PROCEEDINGS OF THE INTERNATIONAL CONFERENCE
ON FIFTH GENERATION COMPUTER SYSTEMS 1988,
edited by ICOT.

PROSPECTS FOR COGNITIVE SCIENCE

Herbert A. Simon

Professor of Computer Science and Psychology
Carnegie-Mellon University
Psychology Department
Pittsburgh, Pennsylvania 15213, U.S.A.

ABSTRACT

From the very beginning, researchers in artificial intelligence and cognitive science have been accused of excessive optimism. I hope we have been guilty of some optimism, and in a field that has moved as far in thirty years as this one has, I deny that there has been excess.

Our understanding of both human intelligence and machine intelligence continues to widen and deepen at a rapid pace, and if there are any limits to the kinds of intelligence that can be represented by computer programs, those limits have not yet made themselves evident. If I have been skeptical that we need anything that is properly described as a "breakthrough" we can proceed further, I am not at all skeptical about the research possibilities for important new ideas and advances.

Mankind has been enthralled by four great questions: the nature of matter, the origins of the universe, the nature of life, and the emergence of mind from matter. It is the privilege of all us in the cognitive sciences to spend our professional lives grappling with the fourth of these questions.

Until the computer was recognized as the general physical symbol system that it is, we had almost no tools for investigating the nature of intelligence and mind. Combining its intelligence with ours, we will continue to move rapidly toward a fuller and clearer conception of the minds of both computers and people.

1 PROSPECTS FOR COGNITIVE SCIENCE

Ten year predictions of the development of a science are rather more feasible than one-day predictions of the stock market. Science and technology do not proceed in instantaneous breakthroughs. Instead, momentous events cast long shadows before them. It was forty years from the recognition of the black body problem to Planck's law, another five years before the significance of the quantum in that law was recognized, another eight before Bohr constructed his theory of atomic structure, and another thirteen before Heisenberg and Schroedinger provided the modern equations of quantum mechanics. While no one (without making the discoveries themselves!) could have predicted these discoveries, much less their exact timing, there was no great problem in predicting where the good research problems lay, and hence, the promising directions of research.

A ten year look ahead to the future of cognitive science does not, therefore, seem too formidable a task - provided that you do not ask me what the precise results of the research will be or just when they will occur. If we wish to sound bold, we can even call this a look into the next century, which is, after all, little more than a decade away. To look ahead in this way, we must first study the shadows that portend the events: We must look briefly at where cognitive science has been, and at where it now is. That is where I will begin.

2 THE ACHIEVEMENTS OF COGNITIVE SCIENCE

In what follows, the terms "cognitive science" and "artificial intelligence" will be used more or less interchangeably. Both domains are concerned with producing intelligent behavior in computers. Cognitive science wishes to do so in order to understand human intelligence, and the programs it writes are intended to use the same kinds of methods that people use in their thinking, speaking, understanding, learning, and so on. AI wishes to produce machine intelligence in order to augment human intelligence, and in the programs it writes, no holds are barred. AI can use machine nanosecond or picosecond speed that is unavailable to the millisecond human brain. But the two fields have worked so closely together since their origin, and have borrowed so freely back and forth, that it is not necessary to consider them separately; the progress of each will continue to depend upon and to support the progress of the other.

2.1 Programming Languages

The very existence of cognitive science depended on having programming languages that allow complex, irregular, constantly changing, unpredictable structures to be stored in memory and processed. The invention of list processing languages, as early as 1956, provided cognitive science and AI with the programming tools it needed and which it has relied upon since. Initially rebuffed and scorned by systems programmers as intolerably slow and wasteful of computer memory, list processing languages turned out to be a major contribution of AI to computer science generally.

Beginning around 1972, LISP, which had become the standard list processing language, was joined by production system languages like OPS5; and more recently by logic programming, as exemplified in PROLOG, has been added to the kit of AI tools. The current interest in connectionism and parallel networks is sure to spawn still another class of programming languages - early examples can already be seen.

2.2 Hardware

The obvious precondition for cognitive science was the existence of computer hardware to support the execution of its programs. The dependence of the research on computers is beyond question or discussion. What is less obvious is whether hardware availability has been a major determinant of the speed at which the research has advanced. Has hardware been the bottleneck, or has it always been available when needed, with the required memory capacities and operation speeds?

