as well. Similarly, establishing an alliance to aid in future mutual fulfillment of different primary goals, or terminating an adversary relation can be considered subsumption goals [5].

      iv. *Optimization* of a plan, or *saving mental effort* while planning could be construed as indirect instrumental goals to the primary objective.

2. **INTERNAL DRIVES** -- these may be considered psychologically inate goals in an individual

   a. **Cyclic physiological drives** -- these are goals generated in response to internal physiological states that change with a certain periodicity. A cognitive model may treat the mechanism that generates basic drives of this sort as a black box. Schank and Abelson label these "Sigma goals". A partial enumeration of cyclic physiological drives includes:

      i. Satiation of hunger

      ii. Satiation of thirst

      iii. Desire for rest or sleep

      iv. Desire for sexual activity

   b. **Non-cyclic physiological drives** -- these occur primarily in response to adverse changes in the environment, and perhaps should also be considered as black boxes when constructing a cognitive model. These goals have no correlate in the Schank and Abelson taxonomy. A representative sampling includes:

      i. Self-preservation (in response to overt threats)

      ii. Protection of one's offspring (again in response to overt threats)

      iii. Seeking warmth (if the external temperature drops)

      iv. Satisfying curiosity (e.g., in response to unexpected external events)

      v. Seeking companionship (in its absence)

3. **SOCIAL GOALS** -- These are goals that arise by virtue of interaction with other members of the species.

   a. **Semi-autonomous social dynamics** -- these goals

---

[2] The reader is referred to the "World Modeller's Project" [8, 7] for a discussion of a general experimental system that simulates a reactive environment in which one may build simple planning systems that must cope with changes in the environment. Such a system is an experimental tool that expedites research and sheds light on significant problems not heretofore investigated in the appropriate context. (Such problems include the topic of this paper.)

[3] Sloman argues convincingly that evaluating a theory based solely on breadth of coverage and predictive generality ignores issues of internal structure and commitment to detail, which often differentiate useful theories from general truisms [17].

appear to require no explicit learning, but arise only if an individual interacts with other members of the species. Again, these goals have no direct correlate in Schank and Abelson's taxonomy. Types of semi-autonomous social goals include:

i. Simple social ambition (e.g., become the king of the hill, or the leader of the pack, or the respected medicine man)

ii. Property ownership, acquisition and protection from others (There can be no meaning to ownership without the notion of restricting access to others of the objects owned.)

iii. Protection of others within the social group from external threats (This clearly goes beyond protection of self or biological offspring)

iv. Protection of the nature and makeup of the social group itself (e.g. from other members of the species who may pose no threat to individuals within the social group, but pose a threat to the established social order)

v. Jealousy, wanting something merely because another member of the social group has acquired it

vi. Avoid banishment by the social group

b. **Socially taught or imposed goals** -- unlike the previous category, these goals vary across social groups within the species, and therefore must be learned by individuals (from observation of more mature members of the social group, or by direct instruction). Here I defer to anthropologists or social psychologists to provide a more comprehensive list; the following is meant as an illustrative sample:

i. Abide by the formal and unwritten laws of the society

ii. Live according to the ethics and morals adopted or imposed by the society on the individual

iii. Contribute to the communal wealth and well being (in some societies)

iv. Seek to attain those qualities that comprise a metric of status in the society (wealth, power, respect, wisdom, notoriety, etc. depending on the particular society)

4. ENJOYMENT GOALS -- these correspond roughly with Schank and Abelson's "E-goals".

a. **Direct (physiological) pleasurable experience** -- these goals overlap substantially with cyclic and other physiological goals discussed earlier; the central distinction is based on the circumstances in which they arise (e.g., the motivation to walk into a hot tub or a steam bath differs from the motivation to seek shelter in frigid weather, although the resulting goal states overlap in terms of the physical state change sought).

i. Physical exertion for pleasure (as opposed to exertion instrumental to other primary goals), such as exercise, some forms of children's play, etc.

ii. Direct sensual gratification (such as eating for pleasure in "gourmet" dining, tactile gratification, etc.)

iii. Aesthetic gratification (such as enjoying a painting, a sunset, a concert, a good novel, etc.)

b. **Derived psychological pleasure** -- satisfaction of most non-trivial goals yields a measure of resultant pleasure, but some goals appear to be caused by no internal or external reason other than experiencing this measure of indirect pleasure. For instance:

i. Vicarious pleasure (role playing, identification with characters in movies, novels or sporting events, etc.)

ii. Acquisition of knowledge for its own sake, when the knowledge is not instrumental to any primary goals, nor is its presence a realistic subsumption state (e.g., assorted trivia, half of the features stories in newspapers and magazines that bear no impact on any conceivable goal of the reader, intellectual curiosity, etc.)

iii. Acquisition of objects for their own sake (For instance, most stamp and coin collectors are not primarily motivated by the prospect of making money from their collections, but rather amassing and classifying their precious objects becomes an end in itself.)

5. MENTALLY-DERIVED GOALS -- these are goals resulting from deliberate reasoning processes, including:

a. Goals arising from mentally deduced information (as opposed to directly observed information). These goals may bear similarity in content with previous goals, but not in their method of inception (such as deciding that the disturbance in the campsite could have been caused by a grizzly bear, and hence activating the self-preservation goal).

b. Goals arising from the result of purposeful reasoning (such as deciding on a particular career to pursue after much thought). These are not instrumental goals, but often long-range personal-objective goals.

4. Concluding Remark

The goal categorization above, however imperfect or incomplete, is offered as an initial step towards developing effective models of the goal acquisition process, and thereby eventually creating more complete models of human cognition. Subsequent to the postulation of a particular taxonomy motivated by plausible sources of the various classes of goals, I intend to focus on modeling explicitly a planning agent that acquires its own goals and refocuses its attention in an interrupt-driven manner. The World Modellers project offers an amenable environment in which to create progressively more complex, cognitively plausible models that interact with a simulated environment.

5. References

1. Bower, G. H. & Cohen, P. R., "Emotional influences in memory and thinking: Data and theory," in *Affect and cognition: The 17th Annual Carnegie Symposium on Cognition*, M.S. Clark & S.T. Fiske, ed., Erlbaum, Hillsdale, N.J., 1982.

2. Carbonell, J. G., "Towards a Process Model of Human Personality Traits," *Artificial Intelligence*, Vol. 15, No. 1,2, november 1980 , pp. 49-74.

3. Carbonell, J. G., "POLITICS: An Experiment in Subjective Understanding and Integrated Reasoning," in *Inside Computer Understanding: Five Programs Plus Miniatures*, R. C. Schank and C. K. Riesbeck, eds., New Jersey: Erlbaum, 1981.

4. Carbonell, J. G., "Counterplanning: A Strategy-Based Model of Adversary Planning in Real-World Situations," *Artificial Intelligence*, Vol. 16, 1981 , pp. 295-329.

5. Carbonell, J. G., *Subjective Understanding: Computer Models of Belief Systems*, Ann Arbor, MI: UMI research press, 1981.

6. Fikes, R. E. and Nilsson, N. J., "STRIPS: A New Approach to the Application of Theorem Proving to Problem Solving," *Artificial Intelligence*, Vol. 2, 1971 , pp. 189-208.

7. Hood, G. and Carbonell, J. G., "The World Modelers Project: Constructing a Simulated Environment to Aid AI Research," *Proceedings of the Thirteenth Annual*

Pittsburgh Conference on Modeling and Simulation, 1982, Pittsburgh, PA.

8. Langley, P., Nicholas, D., Klahr, D. and Hood, G., "A Simulated World for Modelling Learning and Development," *Proceedings of the Third Annual Conference of the Cognitive Science Society*, 1981.

9. Lehnert, W.G., "Affect units and narrative summarization," Tech. report 179, Yale Univ., Dept. of Computer Science, May 1980.

10. Meehan, J. R., *The Metanovel: Writing Stories by Computer*, PhD dissertation, Yale University, Sept. 1976.

11. Newell, A. and Simon, H. A., *Human Problem Solving*, New Jersey: Prentice-Hall, 1972.

12. Newell, A., "The Knowledge Level," Tech. report, Dept. of Computer Science, Carnegie-Mellon University, 1981, CMU-CS-81-131.

13. Pfeifer, R., "Cognition and emotion: an information processing approach," Tech. report 436, Dept. of Psychology, Carnegie-Mellon University, May 1982, C.I.P. Working Paper.

14. Sacerdoti, E. D., *A Structure for Plans and Behavior*, Amsterdam: North-Holland, 1977.

15. Schank, R. C. and Abelson, R. P., *Scripts, Goals, Plans and Understanding*, Hillside, NJ: Lawrence Erlbaum, 1977.

16. Schmidt, C., Sridharan, N. and Goodson, J., "The Plan Recognition Problem," *Artificial Intelligence*, Vol. 11, No. 2, 1978, pp. 45-83.

17. Sloman, A., *The Computer Revolution in Philosophy: Philosophy, Science And Models of the Mind*, Harvester Press, 1978.

18. Wilensky, R., *Understanding Goal-Based Stories*, PhD dissertation, Yale University, Sept. 1978.

19. Wilensky, R., *Planning and Understanding*, Addison Wesley, Reading, MA, 1983.

Surprise and Coherence:
Sensitivity to Verbal Humor
in Right Hemisphere Patients

Hiram H. Brownell, Dee Michel,
John Powelson, and Howard Gardner

Boston Veterans Administration Medical
Center and Aphasia Research Center,
Dept. of Neurology, Boston University
School of Medicine, and
Harvard Project Zero

Address for correspondence:
Psychology Service 116B,
Veterans Administration Medical Center,
150 S. Huntington Ave., Boston, MA 02130.

Surprise and Coherence:
Sensitivity to Verbal Humor
in Right Hemisphere Patients

Jokes reflect one of the most intriguing human competences. In this paper, we focus on jokes as a narrative form and examine how they are processed by patients with cortical brain damage. These results provide empirical support for theoretical components of normal humor processing.

Jokes have been subjected to considerable analysis by scholars representing several disciplines. Despite numerous differences in focus, nearly all formulations about jokes stress the importance in humor of incongruity: A feature or features are surprising and unexpected at one level, but follow plausibly when another level or dimension is considered (see Goldstein and McGhee, 1972; McGhee, 1979, for reviews). Take, for example, the following joke:

> The neighborhood borrower approached Mr. Smith on Sunday afternoon and inquired: "Say Smith, are you using your lawnmower this afternoon?"
>
> "Yes, I am," Smith replied warily.
>
> The neighborhood borrower then answered: "Fine, then you won't be needing your golf clubs. I'll just borrow them."

Upon hearing the body of the joke, the listener has an expectation of what will follow plausibly: For example, the borrower will be disappointed, or he will ask to borrow the lawnmower on a subsequent occasion. The punchline is surprising at one level precisely because it departs radically from these expectations about the normal course of events. What converts the feeling of surprise into a reaction of humor is the fact that, viewed from a different perspective, the punchline does follow from the premises introduced in the body of the joke. After all, the borrower does end up asking for a loan, and the wariness displayed by the possessor of the lawnmower provides the perfect pretext for the second request.

This analysis identifies two potentially separable components of jokes, termed here surprise and coherence, that are utilized in the normal appreciation of verbal humor. Assuming for the moment that an individual has an intact understanding of the ordinary meanings and uses of language, he must also possess a schema, or script (cf. Abelson, 1981), which covers the normal course of events (in this case, a request to borrow an item from a neighbor). Against this background, the individual must be able to detect discrepancies from the normal course (sensitivity to surprise). However, in order to appreciate the joke, mere detection of discrepancy does not suffice; the listener must be able to appreciate the relation among the elements in the body of the joke and keep them sufficiently in mind so that he can attempt to relate them to the punchline (appreciation of coherence).

Of course, appreciation of jokes requires syntactic and lexical-semantic skills. In view of this account, two questions arise. First, can narrative competences such as sensitivity to surprise and the ability to generate a coherent interpretation of the punchline in the light of the joke's beginning be impaired apart from other linguistic abilities? Second, can the two hypothesized components of the joke narrative form be distinguished empirically?

Patients with unilateral right hemisphere disease provide a useful population for studying these issues. First, these patients have superficially intact syntactic and semantic capacities and so, unlike aphasic patients who have damage in the left cerebral hemisphere, their difficulties with jokes or other forms of connected discourse cannot be attributed to difficulties in processing individual words or sentences. Second, it has recently been suggested by Wapner, Hamby and Gardner (1981) that right hemisphere patients can understand the details of a story but may have difficulty weaving them together into a single coherent interpretation. According to this line of analysis, right hemisphere damaged patients should understand the details presented in the body of a joke but may demonstrate difficulty relating the punchline to the body of the joke. They should detect when a punchline is at variance with the overt content of the rest of the joke and yet may prove unable to find in the joke a second level of interpretation that integrates the punchline with the body of the joke. Right hemisphere damage, then, may selectively impair patients' sensitivity to one of two vital components of verbal humor.

Method

To secure information on these issues, a joke completion task was administered to right hemisphere damaged stroke patients and to a set of matched normal controls. The task required a subject to listen to the body of a joke and then to select from a set of four alternatives the correct punchline.

To illustrate each of the four types of alternative, consider the joke described above:

> The neighborhood borrower approached Mr. Smith on Sunday afternoon and inquired: "Say Smith, are you using your lawnmower this afternoon?"
>
> "Yes, I am," Smith replied warily.
>
> The neighborhood borrower then answered:

1) Correct ending: "Fine, then you won't be needing your golf clubs, I'll just borrow them."

2) Nonsequitur ending: "You know, the grass is greener on the other side."

This latter ending, like the correct punchline, includes an element of surprise - it does not follow directly from the joke's beginning. However, unlike the correct ending, the nonsequitur could not be coherently integrated with the premises on a second level to form an acceptable resolution to the joke's story. Thus, the choice of a nonsequitur ending would indicate a preserved sensitivity to the surprise component of humor, but an inability to integrate the body of the joke and its punchline into a coherent interpretation.

The nonsequiturs were divided into two groups. Half were topically unrelated to the body of the joke, and half were topically related to the body of the joke, including, for instance, a word associated with an element of the joke. Of this last group, half were common sayings. The nonsequitur above, for example, is a proverb ("The grass is greener on the other side"), in which "grass" in related to "lawnmower". Neither of these factors of topical relatedness or familiarity as a proverb had a significant effect, and they will not be discussed in detail.

3) Straightforward neutral ending: "Do you think I could use it when you're done?"

This ending follows directly from the joke's beginning. The straightforward endings complemented the nonsequiturs in that they preserved a coherent sense of story but provided no disconfirmation of expectations; choice of this incorrect ending would indicate an insensitivity to the importance of surprise in humor.

4) Straightforward sad ending: "Gee, if only I had enough money, I could buy my own."

The straightforward sad endings, like the straightforward neutral endings, are coherent but provide no disconfirmation of expectancies; in addition, they reflect on characters mentioned in the joke in a sad or pathetic fashion. Choice of this ending would indicate not only an insensitivity to the importance of surprise in humor, but also an attraction to negatively toned emotional content.

Results and Discussion

Data analysis was performed in two stages. First, subjects' proportions of correct choices were examined. In this analysis of variance, there was a clear effect of subject group, $p < .05$; the right hemisphere subjects (mean proportion correct = .60) performed significantly worse overall than did the normal controls (mean proportion correct = .81). This result provides a clear demonstration that right hemisphere damage, and possibly brain damage in general, results in a humor deficit.

In the second stage of data analysis, subjects' error patterns were examined more closely. On any trial, if a subject did not choose the correct alternative, he might have chosen any of the three incorrect alternatives. Three separate ANOVA's, which as a group were independent of the original analysis of proportion correct, were performed -- one for each error type. A data point in these analyses consisted of the number of times a subject chose a certain type of ending from among the three incorrect alternatives, divided by his total number of errors. Neither the straightforward neutral endings nor the straightforward sad endings ANOVA's revealed any effects that approached significance, $F < 1.0$ for both ending types. However, analysis of subjects' choice of the nonsequitur endings (collapsing across the three subtypes of nonsequitur) showed that the right hemisphere subjects were significantly more attracted to this ending type (mean proportion of total errors = .50) than were the normal controls (mean proportion of total errors = .18), $p < .01$. Error data from the right hemisphere subjects were further examined using t-tests for effects of the three subtypes of nonsequitur endings. These tests did not reveal any reliable effects, although within the associated nonsequiturs, the common sayings were marginally ($p < .10$) more attractive than the unfamiliar associates.

In summary, there are two major results of this experiment. First, the right hemisphere patients showed a marked disability relative to control subjects in selecting correct punchlines. Second, right hemisphere patients were clearly more attracted to or fooled by the nonsequitur endings than were the normals. This pattern of results supports a model of humor processing based on two narrative skills: the ability to detect surprise, and the capacity to establish coherence, in these cases between the surprising ending and the body of the joke. The confusion by right hemisphere patients between the nonsequitur and the correct endings suggests a preservation of the first narrative skill and an impairment of the second. The right hemisphere patients appreciate that a joke must end in a surprise, and they recognize which endings are surprising; but they cannot establish a second level of interpretation that ties the ending coherently to the body of the joke.

The present study does not establish whether this impairment is the result of right hemisphere damage specifically, or of brain damage in general. The obvious control for unilateral right hemisphere disease - unilateral left hemisphere disease - is of course inappropriate because of the effects of aphasia. Similarly, the study does not conclusively demonstrate a dissociation between narrative competence and linguistic competence; it only suggests that a narrative skill can be impaired in the face of intact linguistic ability at the sentence level. Nevertheless, an inability to integrate the body of a joke and its ending into a coherent interpretation is consistent with earlier claims (cf. Wapner et al., 1981) that right hemisphere patients exhibit an inability to integrate content across parts of a narrative unit.

REFERENCES

Abelson, R.P. Psychological status of the script concept. American Psychologist, 1981, 7, 715-729.

Goldstein, J.H., & McGhee, P.E. (Eds.) The psychology of humor: Theoretical perspectives and empirical issues. New York: Academic Press, 1972.

McGhee, P.E. Humor: Its origin and development. San Francisco: W.H. Freeman, 1979.

Wapner, W., Hamby, S., & Gardner, H. The role of the right hemisphere in the apprehension of complex linguistic materials. Brain and Language, 1981, 14, 15-32.

Language dominance and gesture hand preferences

Debra Stephens
The University of Chicago
(Cognitive Science Society sponsor: David McNeill)

In general, adults gesture only when speaking, and with one hand more than with the other (Kimura, 1973a, b; McNeill & Levy, 1982; Sousa-Poza, Rohrberg & Mercure, 1979). Kimura (1973a) observed the hand motions of right-handed adults during speech, nonverbal vocalization (humming), and during the silent performance of a verbal and a nonverbal task. A subject's hands were empty throughout a session. She found that most of the hand movements were classifiable either as self-touching (e.g., pushing back hair, adjusting eye glasses) or as "free movements--any motion of the limb which did not result in touching of the body or coming to rest (p. 46)." Self-touching occurred frequently during all activities, while free movements (which we shall hence-forth call gestures) were limited almost exclusively to the speaking condition. Moreover, subjects displayed no hand preference in self-touching, but a right-hand predominance in gesturing.

Kimura (1973b) recorded gestures of both right-and left-handers during spontaneous speech. A subject was classified as sinistral or dextral if he or she wrote and performed at least six of seven other common activities (e.g., combining hair, striking a match) with the given hand. Language dominance was inferred from left- or right-ear superiority in the perception of words presented in a dichotic listening task. Right-handers with inferred left hemisphere language, as well as left-handers with inferred right language, gestured primarily with the dominant hand, which is opposite and presumably controlled by the hemisphere dominant for speech. Sinistrals with inferred left hemisphere language gestured about equally often with either hand. Since all the left-handers demonstrated a strong left hand preference in performing other activities, the difference between the two groups may reflect discrepant organization of expressive language functions, with greater bilateral representation in the subjects who displayed no hand preference in gesturing. Although the dichotic test indicates left dominance in this group, it is, as Kimura points out, primarily a perceptual task; and studies of brain-damaged populations suggest that left-handers are more likely than right-handers to have diffusely organized language functions (e.g., Hecáen & Piercy, 1956; Marcie, 1972; Milner, Branch & Rasmussen, 1964).

Our own preliminary observations largely confirm Kimura's findings. We videotaped each of 23 adults narrating an animated cartoon he or she had just seen, to a listener who had not viewed it. Six subjects were participants in a study by McNeill and Levy (1982), the primary purpose of which was not to examine hand preference in gesturing, but to illuminate the ways that speech and various types of gestures represent the speaker's conceptual structures. Four of the six subjects in that investigation reported later, by telephone, that they write and perform other common activities with the right hand, and the other two were self-reported left-handers. Subsequently we analyzed the gestures of an additional six dextrals and 11 sinistrals narrating the same cartoon. We required each of these 17 subjects not only to report his or her preferred hand for performing nine common activities (e.g., brushing teeth, eating with a spoon), but also to pantomime each action, and to write a short phrase. We classified a subject as right- or left-handed on the basis of the hand preferred for writing. All 17 subjects reported that they always write with the same hand. In more than 99% of the cases, reported hand preference for the other nine tasks matched the hand used in pantomiming. Right- and left-hand preferences on a task were scored respectively as 1 and -1, and the absence of a preference received a zero. Thus an overall score of 9 indicates strong dextrality, and a -9 maximum sinistrality.

The subjects were also administered a questionnaire regarding the handedness of immediate family members (parents, grandparents and siblings). Each was assigned an index of familial sinistrality, which we computed using the method described by Levy and Reid (1978, p. 135). Every left-handed or ambidextrous parent or sibling was weighted as 1, and each left-handed or ambidextrous grandparent was assigned a weight of .5. The weights were totaled and divided by the number of family members whose handedness the subject reported. This index did not correlate with gesture hand preference, or with the measure of general hand preference.

We classifed almost every gesture (i.e., more than 80%) of each subject either as "iconic" or as a "beat" in accordance with the criteria devised by McNeill and Levy. An iconic gesture is one which "seems to bear a formal similarity to some aspect of the situation described by the accompanying speech (p. 272)." For example, most of our subjects accompanied a description of a cat climbing up a drainpipe with a gradual upward motion of one hand. In this case both speech and gesture describe the direction of the cat's movement. A beat, on the other hand, is "small and formless, often quickly made (p. 273)." It shows no relation to the speech content but is associated with the discourse structure. Two lines of argument led us to suspect that iconics in particular would be generated by the speech-dominant hemisphere, while beats might be produced by either. First, the former are intimately tied to speech content, while the latter are not. McNeill and Levy postulate that in fact an iconic gesture and the accompanying utterance emerge from a common conceptual representation. Second, iconics involve sequences of movements, while beats are discrete motions. Kimura and Archibald (1974) found that a group of aphasics was impaired in performing manual sequences, but not on tasks requiring single motions. Beats are not only simple and largely devoid of content, but insofar as they are associated with discourse structure, are connected to a function that may involve the whole brain performing in an integrated manner. This is because discourse planning includes an interrelation of global and sequential planning which could draw on the special skills of both sides of the brain.

Shown in Table 1 is the index of general hand preference, for the 17 subjects from whom these data were obtained. In addition, for each of the 23 subjects, Table 1 displays the number of surface grammatical clauses in the narration. We define a

clause as any linguistic unit containing precisely one subject and predicate, either of which might not be explicitly stated, but inferred from context. Finally, the numbers of iconics and beats performed with the left, right, and both hands, respectively, are presented. As shown in Table 1, seven of our ten dextrals made iconic gestures primarily with the right hand or with both hands, and much less frequently with the left hand alone. Two of the other three performed iconics with the right hand almost exclusively, while the remaining right-hander showed a predominance of left-handed iconics. The pattern for beats is more complicated: five subjects show a right hand preference, four a left-hand one, and one performed mostly two-handed beats.

As Table 1 indicates, three of the 13 sinistrals produced more iconics with the right hand, than with the left or both. Four performed a greater number of two-handed than left- or right-handed iconics, and the remaining six left-handers displayed a left hand preference. Most sinistral individuals showed the same preference in making beats, as in producing iconics, though the numbers of beats are rather small in many cases.