One must give a mixed answer to this question. Current systems for visual or auditory pattern recognition, chess programs, and some other expert systems, could not operate at tolerable speeds on the computers available as recently as five or ten years ago. In this sense, the remarkable and continuing advance of hardware has been absolutely essential to the development of AI and cognitive science. We may expect this to continue to be true in the future: we will find needs for even faster computers, with ever larger memory capacities.

But one can ask a different question: Has hardware development been the bottleneck that has limited the rate of progress in cognitive science? Here, the answer is largely negative. The rate at which machine intelligence has been pushed into new domains and new levels of performance has depended mainly on the ingenuity of the researchers. When new ideas have been invented, hardware has usually been available to implement them - not always with the speed and scope that we might wish, but sufficiently well to test the soundness of the ideas. Basic research seldom has to be carried out in real time.

There are exceptions. In designing programs to play chess, machine speed has been of the essence, and much (not all) of the rapid progress of the past five years has come from the availability of special purpose hardware. Some of us have long believed that computer chess research should put more emphasis on incorporating chess knowledge in the programs, and less on speeding up brute-force search. However, the history of progress in that field does not support our position. Will the future, as chess programs reach grandmaster levels, be different? That spot on my crystal ball is rather foggy.

A few years ago, the idea was popular that AI programming would be greatly facilitated by the availability of special LISP machines or PROLOG machines. Those machines now exist and they achieved a speedup - but only that. They allow us to execute important primitive operations more rapidly, but they still compete with powerful general-purpose hardware, and the verdict is not clear whether the special-purpose machines will be cost effective. At any rate, they represent not a "breakthrough" but just another source, one of many, of speedup in hardware.

In the case both of PROLOG and of languages for connectionist programming, it is widely believed that major problems of execution speed would be solved if we had massively parallel hardware, and much effort is now being devoted to bringing about that result. I am skeptical on two scores: (1) that parallel hardware is the answer to exponential explosion of search (a problem that plagues PROLOG), or (2) that it is, even in principle, feasible to design parallel hardware that has genuine general purpose capabilities.

I will have more to say later on both these points. For the moment, I would simply observe, first, that some impressive special-purpose parallel hardware has already been produced (e.g., array-processors, or the chess machines mentioned earlier); second, that there are no convincing demonstrations of massively parallel general-purpose hardware; third, the newer supercomputers, with only a little parallel capacity, but offering the fastest computing that is available, are used mostly for numerical analysis, and except for connectionist research, have found relatively little application in cognitive science.

The significance of the first observation is that we can, indeed, achieve major speedups (which wound, but do not slay, the dragon of exponential explosion) by parallelism adapted to special uses. The significance of the second is that we do not now know how to bring about several-order-of-magnitude speedups in general-purpose parallel architectures. The significance of the third is that success in designing effective parallel systems may not be the key to progress in cognitive science.

2.3 Programs

For this audience, I do not need to list the many domains in which computer programs exist that reach or surpass human levels of intelligent behavior, or the many answers we have gained to our questions about how the human mind manipulates symbols in thinking, problem solving, language understanding or learning. I shall only try to summarize some of the common characteristics of these programs, characteristics that seem important to defining the nature of intelligence itself.

First, we do achieve speeds in computer programs that are simply unattainable for people - speed in arithmetic operations being the most striking example. Nevertheless, we have found that speed and brute force, unless combined with heuristics borrowed from our understanding of human cunning, does not go far toward achieving intelligence.

Very early in research on human intelligence, some of the heuristics were discovered that permit people to search very selectively in problem spaces that would otherwise be far too large for human computational capabilities. Even rather simple hill-climbing heuristics, which select the next search step with the aim of increasing some evaluation function, have proved powerful for reducing the necessity for extensive search. More sophisticated, and widely used, is means-ends analysis, which guides search by comparing the current problem state with the goal state, detects differences between them, and takes actions to reduce the differences.

These and other search heuristics were found through research on relatively simple puzzle-like problems, which don't require much real-world knowledge of the solver, but which nevertheless can be quite difficult for people when they first encounter them.

The intelligence of experts, on the other hand, is most often applied to domains that have a large information content. We know today that the human world-class expert (in every one of the dozen or more domains that has been studied intensively) bases his or her expertise on the possession of vast knowledge as well as on the ability to do means-ends analysis or other forms of inference. The expert typically knows 50,000 or more "things" (we call them "chunks") in the domain of expertise. We have evidence that this knowledge is stored in a production-like form, like an indexed encyclopedia. The index enables the expert to recognize key factors (cues, symptoms) in situations in his or her domain, and thereby to access the knowledge stored in the encyclopedia about the significance of those symptoms.