For tasks other than gesturing, all right-handers received scores indicating strong dextrality. The variation in the scores for left-handers prompted us to compute the correlation between this index and the respective percentages of left-, right- and two-handed iconics and beats, for this group alone. As the hand preference score decreases, signifying an increase in strength of left hand preference, the percentage of left-handed iconics rises ($r = -.67$, $df = 11$, $p<.05$), and the percentage of two-handed iconics decreases ($r = .64$, $df = 11$, $p<.05$). No significant correlations were found for beats.

For each subject, we divided the total number of iconics, the total number of beats, and the sum of both, by the number of clauses in the narration, thus obtaining measures of the rate at which the two types of gestures were produced, separately and in combination. Right- and left-handers produced iconics at about the same rate, but the former performed, on the average, one beat for every four clauses, while the latter made one beat for every six clauses.

We also wished to determine if hand preference for iconics was associated with aspects of the gestures themselves. First, we checked to see if direction of lateral motion varied with gesture hand. Most subjects used the left and right hands about equally often to gesture either to the left or to the right. Interestingly, though, subjects usually reproduced actions in the direction they were performed in the cartoon, from the watcher's perspective. Thus a gesture depicting a cat running to the subject's right was likely to involve a rightward hand motion.

Second, we searched for systematic differences in the meanings of iconics performed with the preferred versus the non-preferred hand. Here we noted whether the action depicted in the gesture was that of a major or minor character, and if major, whether the active pursuer (the cat) or the pursued (a bird). We hypothesized that the preferred gesture hand would portray the cat's actions, and that the other hand would depict those of the bird and of the minor characters. However, either hand was equally likely to describe the actions of any character.

In addition, we examined the speech accompanying iconic gestures of the preferred and non-preferred hands. We suspected that iconics produced by the non-preferred hand might appear with dependent clauses, passives, and information not central to the narrative; while the preferred hand would perform iconics accompanying independent clauses, active verbs and statements about important events in the story. Again we uncovered no systematic variations.

Two major findings thus emerge from our observations of the production of iconics and beats. First, in dextrals, preferential gesturing with the right hand consistently occurs for iconics, which are very closely associated with speech content, but not for beats, which bear no formal relation to what is being said. This result is consistent with the finding of Sousa-Poza et al. (1979), that 25 of 28 right-handed males displayed a right hand preference in producing "representational" gestures, but no asymmetry for "non-representational" ones. Since iconic gestures, as mentioned previously, involve motor sequences, whereas beats are discrete movements, it is possible that the dominant hand performs more iconics than the non-dominant, simply because it possesses greater motor skill; but a contribution of speech laterality cannot be ruled out on the basis of these data.

Second, for our sinistrals, strength of hand preference on other tasks correlates with hand asymmetry in the production of iconics but not of beats. The fact that many of Kimura's strong left-handers exhibited no gesture hand preference is impossible to evaluate without knowledge of what types of gestures her subjects produced.

We are now conducting an experiment to determine the strength of association between each of several indices of handedness as well as language dominance, and hand preference in the production of iconics and beats. To elicit large numbers of both types of gesture, we require each subject to view a feature-length film, which he then narrates to a listener who has not seen it. Two measures of dominance for receptive language function--a reading test developed by Levy and Reid, and a dichotic listening task--are administered to the narrators.

Unfortunately, we know of no non-intrusive measure of the lateralization of expressive speech. For most right-handers we can safely assume that the left hemisphere is dominant, and has primary control of the right hand. However, we cannot make the same assumptions concerning either hemisphere in sinistrals. Levy and Reid suggested that left-handers who write with an inverted posture (with the hand above the line of writing) control fine movements of the writing hand via ipsilateral motor pathways (p. 136). Smith and Moscovitch (1979) found some support for this theory, but it has not been established as fact. Therefore, we cannot say which hemisphere controls the preferred gesture hand in a left inverter.

Despite these unresolved issues, we can ascertain which hand probably is controlled by the speech dominant hemisphere in the performance of at least some activities. Numerous researchers have found that if a right-handed subject is required to tap a key with one finger or hand, in isolation and concurrently with speaking, the right hand, but not the left, shows a decrement in tapping rate when the subject is speaking (e.g., Kinsbourne & Cook, 1971; Lomas & Kimura, 1976; McFarland & Ashton, 1975; see Kinsbourne & Hicks, 1978, for a review). Kinsbourne and Hicks (1978)

interpreted this result to indicate that the speech center or a nearby area also controls the right hand in its performance of the manual activity, and when a limited area subserves two competing functions, a decrement will be observed in the performance of at least one activity. In our study, subjects are required to tap silently and when reading aloud for comprehension. Hellige and Longstreth (1981) found that for dextrals, reading concurrent with unimanual tapping produces a greater decrement in right hand than in left hand tapping rate, and that the maximal rate reduction occurs when subjects read aloud with the expectation of a comprehension test afterward.

Finally, we assess the hand preference of each subject in the performance of a number of common tasks, and measure his skill on a peg-moving test which involves sequencing of hand and arm movements (Annett, 1970).

One observer will classify every gesture, and a second one will classify the gestures occurring during a brief segment of each filming session, so that reliability may be computed. Hand preference in the production of each type of gesture will be correlated with the indices of language dominance and general hand preference and skill. Results will be available by the time of the conference.

Table 1

Gesture hand predominance in relation to handedness and strength of general hand preference

Subject	Hand Pref. Strength	N Clauses in Nar.	Hand of Gesture: Iconic Left	Iconic Right	Iconic Both	Beat Left	Beat Right	Beat Both
Right-handers			N	N	N	N	N	N
C.	-	130	9	33	26	6	18	14
L.	-	98	4	21	24	22	1	17
S.	-	101	2	24	25	4	7	11
K.C.	9	133	3	23	20	0	1	5
E.O.	9	131	10	27	39	30	14	3
T.S.	7	222	21	75	41	0	32	0
D.	-	90	4	14	6	21	2	5
M.N.	9	126	1	26	1	1	4	1
D.H.	9	191.	3	77	4	0	27	1
V.F.	9	71	16	3	8	19	5	5
Left-handers								
V.G.	-6	141	9	37	10	5	6	3
M.V.	-3	114	9	22	10	0	6	5
S.M.	-1	92	7	27	25	12	7	12
V.	-	149	6	13	27	8	6	17
D.R.	3	228	10	23	57	3	3	8
J.B.	-5	175	26	24	33	4	2	6
A.B.	0	115	14	4	25	6	1	10
D.S.	-9	62	15	2	10	3	2	10
D.G.	-9	202	49	27	18	31	27	9
C.C.	-3	114	28	21	18	7	0	2
J.	-	134	21	11	15	41	3	7
K.C.	-5	130	11	6	11	4	0	5
E.B.	-3	86	11	9	4	8	1	2

References

Annett, M. The growth of manual preference and speed. British Journal of Psychology, 1970, 61, 545-558.

Hecaen, H. and M. Piercy. Paroxysmal dysphasia and the problem of cerebral dominance. Journal of Neurology, Neurosurgery and Psychiatry, 1956, 19, 194-201.

Hellige, J. B. and L. E. Longstreth, Effects of concurrent hemisphere-specific activity on unimanual tapping rate. Neuropsychologia, 1981, 19, 395-405.

Kimura, D. Manual activity during speaking--I. Right-handers. Neuropsychologia, 1973, 11, 45-50.

Kimura, D. Manual activity during speaking--II. Left-handers. Neuropsychologia, 1973, 11, 51-55.

Kimura, D. and Y. Archibald. Motor functions of the left hemisphere. Brain, 1974, 97, 337-350.

Kinsbourne, M. and J. Cook. Generalized and lateralized effects of concurrent verbalization on a unimanual skill. Quarterly Journal of Experimental Psychology, 1971, 23, 341-345.

Kinsbourne, M. and R. E. Hicks. Mapping cerebral functional space: competition and collaboration in human performance. In M. Kinsbourne (Ed.), Asymmetrical function of the brain. Cambridge, England: Cambridge University Press, 1978, 267-273.

Levy, J. and M. Reid. Variations in cerebral organization as a function of handedness, hand posture in writing, and sex. Journal of Experimental Psychology--General, 1978, 107, 119-144.

Lomas, J. and D. Kimura. Intrahemispheric interaction between speaking and sequential manual activity. Neuropsychologia, 1976, 14, 23-33.

Marcie, P. Writing disorders in 47 left-handed patients with unilateral cerebral lesions. International Journal of Mental Health, 1972, 3, 30-37.

McFarland, K.A. and R. Ashton. The lateralized effects of concurrent cognitive activity on a unimanual skill. Cortex, 1975, 11, 283-290.

McNeill, D. and E. Levy. Conceptual representations in language activity and gesture. In R. Jarvella and W. Klein (Eds.), Language and place. London: Wiley, 1982.

Milner, B., C. Branch and Th. Rasmussen. Observations on cerebral dominance. In A. V. S. de Rueck and M. O'Conner (Eds.), Ciba Foundation Symposium on Disorders of Language. London: Churchill, 1964.

Smith, L. C. and M. Moscovitch. Writing posture, hemispheric control of movement and cerebral dominance in individuals with inverted and noninverted hand postures during writing. Neuropsychologia, 1979, 17, 637-644.

Sousa-Poza, J. F., R. Rohrberg and A. Mercure. Effects of type of information (abstract-concrete) and field dependence on asymmetry of hand movements during speech. Perceptual and Motor Skills, 1979, 48, 1323-1330.

KNOWLEDGE CONSTRAINTS AND LANGUAGE COMPREHENSION IN APHASIA

Victor Rosenthal*, Patrizia Bisiacchi** and Evelyne Andreewsky***

*U.E.R. de Psychologie, Université Paris VIII, Saint-Denis, France.
** Istituto di Psicologia, Università di Padova, Italy.
*** I.N.S.E.R.M. U-84, Hôpital de la Salpêtrière, Paris, France.

Introduction

You are walking in the street and hear a sentence "Paul didn't want...". As you neither know who is Paul nor the person talking, you can hardly grasp the problem in its complexity. Yet, you have sufficient metapsychological knowledge on wanting, human relations, etc... to have some idea about the meaning of the sentence. That is, instead of explicit relevant knowledge, you have normally sufficient tacit knowledge to fulfill the *minimal requirements of understanding*.

The controversy regarding the ubiquity of the penetrations of knowledge into mental functions continues to flourish in cognitive psychology (see for instance Pylyshyn, 1981). The question might be crucial to some extent, for we are all intuitively tempted to believe that words have mentally encoded independent meanings that are reactivated on each occurrence of a word - and we sometimes have an impression of being able to undergo a linguistic, knowledge-independent comprehension. The trouble is that, in normal conditions, the use of tacit knowledge in the meaning-making acts is so indissociable from knowledge-independent contributions that it is impenetrable to our insights, and, until present, unisolatable in experimental designs. The findings from psycholinguistic laboratories that were believed to provide evidence for the consulting participation of knowledge in the act of understanding (e.g. the findings on inferential intrusions) could always be interpreted either as compatible with an alternative hypothesis of post-understanding facultative contributions or as limited to the particular experimental settings from which these findings have arisen.

In the present discussion we shall consider a two-stage model of understanding language in which we assume that related pre-existing knowledge is necessarily consulted. Our arguments will be mostly based on findings stemming from studies on language comprehension in aphasia. The most salient characteristic of aphasic disorders is a deficit (resulting from a brain damage) in the expression and comprehension of language. Such a deficit is not equivalent to a uniform decrease of linguistic performance: often aphasic patients suffer from a discrete impairment of a functionally distinct part of the language mechanism. As Saffran (1982) stated: "It is not unusual to find that some aspects of language function have been severely disrupted, while others remain relatively intact. (...) When subsystems that normally operate in concert break down independently, it becomes possible to investigate the residual systems in isolation. In some cases, the investigator can exploit specific functional deficits to control processes that may be difficult to manipulate under normal conditions". The analysis of these selective disturbances could lead us to identify the aspects of language that are subserved by functionally distinct mechanisms. Aphasic disorders might stand, therefore, for some sort of natural "pseudo-experimentation" as they allow us to observe functional dissociations in the language mechanism which are unconceivable in the psychological laboratories. With reference to our topic, cognitive neuropsychology offers the possibility of dissociating tacit knowledge contributions from knowledge-independent contributions to the understanding of language. We shall refer henceforth to any contributions of knowledge by using the broader term - *knowledge constraints*.

Two stages of comprehension: from pre-understanding to mental scenario

In terms of naive rationalism there is a simple correspondence between words and sentence meanings. The meaning of a sentence is a function of particular meanings of words and their structural arrangements. However, it is easy to demonstrate that there is no direct lexical basis for interpreting a sentence of the kind "can you give me the salt" (see Deloche and Andreewsky, 1981), and we all know simple examples showing that the meaning of a word can considerably differ as a function of the context in which the word is used (see Bransford and McCarrell, 1974). Winograd (1980) calls this paradox - the *hermeneutic circle:* you have to understand words in order to understand a sentence but in order to understand words you must understand the sentence. The hermeneutic circle is intrinsically linked to lexical-semantic approach to comprehension. *As long as you believe that words have mentally encoded independent and stationary meanings, and the meaning of a sentence is a combination of particular lexical meanings, the hermeneutic circle may prevent you from accessing any further understanding of comprehension.*

It seems worthwhile to distinguish two stages in the process of understanding (but we make no claim as to exclusiveness of these two stages). The first stage involves an introductory pre-processing of a sentence (see also Flores and Winograd, 1981). This pre-processing appears to be twofold:

- structural analysis of a sentence is done. This analysis leads the system to detect and syntactically disambiguate the key-words of a sentence, and globally to extract structural-relational information concerning the actual "state of affairs".
- as a consequence of detecting key-words, related knowledge constraints can be selected. The selection of knowledge constraints entails pre-understanding. (But note that,according to the present approach, words are considered only as abstract clues guiding the selection process).

The second stage of processing leads to a mental representation of the sentence content. This representation may be conceptualized as a scenario that you put on your mental stage. Here the information is no longer linguistic (nor semantic), rather a mental scenario represents *events* or *situations* described in sentences and constrained by your knowledge. Two complementary procedures appear to be involved in creating and staging a mental scenario.

The selection of related knowledge constraints allows the system to release appropriate knowledge-based routines which can promptly structure a scenario of the event. Their main advantage lies in the fact that they allow systematic processing of every item of information to be avoided. This reduces the processing load on the cognitive system, and, as a consequence, increases its capacity. Routines based on knowledge constraints cannot, however, supplant systematic processing of actual and specific aspects of situations. Casting actors for the parts they really play in an event (e.g. agent, recipient), situating an event in time and space (e.g. past, present,future,precedence, simultaneity), setting up each relevant relation (on time, space, causality, instrumentality), all this requires systematic processing (based in part on structural-relational information stemming from the pre-processing stage) that follows strictly determined rules (see Rosenthal and Bisiacchi, 1982). In short, systematic processing is responsible for the precise and actual "state of affairs" and assumes the role of cognitive controls preventing from an over-application of knowledge. These controls can sometimes be ineffective, as in the case of some common misunderstandings or as in certain artificial experimental tasks yielding knowledge-based intrusions. For instance, if you present a subject with a list of sentences such as: "The woman slipped in the staircase" and then test him for the immediate recall, it is very likely that you will notice several reproductions of the sort :

"The woman fell in the staircase" (Rosenthal, 1981).

Two stages of comprehension in the light of neuropsychological investigations -evidence for pre-processing

Let us suppose the feasibility of limiting our comprehension to the outcome of the pre-processing stage. If presented with a sentence, we would have the impression of knowing something about the meaning of the sentence, but would be unable to spell it out accurately. This situation is reminiscent of two experimental findings.

In now classical experiments on subliminal perception (or pattern masking) of individual words, subjects are often found to be unable to report what they saw, but if they are presented subsequently with a list of possible lexical alternatives, they are either capable of recognizing the stimulus or able to point to a semantically related word. In some conditions, they produce errors which bear a striking relationship to the stimulus but little other similarity (e.g. "king" for *queen*, "red" for *yellow*; see Dixon, 1971). That is, the subliminal presentation of a word appears to last long enough for selecting a related knowledge constraint but to be too brief for retaining the morphological pattern of the word.

In language pathology, similar findings have been reported with respect to the cases of deep dyslexia. A deep dyslexic patient cannot read nonsense words and reads function words (prepositions, conjunctions, etc...) very poorly. The reading of content words appears to be better preserved with a clear superiority of concrete nouns over the abstract ones, but a patient often produces semantic errors like: "crocodile" instead of *aligator*, "church" instead of *cathedral* (see Marshall and Newcombe, 1966; Coltheart et al., 1980). In the last few years, several cases of the auditory analogue of deep dyslexia have been discovered (Goldblum, 1979). In repeating words, a deep dysphasic patient performs in a way directly comparable to the way a deep dyslexic performs in reading. It has been noted that, in such a patient, the probability of producing semantic errors is inversely related to the typicality of a word (Goldblum, personal communication). Clearly, these patients are impaired in the ability to retain the perceptual (visual or auditory) pattern of a word but are able to perform the pre-processing leading to the selection of a related knowledge constraint. Accessing knowledge affords pre-understanding, but, since the lexical form is no longer available, a patient asked to reproduce the word would have no choice but to re-create it. Hence the factors such as abstractness, typicality, or number of synonyms should be relatively accurate predictors of the subject's performance.

Evidence for structural pre-processing arises from a study by Andreewsky and Seron (1975). They examined the ability of an agrammatic patient to read sentences aloud. The word *car* in French can be either a noun or a conjunction. The patient presented with the sentence: "Le car ralentit car le moteur chauffe" (The bus slows down because the motor overheats) read "car ralentit moteur chauffe". That is, he was clearly able to utter *car* since he produced the first *car*. In addition, when the second *car* (conjunction) was replaced by an unambiguous noun which he was able to read few minutes before, the patient read the sentence as in the example above. Implicitly, his selective ability to read words was determined by a structural analysis of the sentence. In general, studies on agrammatic patients force us to distinguish the ability to perform syntactic analyses of a sentence and the ability to use some of this structural information as clues for understanding (see Saffran, 1982). In terms of the above-described model, this distinction covers the structural pre-processing and the application of systematic processes during the staging of a mental scenario.

- evidence for knowledge-based routines and systematic processes.

We have seen in the preceding section that agrammatic aphasics are able to perform structural pre-processing and to access related knowledge constraints. It is our impression that their impairment has to be attributed to the representational stage, that is - agrammatic

patients preserve the capacity of using knowledge-based routines but often cannot perform systematic processing (Rosenthal and Bisiacchi, 1982). Hence their comprehension is more related to their knowledge of the world than to the actual state of affairs. In matching sentences to pictures, agrammatic patients perform on the basis of the "standardness of situations" irrespective of the precise characteristics of the situation described (Caramazza and Zurif, 1976; Deloche and Seron, 1981). Provided with reversible sentences they assign roles to actors according to greater plausibility. When the roles are interchanged violating pragmatic habits (i.e. The patient takes care of the doctor) agrammatic aphasics apply a normative strategy inverting the S-O relation. On the other hand, presented with sentences unconstrained by the pragmatic knowledge (e.g. The circle is above the square) they perform on the chance level.

Posterior Wernicke's aphasics show an opposite tendency in comprehension. They are insensitive to the "standardness of situations" (often mismatching both sentences that describe odd events and those that describe highly plausible events; see Deloche and Seron, 1981) and inclined to over-rely on structural information (von Stockert, 1972). This suggests that posterior aphasics could be limited in their ability to use knowledge-based routines but retain the ability to apply systematic processes. It should be recalled that routines afford the possibility of avoiding systematic processing of every bit of information and thus increase the processing capacity of the system. If actually, posterior aphasics suffer from low availability of routines we may predict that their processing capacity should be overall reduced. We examined this prediction in an experiment using riddles composed of two descriptors. The information contained in *both* descriptors was necessary to identify the intended item. Posterior aphasics, provided with a multiple choice array, performed poorly on this task. Most of their errors were responses based on only one descriptor (Rosenthal and Bisiacchi, 1982).

* *
*

In short, the reported findings with aphasic subjects provide at least partial support for the two-stage model of language comprehension, and illustrate some possible contributions of cognitive neuropsychology to adjacent arts.

References

Andreewsky,E. and Seron, X. (1975) - Implicit processing of grammatical rules in a classical case of agrammatism. Cortex XI, 379-390.

Bransford, J., McCarrell, N. (1974) - A sketch of a cognitive approach to comprehension. in Weimer and Palermo (eds.): Cognition and the symbolic processes, Hillsdale: Erlbaum.

Caramazza, A., Zurif, E. (1976)- Dissociation of algorithmic and heuristic processes in language comprehension. Brain and Language, 3, 572-582.

Coltheart, M., Patterson, K., Marshall, J. (1980)- Deep dyslexia. Routledge, London.

Deloche, G., Andreewsky, E. (1981)- From neuropsychological data to reading. Draft

Deloche, G., Seron,X. (1981)- Sentence understanding and knowledge of the world. Brain and Language, 14, 57-69.

Dixon, N. (1971)- Subliminal perception. London: McGraw-Hill.

Flores, F., Winograd, T. (1981)- Understanding computers and cognition. Draft.

Goldblum, M.C. (1979)- Auditory analogue of deep dyslexia. in Hearing Mechanisms and Speech, Berlin, Springer.

Pylyshyn, Z. (1981)- The imagery debate: analogue media vs. tacit knowledge, Psych Rev. 88, 1, 16-45.

Rosenthal, V. (1981)- Contribution à l'étude des configurations sémantiques dans les activités de compréhension. Doctoral dissertation. Université Paris VIII.

Rosenthal, V., Bisiacchi, P. (1982)- Representing sentence content in aphasia. Psychologica Belgica, in press.

Saffran, E. (1982)- Neuropsychological approaches to the study of language. Brit J. of Psychology, in press.

von Stockert, T. (1972)- Recognition of syntactic structures in aphasic patients. Cortex, 8, 323-334.

Winograd, T. (1980)- What does it mean to understand language. Cognitive Science, 4, 209-241.

A Unified Theory of Cognitive Reference Frames

Michael Leyton

Department of Psychology
University of California, Berkeley

The term reference frame is used in a wide variety of studies to describe a remarkably diverse set of phenomena in the field of Human Cognition. No unified theory exists. This paper elaborates such a theory and applies it to a number of examples in the following main areas: (1) Categorization and Prototypicality (2)Visual Shape Perception; (3) Auditory Reference; (4)Motion Perception; (5)Linguistic Deixis.

Cognition as the modeling of logical processes

In a lengthy study of perceptual organization (Leyton, 1974), I concluded that perception is an attempt to represent the world as a set of logical languages. Any such language consists of four components (S,F,A,P)

S= a set of primitive symbols
F= a set of rules of formation
A= a set of axioms
P= a set of rules of procedure

Essentially, the rules P are applied to the axioms A to produce a further set of formations which are called theorems. I argued that perception attempts to distinguish in any environment an axiom set of stimulus formations and derive the other stimuli as theorems generated from A, via perceptual operations P. I argued further that because a logical language is equivalent to a machine (Minsky, 1972), perception is inherently an attempt to give a machine-like (or computational) account of the environment. Because it seemed to me that perception, as a descriptive mechanism, exhibited, in the above respects, general properties of all descriptive processes, I argued that all information or description is inherently a computational account.

Although my argument in Leyton (1974) used purely cognitive evidence, in Leyton (1981a&b),I arrived at the same conclusion using theoretical-biological and statistical-mechanical arguments: Perceptual mechanisms were developed to identify, in the environment, machines to which the organism could couple itself to extract work. Thus, in claiming that all perception is the description of machines (or computational processes), I was claiming that all perception is inherently the identification of available work. The present paper elaborates this view further and shows how it explains cognitive reference phenomena.

Machines as the basis of description.

Essentially, any machine M (a state-output machine) can be described as

$$M = \begin{cases} Q = \text{a set of states (i.e. a state-space)} \\ P = \text{a set of inputs} \\ \text{an action of the input set on the states} \end{cases}$$

The inputs can thus be viewed as transformations causing state-transitions.