Each of us, expert on our own native language, recognizes while reading any one of 50,000 or 100,000 words in this language, and retrieves immediately from memory our knowledge about the meanings of these words. Doctors do the same with medical symptoms, chess masters with visible features of the chess board, and so on. We know (because we have done it) that we can build expert systems, capable of performing at the level of human experts, by constructing such encyclopedias in the form of production systems and endowing them with a little capability to do means-ends reasoning or other inference.

Today, we also know that those responses of human experts that we call "intuitive" or judgmental," or even "creative," are precisely acts of recognition, based on the 50,000 chunks held in memory. As Pasteur put it, "Accidents happen to the prepared mind." Knowing that, we have pushed our computer explorations into the domain of ill-structured problems and creativity. Programs have been constructed and tested, like EURISKO, BACON, and KEKADA, that are capable of creating new concepts out of old, and scientific laws from raw data, and which can plan, intelligently, sequences of experiments for achieving a research goal.

Most of the accomplishments of cognitive science up to the present time relate to the programming of relatively well structured tasks, where the goals and admissible operators are fairly clearly defined. More recent successes with programs that do scientific discovery - and in quite a different realm, with programs that compose music and make creative drawings - raise our aspirations for the field. The tasks performed by such programs involve vaguely defined goals and no clear boundaries for the legality of "moves." There are no longer, if there ever have been, clear limits to the kinds of human cognition that can be analyzed by the methods of cognitive science.

When cognitive science began, I suppose most of us thought it would be easiest to write programs to do ordinary everyday things - observing, recognizing, making physical movements - while it would be very hard to simulate the "higher" flights of the human mind into scientific and professional activity. It turned out exactly the opposite. Professors, engineers, and businessmen have been much easier to simulate than bulldozer drivers. Building systems to match the human eyes and ears, and their control over the fingers and hands, has proved to be the most difficult of all our research challenges.

We should have predicted that. (We didn't.) The mammalian sensory and motor systems have been evolving for nearly a half billion years. There has been plenty of time for natural selection to hone them into complex and finely-tuned devices. The new brain, which distinguishes us from the rest of the mammals, has been developing for less than a million years. With such a short period of shaping, it is probably still a very simple and crude device - indeed that is exactly what our cognitive science research and expert systems have been revealing to us. For this reason, the hardest problems ahead are still those of understanding and simulating the sensory and motor systems.

From the very beginning, cognitive science has been fascinated with the processes of learning, but fascination has not always been accompanied by rapid progress. In the early years, we had one spectacular success, Samuel's program for learning to play checkers. But we also had disappointments, for example in our experiences with Perceptrons and other schemes for self-organizing networks.

In the past decade, however, research in learning has taken off again, and much has been accomplished. One of the most significant accomplishments is the understanding we have gained about how people can learn by examining worked-out examples, and then "re-programming" themselves to retain the skills employed in working the examples - skills that then can be transfered to other problems.

The computer counterparts of schemes for learning from examples and learning by doing (i.e., by solving problems) are adaptive production systems. Adaptive production systems are simply production systems that can form new productions and add them to memory. Beginning with the work of Neves a decade ago, it has been demonstrated convincingly that such systems can be built for learning subjects, like algebra and geometry, at high school level.

The insights we have gained from adaptive production systems that learn from examples have already been applied in several research studies to the improvement of human teaching and learning. For example, in a study carried out with Chinese colleagues in a Beijing public school, we have shown that the entire three-year algebra and geometry curriculum can be taught from examples and problem-solving practice, without lectures or textbooks. Moreover, the new methods are more efficient in terms of student time and level of learning than the standard classroom methods.

2.4 Applications

I have already mentioned some of the main areas of application of cognitive science today. Most of them we bundle together under the heading of "expert systems," but that label covers a vast and growing collection of different kinds of programs, as the recent book by Feigenbaum, McCorduck, and Nii demonstrates.

Other real-world applications are barely on stage, or remain in the wings. Robotics has had much visibility, but most robots working in factories today have their basis in classical control theory rather than artificial intelligence. Developing robots that incorporate genuine AI techniques will depend on our progress in making sensory and motor devices, a topic that was mentioned earlier.

We now understand how humans manipulate natural language and extract meaning from it (or put meaning into it). As a result of that understanding, we now know what the REAL problem is of achieving effective and practical language understanding and language translation by computer. The real problem is that a language translator must itself have a great deal of semantic knowledge about the subjects it is translating. Applications in this domain will be paced by progress in building large semantic memories that are organized so that relevant information can readily be found and accessed as needed.

Clearly, these potential fields of application of cognitive science depend for their realization on progress in basic research. It is time for me to turn now to a discussion of the frontiers and prospects of such research.

3 RESEARCH FRONTIERS

My discussion of where we are today in cognitive science provides the basis for my forward view from the frontiers, and enables me to be relatively brief in describing the prospects that I see from there. I will look successively at some of the areas that I have already identified as critical; then I will say something about our needs for software and hardware supporting systems, and our prospects for meeting these needs.

3.1 Task Domains

The topics on which I wish to comment are robotics, language, expert systems, learning, and representation. All but the last of these has already been discussed briefly.

3.1.1 Robotics

Earlier, I identified the development of sensors and effectors as the key to progress in robotics. These hard problems are attracting much research attention today, and there is not much I can say about them that cannot be said better by the researchers. Progress is slow, but there is definite movement. We should expect it to continue and to accelerate, but we should not look for a sudden "breakthrough" that will dissolve all of our difficulties. My crystal ball shows no breakthrough on the horizon.

Clearly, the sensory domain - visual and auditory pattern recognition - is the area where connectionist ideas could make their earliest and most important contribution. The evidence is very strong that most of the human "higher" mental functions are carried out in a serial, one-at-a-time fashion, all passing through the narrow bottleneck of attention.

The difficulties we experience in carrying on a serious conversation while driving a car in heavy traffic is just one of the evidences that, while we may have some time-sharing capabilities, we are not, at this level, parallel processors.

The evidence is equally clear that the eye and the ear, and to a lesser extent the motor system, are parallel devices. It is here where the main connectionist research effort needs to be focused. Some connectionists are more sanguine. They think the whole of cognitive science can be handled with their models, without the need for a separate symbolic level. Time will tell (obviously I don't agree with them).

Finally, there is more to robotics than sensory and motor systems. There must be a thinking and planning system to connect them. Much of the basic equipment and organization for it is in place (as witness systems like STRIPS), derived from the research on problem solving. But the problem that needs more attention, and is just beginning to receive it, is how a planning system, using a very gross and inexact model of the real world outside, guides a robot that has to survive and operate in that real world. This need presents problems of correction and feedback of planning models, problems that are surely solvable, but that need to be addressed.

3.1.2 Language

The long shadow that predicts the main direction of language research is, as I have already explained, the shadow of semantics. Already, Lenat and his colleagues in Texas are engaged in building an information base of encyclopedic dimensions that can be used to test the use of semantic knowledge in informing and guiding language understanding and translation systems. I would expect to see more enterprises of this kind, guided by what we already know about large expert systems, production systems, and data base architectures.

If there are any fundamentally new ideas that have to be invented before progress can be made, they are not visible to me. Undoubtedly, new ideas will emerge as the work progresses - intelligent empirical work always produces them - but what is needed right now and in the near future is large-scale experimentation with data bases.

3.1.3 Expert Systems

Development work on expert systems hardly needs encouragement or guidance. The demonstrations of commerical value are sufficiently numerous and convincing to support vigorous continuing work. The expansion of expert systems to new domains will depend, however, primarily on the progress that is made in the other dimensions of research I have been discussing: especially sensory systems, robotics, language, learning, and (a topic not yet discussed) representation.

3.1.4 Learning

There are two or three main foci of learning research today. I have already mentioned the two that seem to me most promising. One is connectionist research for the learning of visual and auditory patterns. The other is research on adaptive production systems that learn from examples. I am fairly sure that these two do not cover the whole range of mechanisms the human brain uses to improve its performance, but they seem to be among the most important, and we understand enough about them so that research can progress rapidly.

An intriguing question, on which one can find many opinions but little evidence, is when one should choose learning and when one should choose programming as the preferred method for giving new knowledge to an expert system. Human experts gain all their knowledge by learning, but that is perhaps because we don't know how to open the box and stuff the program in. Would we program people, instead of teaching them, if we could? This is a question that is bound to attract attention as we gain increasing ability to build learning systems, then seek to incorporate these learning capabilities in expert systems.

3.1.5 Representation

Information has to be taken out of Plato's abstract world of ideas and provided with some concrete form of representation before it can be processed by computers or brains. The typical representations we use in cognitive science today include list structures, and schemas constructed from them (alias descriptions, scripts, frames), productions, and declarative statements. All of these are good for stating propositions, or can be made to look as though they were.