I claim that all description (including perception) is an attempt to characterize classes of stimuli as state-spaces of machines. Thus, in particular, I argue that the properties of any single stimulus are split into two classes (1) those properties denoting state, (2) those denoting the object which is undergoing the state (e.g. a falling rock). Thus we have:

Preliminary definition A description of a stimulus set S, is a map from the state-space of a machine onto S; that is, a map

$$D: Q \longrightarrow (S, \emptyset)$$

for some machine $M=(Q,P)$.

(The empty set $\emptyset$ is included because S might not yeild the entire state-space).

Example Consider a hexagon. There are 12 transformations (rotations and reflections) which map it to itself: e, r_{60}, r_{120}, r_{180}, r_{240}, r_{300}, t, tr_{60}, tr_{120}, tr_{180}, tr_{240}, tr_{300}.

where e = no transformation (the identity map)
r_n = rotation by n degrees
t = reflection

In the above view of description,(1) the sides are perceived as the states of a machine, and (2) the state-transitions therefore become the above 12 transformations. All twelve transitions map any one side to some other side, or to itself. The resulting diagram is exactly the state-transition diagram of the associated finite-state machine. For clarity, Fig 1 presents only a part of the diagram. Most of the 72 perceived connections are omitted.

The meaning of reference.

I claim that a viable unified theory of reference frames is obtained if one assumes

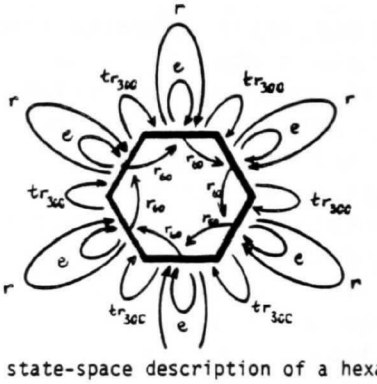


Fig 1. A state-space description of a hexagon

that the brain identifies certain states as initial ones; that is. they are viewed as pre-input or axiomatic. The important result is: Because all other states are then obtained by applying the input operations, P, each state is identifiable by the operation which produced it. Thus the machine description of a hexagon is reduced simply to viewing one edge, e.g. the top edge) as a starting point and viewing the others each as equivalent to only the operations which obtained them from the top. In consequence, the other sides are referred back

to the top one (Fig 2).(I proposed this view of reference, in mathematical-logical terms, in Leyton, 1974).

We therefore have a revised version of what a description is. It is a map from the inputs (or state-transitions) onto the stimulus set. Thus the individual stimuli are described as follows: 'this stimulus is what I obtained after I applied such and such an act to the initial one'.

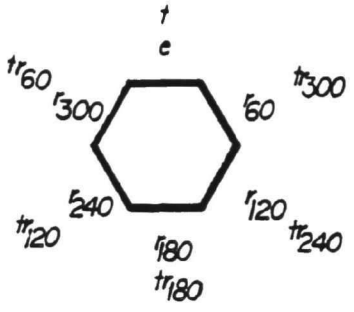


Fig 2. An input description of a hexagon

Group or input descriptions.

The system of state-transitions (or inputs) of a machine obeys a set of conditions which define it to be what mathematicians call a semigroup. We will assume the existence of an extra condition (each input has an inverse input) which makes the system what is called a group. The assumption is psychologically important because it allows the object/state splitting of the stimulus properties.

Thus the input set can be viewed as a group of state-transition functions, or an input group acting on S. But our theory of reference states that all description is the identification of stimuli with members of the input group. Thus we argue that all description is of this form:

Definition: A description of a stimulus set S is the map of the input group G, of a machine,onto S; in fact the map

$$\Gamma: G \longrightarrow (S,\emptyset)$$

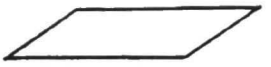
for some machine M=(Q,G)

Therefore, because reference acts with respect to the pre-input or non-transformed state, it acts here with respect to the non-transformation element (the 'identity' element) which every group contains.

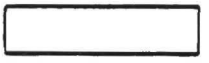
The structure of reference.

In the usual reference situation, the state-space is multidimensional; that is, it is the product of several one-dimensional component groups. In this case reference acts not just with respect to the identity element of the entire group but with respect to the identity elements of each of the 1-dimensional component groups. In fact, I found (Leyton, in preparation) that reference acts successively across the components. For example: a rotated parallelogram

which is referred to a non-rotated one:



which is referred to a straightened one, i.e. a rectangle:



which is referred to the non-elongated version i.e. a square:

In the reference process above, the mind first eliminates the group of rotations,i.e. refers back to the identity of the rotation group, then it eliminates the group of shears, i.e. refers back to the identity of the shear group, and finally eliminates the group of elongations i.e. refers back to the identity of the elongation group. In fact, I have shown (Leyton, 1982: Leyton, in preparation) that the ordering in which elimination occurs is that of the perceived increasing stability of the successive group dimensions i.e. inputs. The above rotation is perceived as less stable than the shear, which is perceived as less stable than the elongation

We thus conclude:

Reference involves the mapping of an input group of a machine to a stimulus set such that the members of the set become viewed as a generated space of states, identifiable with th inputs that obtained them. The reference process successively factors out the 1-dimensional component groups (or machines) in order corresponding to their increasing perceived stability. The reference point in each dimension is the group identity element (i.e. giving the pre-input state).

APPLICATIONS

1. Protypicality and reference.

Rosch (1975) has proposed that natural categories - such as colors, line-orientations and numbers - have reference point stimuli - such as focal colors, vertical and horizontal lines,and number multiples of 10 - with respect to which other catergory members are judged. For example, pink is referenced to red, a leaning object to the vertical, and 99 to the number 100. The reverse references do not happen.

Using the above theory of descriptions, I claim that :

A prototype is a stimulus which is labeled by the identity element of the associated input group.

It is for this reason, for example, that a giraffe is judged as an animal with a long neck, whereas the neck of a more prototypical animal, such as a dog, is not even mentioned. In our theory, the giraffe is viewed as needing a transformation to be obtained (in fact being equivalent to that transformation) whereas a dog is not, i.e. the dog is at the initial (axiomatic, pre-input) state of the

associated dynamical system. Again 99 is obtained by moving 1 down from 100 (i.e. applying the subtraction transformation) whereas 100 is obtained by 'just staying there'.

2. Shape perception

2.1 Shape and orientation

As is now well documented, the perception of shape depends on the assigment of orientation (Rock, 1973) A famous example (Fig 3) is the perceived difference between a square and a diamond, which depends on how the perceiver places a reference coordinate system over the same underlying figure.

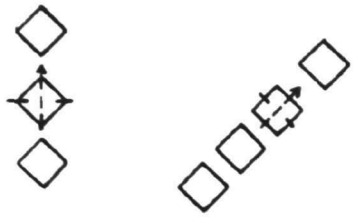


Fig 3. Assigned direction effecting perceived shape.

Leyton (1974, 1978) and Palmer (1981) have independently proposed a theory of shape perception, in terms of the internal symmetry transformations. However, while their view accounts for several important effects, it is clear that it does not account for the effects of orientation on form perception. I claim that the present view does, because it maps the input group directly down onto the stimuli, thus identifying the stimuli totally by the transformations (i.e. inputs) which obtain them. (Note that internal symmetries allow a range of alternative symmetrically related descriptions which do not violate interpretation.) Thus a definite element (or range of elements) has necessarily to be identified as the starting point of the associated machine. Furthermore, specific subsets have definitely to be identified with specific component 1-dimensional groups. A change of interpretation of a figure then becomes an alteration in the elements which perception allows to be labeled by the identity input, or an alteration in the subsets which receive the component groups, or a total change of group. For example,the main perceived axial structure of a square implies that it is interpretable as generated foom initial parts such as those in Fig 4.

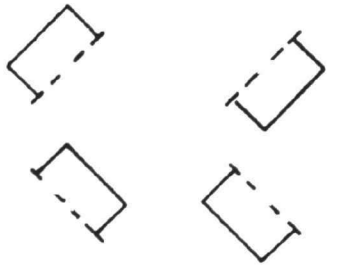


Fig 4 Allowable generators of a square.

However, the main perceived axes of a diamond imply that amongst the allowable generators are the stimuli in Fig 5. Thus interpretations change with the set of allowable generators.

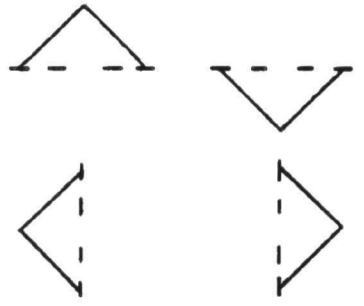


Fig 5 Allowable generators of a diamond.

2.2 What is shape?

A shape is an interaction between two state spaces: its internal state space (e.g. the input description of the hexagon, given in Fig 2) and its external state space; i.e. what the figure can do (e.g. rotate). We have seen that a square and a diamond are distinguished by the mappings of their internal input groups. However, I claim that they are distinguished also between their external input groups. For a square, the more stable input group includes the state transitions in Fig 6. However, for a diamond, it includes more stably the state transitions in Fig 7. Squashing across the corners is not allowed stably for the square. I have identified (Leyton, In preparation) that an important aspect of the interaction of the internal (symmetry) state space and the external one is that the axes of symmetry in the former become identified as the axes of flexibility of the latter (i.e. become the 1-dimensional component groups in the latter).

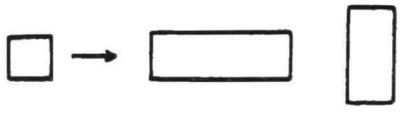


Fig 6 Allowable external inputs of the square

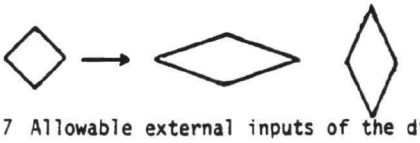


Fig 7 Allowable external inputs of the diamond

In their external descriptions, figures are also clearly identified as particular members of a state transition group, because reference also exists with respect to the initial point. For example, Wiser (1981) found that even if objects such as that in Fig 8 were presented in a non-vertical orientation, they were nevertheless recognized faster when presented again in the vertical orientation than in the initial one. Thus her results show that (1) the stimulus properties are clearly partitioned into those denoting state and those denoting the object

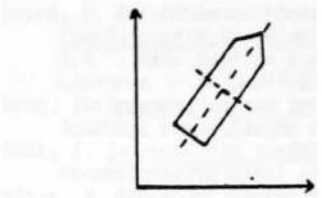


Fig 8 From Wiser (1981)

undergoing the state and (2) that the state is in fact identified as a transformation with respect to a referent initial state.

2.3 Pattern goodness

The relation of goodness to reference takes two important forms: Type 1, where a pattern such as Fig 9 is judged as less good, and is referenced to, its completed version; and Type 2, where a pattern such as Fig 10 is judged as less good, and is referenced to its non-deformed version, a square.

Fig 9

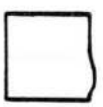


Fig 10

Our theory explains the two phenomena thus: Type 1: The goodness rating in Fig 9 is clearly based on the fact that the sides are perceived as needing more input transformations. That is, the entire machine has not been given. That is, a larger set of internal inputs is assumed for the figure. Thus, pattern goodness, in this case, is evaluated by the ratio

$$\frac{|\Gamma^{-1}(S)|}{|G|}$$

i.e. the proportion of the internal input group G used in the description map, Γ. Type 2: The goodness rating in Fig 10 is clearly based on the positioning of the figure in an external space of inputs (i.e. of deformations) and referencing it to an identity or pre-input element (which we have shown, constitutes the prototype).

We emphasize: Type 1 goodness verifies our postulation of an internal input group, and Type 2 goodness verifies our postulation of an external input group.