There is a great deal of evidence, however, that people use pictures and diagram-like structures as preferred representations in much of their thinking. Einstein was always adamant in insisting that he did not think in words, and many other scientists and mathematicians have endorsed his view. We are just beginning to ask what these non-propositional representations might be, and how they can be implemented in computer systems, and simulated for cognitive research.

Novak's ISAAC program, which understood physics problems stated in words, then wrote the equations and solved them, is an important landmark in our understanding of pictorial representations. For ISAAC understood the words by transforming them into a picture (actually, a list-structure schema) in computer memory. It was easier to form equations from the picture than from the verbal statement of the problem.

It is sometimes said that "a problem well represented is a problem half solved." Certainly, we know problems that are almost impossible to solve as presented, but which become quite easy when a good representation is found. The mutilated checkerboard problem, introduced into AI by John McCarthy, is an example. What we are now learning about diagrammatic representations provides a foundation for substantial progress during the next decade, both in widening the range of representations we can employ, and in building systems that have some capability for finding good representations for the problems that confront them.

4 SUPPORTING SYSTEMS

Having described a number of directions for the future of cognitive science, something needs to be said about the hardware and software supporting systems that will permit the research to go forward - or better yet, that will facilitate it. I will arrange my discussion around four important issues, that have already been raised in our examination of the present state of affairs: serial and parallel architecture, connectionism, logic programming, and non-verbal representations.

4.1 Serial versus Parallel

The brain is a vast network of neurons, whose number has variously been estimated 10^9 or 10+(12). The dendrites are themselves branching structures, so that the total number of connections in the brain may be of the order of 10+(15). In the face of this architecture, it is quite natural to think that intelligence must require parallel computation. Why else would the processes of evolution have created this vast potential for simultaneous activity?

But the matter is not quite this simple. First, our computers also have vast memories holding information in parallel. Of course, these memories are passive structures, not active computing devices. But neurophysiologists have not yet discovered to what extent the neurons are also essentially passive memory structures, and to what extent they carry on active computation that goes beyond self-maintenance.

Second, as pointed out earlier, the human thinking process contains a narrow bottleneck, the bottleneck of attention, which severely limits the number of thoughts that can be entertained at one time. At least at the level of conscious activity, the brain is demonstrably a serial, one-at-a-time system, rather than a parallel one. It is also a very slow system (by computer standards), for even a simple act of recognition takes the better part of a second. Its slowness and seriality have made it possible to simulate such activities as problem solving and language understanding in considerable detail using general purpose serial computers. Even recognition processes, after features have been extracted from the stimulus, are easily accomplished in real time by serial discrimination nets like EPAM.

But we have already seen that, if conscious thought is demonstrably serial, seeing and hearing are demonstrably parallel. In terms of any evidence we have today, the most prudent conjecture, perhaps, is that the brain has both parallel and serial components, and that a complete computational theory of intelligence must accommodate both. There is no need to take an either-or attitude on the serial-parallel debate, in fact the empirical evidence argues against such an extreme resolution of the issue in either direction.

I have already argued that the strongest case for parallelism, especially in connectionist form, lies in the realm of visual and auditory pattern recognition, and to a somewhat lesser extent in the control of motor activity. Since these have turned out to be some of the most refractory aspects of intelligence, offering very stubborn resistance to our attempts to understand and simulate them, progress of research on sensory and motor functions would be greatly facilitated if we could provide the researchers with the right kind of parallel hardware. What "right kind" means is subject to considerable uncertainty.

As anyone who has attempted it will testify, achieving massive parallelism in computation is extremely difficult, except where hardware is custom designed to handle certain special kinds of precisely defined tasks (e.g., array processors). "General purpose" parallel processors like ILLIAC IV, and its ancestors and descendants, have proved very hard to program except for tasks whose precedence requirements matched closely the hardware design. A typical expecttation for an architecture, on tasks not closely matched to it, is to achieve a speedup by a factor of three to five with the use of 30 processors.

There is no reason to believe that someone will invent a clever idea that will suddenly make general-purpose parallelism feasible. The difficulties are not superficial, but fundamental. Basically, parallelism is constrained by the precedence requirements of the subtasks of any complex task. When there is little connection among tasks, a great deal of parallelism is attainable; when connections are dense and rigid, a large part of the potential capacity of the parallel machine goes unused while tasks await the completion of their predecessors (and while knowledge of that completion is communicated).