2.4 The Marr/Nishihara Shape Description Theory.

Marr and Nishihara (1978) claimed that the perceptual description of shape (e.g. the shape of animals) is given by viewing the figure as a concatenation of approximately cylindrical modules (Fig 11) with specific relative widths and lengths (Fig 12). These are obtained by assigning a collection of object-centered local reference frames (axes) to the parts of the stimulus configuration. The relationship between the frames is given by the coordinate system (p,r,θ,i,ø,s) where symbols are as shown in Fig 13. By applying our theory, we see that each of the figures in the Marr/Nishihara paper describes one of the points in an input space. The generation of a module (Fig 11) by translating a circle through space along an axis and by varying the diameter is the perception of external inputs to the circle. (Note that they become internal inputs of the module). The relative position of one module to another, as described by their coordinate system (Fig 13), is clearly a state -space, where the module positioning is essentially a state under the associated group of transformations along these parameters e.g. lowering arms lengthing legs, waving

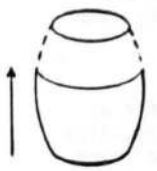


Fig 11 Generating a vase.

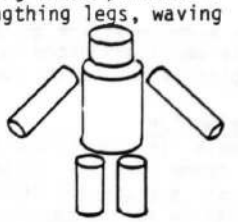


Fig 12 An ape.

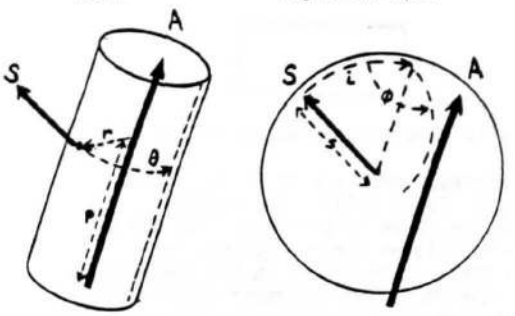


Fig 13 The Marr/Nishihara coordinates for relating two modules (After Marr & Nishihara, 1978).

the hand, nodding the head, etc). Thus the figure is a point in the multidimensional group input space described by the interactions and shapes of the modules. The reference points in this space would be the prototypical animals and prototypical positions identified by the theory and techniques of Rosch (1978). Recall also that we claimed that an important interaction between the internal and external groups is that the invariant axes of the former become the component groups (directions of action) of the latter. This is clearly evidenced in the Marr/Nishihara description: the central axis of a module i.e. the invariant line under internal rotation of the module, becomes the direction along which it can be stretched.

3. Audition

3. Autitory Streams

Auditory input, e.g. a rapid sequence of tones is segregated perceptually into what Bregman and Campbell(1971) call, 'primary auditory streams'. These streams are groupings or frames and any tone can be allocated to only one of them.

Our theory of the situation is as follows: Bregman (1981) himself argued that an auditory stream corresponds to the object in visual perception. Leyton (1974) described the group-theoretic and logical language structure of music. In particular, he showed that musical transposition (change) of pitch is modeled by a group. This group allows the tones of a melody to be perceived as a single tone (object) being moved into different states under an input group. Therefore, the segregation of auditory stimuli into streams is, in our view, the description of the latter as a disjoint set of machines.

3.2 Musical Reference to the tonic.

If my hypothesis is correct that a stimulus becomes identified not just as a

state of an object but as the operation (in an input group) which achieves that state, then there must be a stimulus which is labeled as the identity element of the group. This conjecture is amply evidenced by music: the reference point is called the tonic.

4. Relative motion

An example is the following: When a rectangular frame (Fig 14) is moved relative to an observer, and a point inside the frame is fixed relative to the observer, the point

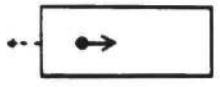


Fig 14 The induced motion effect

is nevertheless perceived to be the moving object (e.g. Rock, 1975). Our theory describes the above in this way: the set of possible velocities clearly defines the relevant state space (which is two-dimensional). However, the reference judgement enters when one identifies each velocity with the transformation which obtains it from the zero velocity, i.e. it is perceived as an increase (or decrease) of speed by a certain amount. This allows it to be referenced back to the '0' or identity element of the input group. The latter element is then assigned to perceptually the most stable object in the field, i.e. the rectangular frame.

5. Linguistic deixis

Deixis (Bühler, 1934) is a term used to denote those linguistic aspects which locate or point to the object of speech; e.g. 'here', 'there', 'this', 'that', 'then'. Bühler claimed that these aspects create a coordinate system, centered on the referent (Fig 15).

Fig 15 The deictic field

The theory, which I have proposed, appears to model Buhler's concept. The deictic field clearly is a dynamical view of the space centered at the origin. "Put the book in front of my chair" means "One can find the place to put the book by inputing a translation forward from my chair's location". Thus, the coordinate system (Fig 15) is - as I belive all coordinate systems are - labeled by the internal inputs (i.e. transformations) which move location with respect to the origin and axes. When an individual gives the pointing gesture, 'there', he is literally translating the deictic input group from himself to another point, such that the axes are properly aligned. As with the gravitational frame, these axes are representations of the 1-dimensional component groups of the internal input group; i.e. they give discrete labels for movement, not for physical packets of stimuli.

General conclusions.

The above presents a large-scale view of cognition. The view is corroborated by the several examples considered. In particular, the examples confirm the following principles

(1) Cognition is the attempt to model the environment as a union of machines.
(2) A reference frame as a machine with initial conditions defined.
(3) Referencing a stimulus is the process of
  (i) deciding on an object/state split of its properties
  and(ii)identifying the state properties with the input needed to obtain the stimulus from the initial conditions of the associated machine

The substantiation of this view of reference corroborates also our proposal that description is a mapping of an input group of a machine, onto a stimulus set.

References

Bregman, A.S.Asking the "What For" Question in Auditory Perception. In: Perceptual Organization, M.Kubovy &J.R. Pomerantz (Eds).Hillsdale, N.J.: Lawrence Erlbaum 1981

Bregman, A.S. & Campbell, J. Primry auditory stream segregation and perception of order in rapid sequence of tones. J. Exp. Psychol 1971, 89, 244-249.

Buhler, K. Sprachtheorie: die Darstellungsfunktion der Sprache. Jena: Gustav Fischer, 1934.

Leyton, M. Principles of Artistic Method: Algebraico-Logical Postulates in the Foundations of the Science of Perception. Research Report, Mathematics Institute, University of Warwick, Coventry, England 1974*

Leyton, M. Artistic Structure: Group-Theoretic, Differential-Geometric, and Mathematical-Logical Factors in Perception. Research Report, Mathematics Institute, University of Warwick, Coventry, England, 1978*

Leyton, M. Artistic Activity and Human Survival: Volume 1. Unpublished book. 1981a*

Leyton, M. Do Structural and Statistical Evaluations of Information Vary Inversely or Directly. Research report. 1981b*

Leyton, M. Description, categorization, and reference - a unified theory. Seminar given at the Department of Psychology University of California, Berkeley, March, 1982*

Marr, D. & Nishihara, H.K. Representation and recognition of the spatial organization of three-dimensional shapes. Proceedings of the Royal Society of London. 1978, B200, 169-294.

Minsky, M. Computation: Finite and Infinite Machines. Englewood Cliffs NJ: Prentice Hall, 1972.

Palmer, S.E. Transformational structure and perceptual organization. Proceedings of the 3rd annual conference of the Cognitive Science Society. 1981.

Rosch, E. Cognitive reference points. Cog. Psychol. 1975, 7, 532-547.

Rosch, E. Principles of Categorization. In: Cognition and Categorization, E. Rosch & B.B. Lloyd (Eds) Hillsdale, NJ: Lawrence Erlbaum. 1978

Rock, I. Orientation and Form. New York: Academic Press, 1973.

Rock, I. Introduction to Perception. New York: Macmillan, 1975.

Wiser, M. The role of intrinsic axes in shape recognition. Proceedings of the 3rd annual conference of the Cognitive Science Society. 1981.

*Copies available from:
Michael Leyton,
Department of Psychology,
University of California,
Berkeley, CA 94720.

KNOWING, UNDERSTANDING, AND BELIEVING

Yutaka Sayeki
University of Tokyo

1. Learning and Knowing How

"Learning" in the ordinary sense simply implies the acquisition of knowledge, or the change in the state of knowledge. However, psychologists have been afraid of being asked by sceptics, "How do you know that your subject has changed his or her state of knowledge?" Their avowed answer follows: "From the subject's behavior may we infer his or her state of knowledge." Thus Bower and Hilgard (1981) define:

> "Learning refers to the change in a subject's behavior or behavior potential to a given situation brought about by ....." (p.11)

However, if we stick with our ordinary notion of learning, then "X learned" simply implies that X has come to know something.

But then we must face with a fundamental problem in epistemology on the distinction between knowing how and knowing that. This distinction has been introduced by Winograd (1975), and Rumelhart and Norman (1981) in relation to the controversy on the representation of knowledge, i.e., procedural vs. declarative representations. However, the original distinction between knowing how and knowing that was on the nature of knowledge itself, rather than on its representation (Ryle, 1949). In other words, if we focus upon the kind of knowledge characterized by the subject's performance approaching to a certain criterion, then we are primarily concerned with subject's knowing how, rather than knowing that. On the other hand, if we focus upon the other kind of knowledge characterized by the subject's belief in the truth of a proposition, then we are concerned with his knowing that.

Although Ryle originary made the basic distinction, he was primarily concerned with knowing how. He specified the subject's intellectual disposition by his potential tendency of behavior to act properly and correctly under the given situation, not as a result of simple habit, but as a result of deliberate consideration. Thus the state of subject's knowledge that traditional psychologists have been concerned with seems to correspond to Ryle's definition of "knowing how" exclusively.

2. Understanding and Knowing That

The nature of "knowing that" has been extensively analyzed by Scheffler (1965). According to Scheffler, X knows that Q if and only if

(1) Belief condition: X believes that Q,
(2) Evidence condition: X has adequate evidence that Q,
(3) Truth condition: Q.

(Here, the third condition is purely epistemological, and will not be discussed in the present paper.)

Petrie (1965), independently of Scheffler's work, reached at almost the same conclusion in his analysis of "learning with understanding," to be distinguished from rote learning. He asked the question, "What is to learn a fact or a methodology with understanding?" Then he proposed first on learning proposition P with understanding such as: X learned with understanding that P if and only if

(P1) X has come to believe through experience that P,
(P2) X has good (justifying) reasons for believing that P,
(P3) P (the truth condition).

Then he examined if there is any sense in saying, "X learned methodology M with understanding." Obviously, there seems to be some factual learning about M, such as learning that the rules and principles underlying M are indeed valid and appropriate to attain a goal under given circumstances. In order to allege learning of M with understanding, learning of the principles seems to be requisite.

In addition to the learning of principles for M, Petrie requires that the reasons for believing these methodological principles should include not only inductive evidence that they do work, but also that they are only heuristic, i.e., there may be the better way to attain the same goal. The reason for this comes from the fact that methodology must always be improving.

Petrie's suggestions may be further elaborated as follows: If X learned M with understanding,

(M1) X has come to believe through experience that the basic procedures of M are appropriate under the given circumstance,

(M2) X has good (justifying) reasons for believing the appropriateness of the procedures,

(M3) X is trying to discover the better procedures by improving M.

Although conditions M1-M3 are necessary for learning M with understanding, they are by no means sufficient. It still remains true that one could learn all the facts about M without becoming an expert on M, that is, without learning how. In order to become a real expert, one must acquire the automatization of component skills to act smoothly. Although such automatization may occur without understanding, its formation helps people to obtain the deeper understanding ofthe basic principles than non-automatized learning of the principles, because of the proper encoding of chunks and the organization of the entire task. Moreover, the formation of automatization strenthens the understanding, because one would realize the appropreateness of the procedures together with the points to improve, through the excercise of the present methodology. Cross-cultural studies on cognition revealed that people's performances on reasoning and problem-solving are quite "domain specific," which may be interpreted as the outcome of such interactive effects between automatization and understanding (Cole and Scribner, 1974).

Recently, a number of authors (Anderson, et al., 1981; Greeno, 1980; Simon, 1980) attempted to clarify the concept of understanding in "meaningful" (instead of "rote") learning within the information-processing framework. They regard understanding as the proper use of higher order schema, representing the conceptual meaning in

declarative form, from which necessary procedures are derived to solve seemingly different, but conceptually the same problems. VanLehn and Brown (1980) proposed a model called "planning nets" for the knowledge about the purposes of every component of procedural skills, reflecting teleologic semantics. The concept of understanding in these and other studies in cognitive science clearly indicates the importance of Condition M2, the process of having good (justifying) reasons for the parts of procedural skills.

Condition M3, invention of new strategy through experience, has been extensively observed for learning arithmetics (Resnick, 1980). The process has been simulated by ACT production system (Anderson, et al., 1981). Adaptive production system (Anzai and Simon, 1979) also deals with natural development of skills through experience. Thus we may conclude that Conditions M2 and M3 are properly taken to account in cognitive science. Then, what about Condition M1?

Unfortunately, belief condition of "knowing" has been virtually ignored in the past studies on cognition (except for beliefs in interpersonal relations or political judgments, simulated by Colby, 1973, or Abelson, 1973). The condition is missing in the discussion of procedural knowledge, as well as semantic knowledge.

The treatment of semantic knowledge in cognitive science seems to have been close to Hartland-Swann's (1954) interpretation of "knowing that." He claimed that Ryle's "knowing that" should be interpreted as another kind of "knowing how," that is, "knowing how to answer correctly to the expected questions." This proposal was immediately criticized by Ammerman (1956) asking, "How do you know that your answer is indeed 'correct'?" One can produce "correct answers" without knowing their truthfulness.

3. When and How People Are Convinced

We all know that the results of logical reasoning, mathematical deduction, and statistical inference do not always convince ourselves. Tversky and Kahnemann (1974) demonstrated a variety of our "heuristic biases" in probabilistic judgments, differred from those prescribed by probability theory, i.e., availability, imaginability, and representativeness. Here, we may extend their notions to people's strategies to convince themselves or others of the truth of logical conclusion, physical descriptions, causal attribution, and the validities of procedural skills. We are easily convinced by being shown a "good example" (availability). An elaborated episode which stimulates our imagination often makes a plausible explanation (imaginability). We often cite proverbs and old sayings, insisting on the similarity to the "typical case" (representativeness). Obviously, we should not use these biased tendencies to believe, for convincing children of false propositions. However, some of them may be quite helpful in our classroom instruction to explain new subject matter, which is quite unfamiliar at the moment, but is to be examined rationally later. In classroom, however, experienced teachers adopt various strategies to convince children of the truth and validity of principles in subject matters. "Decomposition Strategy" breakes down the problem into familiar, manipulable, subproblems. "Reduction Strategy" reduces the problem into a simple case. "Transformation Strategy" transforms the problem into different views, keeping the essential part the same.

We are investigating why and how these strategies work (or do not work) in a variety of learning, convincing children the reality and truthfulness of the knowledge.

REFERENCES

Ableson, R. P. The structure of belief system. In Schank & Colby (eds.) Computer Models of Thought and Language, Freeman, 1973.

Ammerman, R. A note on 'Knowing that.' Analysis, 17, 30-32.

Anderson, J. R., Greeno, J. G., Kline, P. J., & Neves, D. M. Acquisition of problem-solving skill. In J. R. Anderson (ed.) Cognitive Skills and Their Acquisition, Erlbaum, 1981.

Anzai, Y. and Simon, H. A. The theory of learning by doing. Psychol. Rev., 1979, 86, 124-120.

Colby, K. M. Simulations of belief systems. In Schank & Colby (eds), Computer Models of Thought and Language, Freeman, 1973.

Greeno, J. G. Analysis of understanding in problem solving. In R. H. Kluwe & H. Spada (eds.), Developmental Models of Thinking, Academic Press, 1980.

Hartland-Swann, J. The logical satus of 'Knowing that'. Analysis, 1956, 16, 111-115.

Petrie, H. G. Rote learning and learning with understanding. Doctoral dissertation, Stanford University, 1965.

Resnick, L. R. The role of invention in the development of mathematical competence. In Kluwe & Spada (eds.), Developmental Models of Thinking, Academic Press, 1980.

Rumelhart, D. E., & Norman, D. A. Analogical processes in learning. In Anderson (ed.), Cognitive Skills and Their Acquisition, Erlbaum, 1981.

Ryle, G. The Concept of Mind. Hutchinson House, 1949.

Scheffler, I. Conditions of Knowledge. Scott, Foresman, 1965.

Simon, H. A. Information-processing explanations of understanding. In P. W. Jusczyk & R. M. Klein (eds.) The Nature of Thought, Erlbaum, 1980.

Tversky, A., & Kahnemann, D. Judgment under uncertainty. Science, 185, 1124-1131. 1974.

Winograd, T. Frame representations and the declarative-procedural controversy. In Bobrow & Collins (eds.), Representation and Understanding. Academic Press, 1975.

KNOWLEDGE AND BELIEF AS LOGICAL LEVELS OF REPRESENTATION

Gabriella Airenti[o], Bruno G. Bara[o], Marco Colombetti[+]

[o]Unità di ricerca di intelligenza artificiale, Università di Milano

[+]Progetto di intelligenza artificiale, Politecnico di Milano

We propose a representation system consisting of two interacting subsystems, named K-theory and K-model, which play the respective roles of conceptual and episodic knowledge. We define belief a model M used by thought processes not directly, but through a meta-structure which predicates a relation of M to other models. We claim that from an intrasystemic point of view the difference between knowledge and belief is determined neither by the structure and content of a model nor by its relation to objective truth, but by the logical level of its representation.

1. THEORETICAL FRAMEWORK

A number of different approaches to the distinction between knowledge and belief have been proposed in philosophical, AI and psychological literature. A first classification of such proposals is based on the distinction between:

- the aim of globally classifying a representation system as either a knowledge system or a belief system (Abelson, 1980);
- the aim of attributing the status of knowledge or belief to individual representational items within a system (Hintikka, 1962; Miller and Johnson-Laird, 1976).

A second classification, independent of the previous one, relies on the criteria used to assign the status of knowledge or belief to a system or to a single item. The main existing approaches are:

(i) an observer judges a representation system with respect to the objective reality; all representations are a priori considered as beliefs, and whenever a belief happens to be true it is considered as knowledge (Hintikka, 1962). For an AI application of such an approach see Cohen and Perrault (1979), and Perrault and Allen (1980);

(ii) an observer judges a representation system with respect to another representation system; see for example the "nontransparent" criteria by Abelson (1980): nonconsensuality and different in existence assumptions;

(iii) an observer judges a representation system on the basis of its structure; see for example the "transparent" criteria by Abelson (1980): presence of alternative worlds, of evaluative and affective components, of a substantial amount of episodic material, unboundedness, varying degress of certitude;

(iv) an observer judges a representation system S with respect to his own representations, assumed as knowledge. Whenever the representations of S agree with those of the observer, they are considered as knowledge; otherwise, they are regarded as beliefs. See for example the "deictic" definitions of KNOW and BELIEVE in Miller and Johnson-Laird (1976);

(v) a system judges his own representations. Abelson (1980) accounts for his case by emphasizing the possibility of "awareness of nonconsensuality". Instead, Miller and Johnson-Laird (1976) discuss the relation between KNOW and BELIEVE and the degree of dubiety of a representation.

Finally, note that the approach implicit in most AI representation systems is not to deal with the problem of beliefs, therefore considering all representations straightfowardly as knowledge.

From a psychological point of view, a human system can have access to external facts only through their internal representations. Therefore, the question becomes the ability of humans to subjectively assign to their own representations the status of knowledge or belief (Airenti, Bara, Colombetti, 1982). In Section 3 we shall argue that this distinction relies on the logical levels of representations.

2. THE REPRESENTATION OF KNOWLEDGE

We propose a knowledge representation system consisting of two interacting subsystems, named K-theory and K-model, playing the respective roles of conceptual and episodic knowledge (Airenti, Bara, Colombetti, 1980, 1981).

K-theory is a theory of the world, and can be conceived as a network of conceptual entities describing classes of objects, relations, processes, actions, etc. (for example: the concept of a book; of x being on y; of x falling from y; of an agent z opening y; etc.).

The cognitive system does not deal directly with the world, but with partial representations of it, which constitute what we call K-model. In fact, there is no way for K-theory to reference entities in an external world: in the cognitive system, representations only can be mentioned and used. K-model contains all episodic knowledge of the cognitive system, i.e. knowledge about particular objects, facts, episodes, etc. (for example: the book B Maria is now reading; the fact that B is presently on desk D; the fall of B from D; Maria's opening of B; etc.). These can only be expressed by means of the conceptual machinery provided by K-theory.

We assume that the essential feature of K-theory is the ability to generate models for insertion and subsequent manipulation in K-model. For example, the concept of a "book" is a structure in K-theory allowing the cognitive system to construct models of books whenever needed by a thought process. K-model contains the representation of the perceived world, which is continuously changing through time and space. As K-model is intended to capture the cognitive system's subjective experience, it does not only represent the perceived world, but also any possible imagined world. This corresponds to saying that any imagination process must produce data which belong to K-model, and thus satisfy K-theory. So K-theory determines the set of worlds

which are possible for the cognitive system, i.e. the spectrum of all its possible subjective experience. This leads to conceiving K-model as a set of models of K-theory, each representing parts of a possible external world - presently perceived or remembered or imagined.

The partition of knowledge into K-theory and K-model is logical rather than functional. This is reflected by the fact that K-model, as noted above, collects data used by different thought processes. Actually, all data involved by perception, imagination, illusions, dreams, plan formation and execution, language comprehension, etc., are introduced through different thought processes, but share the same logical structure. We emphasize that models are used by such thought processes as data; in these cases the system is not concerned with problems of existence of entities or truth of facts represented in a model.

3. THE LOGICAL STRUCTURE OF BELIEFS

On the basis of our previous definitions of K-theory and K-model, we assume a constructional standpoint. That is, the sole reality for the cognitive system is what is constructed by its thought processes using K-theory. It follows that the position which, according to Hintikka's approach, defines as knowledge a belief satisfied by the real world, cannot be applied. In a constructional approach, in fact, K-model necessarily satisfies the part of K-theory used to build it. For instance, if Margaret assumes in her K-theory that seawater is sweet (i.e. unsalty), in all models produced by her the sea will be sweet, regardless of the objective truth of this fact. From a subjective point of view it is appropriate to say that Margaret knows that the sea is sweet.

Now suppose that Margaret happens to taste seawater. Let us assume that Margaret is able to distinguish between sweet and salty water, and that her K-theory represents the two tastes as mutually exclusive when attributed to the same object. We suggest that the relevant possibilities in this case are:

(i) since the construction of the model of salty seawater would be conflicting with the previous models, either the new model is not constructed at all, or the new model is constructed but the discrepancy is not appreciated (in this case Margaret maintains her theory about the sweetness of seawater);

(ii) the model is reinterpreted (Margaret may think that her perception of a salty taste depends on a particular kind of salty rocks);

(iii) the discrepancy between the two inconsistent models is appreciated and faced by assigning to the discrepant models that status of beliefs (Margaret acknowledges the existence of two contradictory models).

Our hypothesis on case (iii) is that the coexistence in K-model of two contradictory models makes the thought processes unable to use them straightforwardly. To face this situation the system introduces in K-model a structure, which refers to the two models and represents the conflict between them (see Fig. 1). As such a structure predicates a relation between the two models, it can be considered at a meta-level with respect to them. It is by using this meta-structure that thought processes handle the conflict. The possible outcomes of such processes are: the old model is privileged and the new one is discarded; the reversed situation, i.e. the new model is privileged and the old one is discarded; a situation of uncertainty, where both models are maintained.

We define belief a model M used by thought processes not directly, but through a meta-structure which predicates a relation of M with other models. Such meta-structures are necessarily used by the system whenever two models cannot be interpreted simultaneously, i.e. cannot be predicated at the same time of the same thing.

This definition of the term "belief" seems to be the most appropriate within a subjective, constructional approach to the human mind. The difference between knowledge and belief is reduced to the different use that thought processes make of a representation, assumed either as absolute or as relative to other representations. Whenever the system does not directly manipulate a model M of the world, but reasons on it through a second level representation, M assumes the role of belief. Note that both structure and content of M remain the same when used as knowledge or as belief.

4. DISCUSSION

Referring to the first classification introduced in Section 1, we have discussed the problem of attributing the status of knowledge or belief to an individual representational item within a cognitive system. Many researchers who deal with this problem commit themselves on the assumption that the difference between knowledge and belief can be defined in terms of the objective truth of a fact. That human beings cannot have access to an ultimate, absolute truth is a trivial statement. As Miller and Johnson-Laird (1976) point out, it is not acceptable, either from a psychological or a linguistic point of view, to assume that "...knowledge is simply justified true belief and that one cannot be said to know something that is false".

From our psychological standpoint, we have therefore assumed a different position and focussed on the internal structure of knowledge and belief. We have shown that our definition of belief is significant in the case a system has to deal with conflicting models of the world. The idea of a second level structure seems not to be restricted to such a case, but it can be applied whenever the system evaluates properties of a model. Among these are the degree of certainty of facts and the existence in the world of the entities represented in a model. In fact, both existence and degree of certainty are not part of a model, but are predicated on it.

Our treatment of beliefs opens a problem about the thought processes manipulating K-model. The two possibilities are:

- thought processes treat in a uniform way both the models of the world and the meta-structures mentioning them; an analogous approach is taken by Wilensky (1981) in his work on planning and meta-planning;
- there exist a type of thought processes specialized in manipulating meta-structures; in this case the two levels of representation would reflect into two corresponding levels of thought.

REFERENCES

Abelson R.P., 1980. Differences between belief and knowledge systems, Cognitive Science Technical Report No. 1, Yale University.

Airenti G., Bara B.G., Colombetti M., 1980. A semantic memory model as a basis for a problem

solving system, Italian Journal of Psychology, VII, 2.

Airenti G., Bara B.G., Colombetti M., 1981. An artificial intelligence approach to the study of cognitive processes, URIA Internal Report, Università di Milano.

Airenti G., Bara B.G., Colombetti M., 1982. A two level model of knowledge and belief, in Trappl R., ed., Proceedings of the 6th European Meeting on Cybernetics and System Research, North Holland, Amsterdam (in press).

Cohen P.R., Perrault C.R., 1979. Elements of a plan based theory of speech acts, Cognitive Science, 3, 3.

Hintikka J., 1962. Knowledge and belief, Cornell University Press, Ithaca, N.Y.

Miller G., Johnson-Laird P.N., 1976, Language and perception, Cambridge University Press, Cambridge.

Perrault C.R., Allen J.F., 1980. A plan based theory of indirect speech acts, American Journal of Computational Linguistics, 6, 3-4.

Wilensky R., 1981. Meta-planning: representing and using knowledge about planning in problem solving and natural language understanding, Cognitive Science, 5, 3.

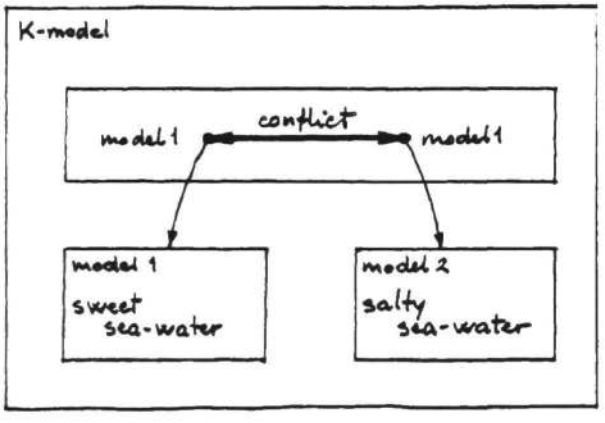


Figure 1. The meta-structure representing a conflict between two models.

ACKNOWLEDGMENT

This research has been supported by a grant for the year 1982 of the Consiglio Nazionale delle Ricerche, Comitato di Medicina, Gruppo di Scienze del Comportamento.

Representativeness Reconsidered

Maya Bar-Hillel
Hebrew University

People's tendency to rely on representativeness (R) when making judgments of the probability (P) of various events can result in two major kinds of fallacies. Those that are inherent in the very substitution of P by R, and those that accompany the reliance on R as a side effect. By the first I mean fallacies that result from the fact that the logic of similarity differs from the logic of P. Thus, adding detail to the description of some event enriches it, and may thereby enhance its judged similarity to some criterion (Tversky, 1977). But this adding of detail also makes the event more specific, hence necessarily less probable. Kahneman & Tversky (K&T) showed, e.g., that Ss consider it more likely for Bjorn Borg to lose the first set in a tennis match and then win the entire game than merely to lose the first set, though the latter event includes the former. Fallacies that are side-effects of R are those that result when the outcome of judgment by R is not modified or integrated with other relevant considerations.

Early studies of P judgments linked certain common judgmental errors to R causally. In particular, people's tendency to neglect the effects of base rate, sample size and data reliability was seen as resulting directly from the fact that these factors do not affect R. Later studies cast some doubt on this link, for the following reasons.

a. These factors are sometimes ignored even in R-free tasks. Consider, e.g., the Suicide Problem (B-H, 1980)

> A study of suicide among young adults found that the rate of suicide is 3 times higher among singles that among marrieds in this age group. What would be the proportion of singles in a sample of suicide deaths of young adults?

The common response to this problem is 75%.

b. In R-free tasks, these factors sometimes exhibit a systematic effect on judgments of P. E.g., Ss judge it more likely that a large sample would provide an accurate estimate of the population mean than a small sample, ceteris paribus (B-H, 1979).

c. This effect is sometimes manifest even in the presence of R. In one version of the Tom W. prediction task, subjects were lead to expect either high or low predictive accuracy. While both groups gave essentially the same predictions, the low expected accuracy group expressed less confidence in their predictions. Thus, data reliability was not altogether ignored, though it wasn't properly combined with the R considerations either. Rather, it was translated into an expression of confidence in those considerations (K&T, 1973; B-H, 1981).

As a result of such findings, K&T recently moderated their formulation of the R heuristic, saying: "The magnitude of R biases and the impact of variables such as sample size, reliability and base rate depend on the nature of the problem, the characteristics of the design ...", etc.

It is illustrative to consider the role which normative staticstical theory assigns these three neglected factors. Take a prototypical statistical problem, that of reconstructing the parameters of some population on the basis of a sample of data. In the case of pure estimation, statistical theory teaches us that many "essential characteristics" of samples are unbiased estimators of corresponding population parameters. Hence estimation reflects R. So does the statistical notion of goodness-of-fit. When, on the other hand, alternative hypotheses compete, as in hypothesis testing, it is a notorious fact that classical statistical theory (but not Bayesian statistics) has no place for prior probability considerations. Yet these play the role that the base rate plays in prediction tasks such as Tom W.

As to sample size and data reliability, their role in both estimation and hypothesis testing lies in determining the width of a given confidence interval, but not the cenral value around which it is constructed. Analogously, these factors typically seem to effect Ss confidence in their predictions though not the predictions themselves.

In the Bayesian approach, P measures an internal state of uncertainty. Through the subjective filter all sources of uncertainty can be passed and integrated, and thus there is no call for higher order Ps. Psychologically speaking, however, people seem to distinguish between variants of uncertainty (K&T, 1982), and so may hold 2nd order P distributions (e.g., confidence) over 1st order P distributions (e.g., propensities) that are, subjectively, nonintegrable. It is compatible with points a., b. and c. above to hypothesize that R may be a heuristic for assessing 1st order Ps, and that factors which do not affect R may still influence 2nd order Ps. Whether they affect the ultimate P value may depend on the integrability of 1st and 2nd order considerations (B-H, 1982).

It should be apparent that the attempt at drawing analogies between the intuitive treatment of variables and the one formalized by normative theories is in no way an apologia for people's fallacies, which are genuine and worrisome. Cohen (1981) claimed that since the "presence of fallacies in reasoning is evaluated by referring to normative criteria which ultimately derive their credentials from a systematization of the intuitions that agree with them", people's deeply rooted statistical intuitions cannot, in principle, be fallacious. The point is moot, however, since clearly the output of defensible intuitions may itself be indefensible.

So far, I have tried to make the case that R is not just a fundamental feature of lay judgments under uncertainty, but of normative statistical theory as well. A world not governed by R might well be unthinkable. Just try to imagine a breakdown of the "law of averages". Physically uniform coins fall on Heads much more often than on Tails; well shuffled decks of cards yield Hearts more frequently than other suits; repeated independent measurements yield skewed, bimodal distributions; etc. Such a world, to rephrase Einstein, can only be the creation of a God who is not only subtle, but malicious as well.

Even though R may be essential to everyone's basic metaphysics, in particulars an

ideal statistician, IS, may apply R more astutely to statistical inference problems than a layperson, L. We will now consider some such particulars, the idea being to show how refining R by simple, qualitative, statistical principles can lead to more appropriate solutions than R "in the raw".

i. Predicting sample features by R. Often the best prediction for an as yet unobserved sample is that it will resemble an already observed one, or the population that is its source. Clearly, however, it is too much to expect every feature of the past sample to be repeated in the future one. Yet, sophisticated respondents believed that, having obtained a just significant result in an experiment with 20 Ss, the chances of now obtaining a significant result on a new sample of 10 is 85% (K&T, 1971). Result significance, however, is a somewhat arbitrary notion. Since it depends on the sample size as well as the mean, expecting the sample mean to replicate (which is reasonable) should lead to more uncertainty about that mean's significance, since sample size was halved.

Other respondents expected a sample (n=50) from a population with mean=100 to have such a mean as well. They held on to that expectation even when told that the first observation was 150. It is impossible for both the unknown portion of the sample (n=49) to repeat the population mean, and for the sample as a whole to do so (K&T, 1972).

In some school, program A consists of 65% boys, while program B of 45% boys. Ss expected classes belonging to Program A to resemble the program's composition more than the other program's. The similarity of some class' proportion of males to 65% versus 45% should be evaluated in terms of standard deviations. Ss seemed to evaluate it in terms of which sex was the majority, thus expecting a class of 53% boys to belong to Program A.

ii. Features of Gestalts versus features of data points. The statistical properties of samples are completely determined by the individual data points of which they are comprised. Features that accrue to the sample as a whole, but not to its constituents (e.g., its mean) are significant insofar as the individual data points are unknown or discarded. Thus, a sample whose mean is near the population mean is more likely, ceteris paribus, than one with a more deviant mean. But this order may be upturned when the specific data points are given. L seems to find it difficult to ignore the emergent properties of samples as Gestalts, even when they are completely specified. IS, on the other hand, would ignore thses emergent properties when specific data points are available. Hence, unlike L, IS, believing that Heads and Tails are equally likely outcomes for the toss of a fair coin, would consider any fully specified sequence of fixed length comprised of equiprobable outcomes to be equiprobable. Similarly, IS would judge the P of a sample of fixed size drawn from a normal distribution to depend on the magnitude of the standardized deviation between the sample points and the population mean, rather than on its directionality. (L's errors are documented in K&T, 1972, B-H, 1980b).

Clearly, the Bjorn Borg example at the beginning of this paper can also be understood in terms of emergent properties. The P value of wholes is derivable from their parts. The R value may not be.

Summary. The reliance on R as a judgmental heuristic is frequntly justifiable, and seldom avoidable. The modification of R considerations by other considerations of relevance, and the refinement of the domain of R, its metric, etc. is a goal to be sought. Inasmuch as the various judgments of R embody much of our substantive knowledge regarding the issue being judged, R can not be eliminated from the probabilistic reasoning process, but the different logic of R and P poses obstacles that must be watched out for.

References

B-H, 1979. The role of sample size in sample evaluation. OBHP.
B-H, 1980. The base rate fallacy in probability judgments. Acta Psychologica.
B-H, 1980b. What features make samples appear representative? JEP:HP&P.
B-H, 1981. Representativeness reconsidered. unpublished manuscript. DR report, Eugene.
B-H, 1982. Ideal evidence, relevance, and second order probabilities. Erkenntnis.
Cohen, 1981. Can human irrationality be experimentally demonstrated? B&BS.
K&T, 1971. Law of small numbers.
K&T, 1972. Subjective probability: A judgment of representativeness.
K&T, 1973. On the psychology of prediction.
K&T, 1982. Variants of uncertainty.
T&K, 1982. Judgmentsof and by representativeness.
(all these can be found in Kahneman, Slovic & Tversky, Judgment under uncertainty, CUP, 1982)
Tversky, 1977, Features of similarity. Psy Rev.