Nature itself is constrained from full parallelism by the informational complexity it creates. The dominant architecture of natural systems is hierarchical - with each component at a given level interacting intensively with only a few other components at that level - the protons and neutrons in an atom, the atoms in a molecule, the molecules in a cellular microstructure, and so on.

I have speculated elsewhere as to why natural structures should have evolved mainly into hierarchies. The evolutionary lesson is one that designers of computer architectures might examine closely and consider imitating. Of course we have already had considerable experience with the hierarchical organization of memories, but much less with hierarchies of active processors. Computer networks can also teach us something about hierarchization.

The conclusion I would draw is that we will continue to make progress toward the design of effective parallel systems, but probably without a sudden burst of illumination that will make that progress speedy. Moreover, parallel architectures designed with particular applications in mind are likely to advance more rapidly and to reach more satisfactory levels than attempts at general-purpose massive parallelism. As we learn more about the brain, perhaps we will gain useful ideas for parallel design from that knowledge. Meanwhile, the design of hierarchical systems deserves more attention than it has received.

4.2 Connectionism

I have stated my reasons for thinking that connectionist systems may play a large role, perhaps a decisive one, in modeling sensory and motor systems. Some connectionists, in their enthusiasm, believe that there is no longer any need for serial symbolic systems - that they will soon be replaced by connectionist nets. This seems to me exceedingly unlikely. Again, I would appeal to the evidences of hierarchy in nature as a reason for thinking that the mind is arranged in levels - that there is a level of neuronal organization, and that these neuronal systems, in turn, implement the primitive structures and operators of the symbolic systems at the next level above.

The analogy to the relation between hardware and languages, or between assembly languages and higher-level languages in computers seems quite plausible here. The argument can be stated a little more quantitatively. At the neuronal level, we are concerned with events with durations from one to ten milliseconds, while at the symbolic level, we are concerned with events enduring from hundreds of milliseconds to tens of seconds and longer.

As far as research programs are concerned, a good philosophy is to "let a hundred flowers bloom." Both connectionist and symbolic directions of research hold out great promise, and there is no great urgency now to draw exact boundaries between their respective spheres of applicability. But in particular, connectionists should be encouraged to give high priority to the problems of processing sensory stimuli.

4.3 Logic Programming

The analogy between computing and logical inference, and the consequent notion of modeling programming languages on systems of logic has a long and interesting history. Of course, it started the other way around: Aristotle modeled logic on human reasoning, and Turing modeled it on a computing machine. I will not elaborate on this history, but take up the question of logic programming as exemplified by such languages as PROLOG. I am treading on dangerous ground, for there are many persons present here who know a great deal more about PROLOG and logic programming than I do.

Simply put, the idea behind logic programming is that reasoning should be logical, and that programming languages should incorporate from logic the principles and insights that make logic a powerful and rigorous form of reasoning. Underlying any inferential system are principles, some of which are expressed in declarative form, others in procedural form. The former are called axioms, the latter, inference rules. It is an ideal of logic that both axioms and inference rules should be independent of subject matter; that they should give valid results for all possible worlds. When the logic is applied to a particular domain, additional axioms (domain-specific axioms) are supplied to specify what is known about that domain.

Because formal logic has historically been closely connected with questions of rigor in reasoning, systems of logic are usually designed to make verification of proofs as clear and transparent as possible. This is accomplished, first, by separating logical axioms from domain-specific axioms, as already explained, and second, by severely restricting the inference rules (e.g., in the system of Whitehead and Russell they include only substitution and modus ponens.)

A heavy price is paid for adhering to these principles: the reasoning proceeds by tiny steps, huge numbers of which are needed for even the simplest proofs. Whitehead and Russell paid that price (as attested by the thickness of the volumes of PRINCIPIA MATHEMATICA) because rigor was the name of the game they were playing. But there are many other games that intelligence plays and that we want to play on computers. They do not all have the same requirements.

The slow, and to some of us, disappointing, progress in computer theorem proving provides evidence of the cost of adhering to the principles of logic at the expense of alternative possibilities. Only grudgingly did the authors of early theorem proving programs admit inference procedures for equality, commutativity, and transitivity, instead of axiomatizing them. Today, single rules of inference like resolution and its derivatives are still generally preferred over systems with multiple rules. (I would point to the work of Woody Bledsoe and his associates in Austin, Texas, to illustrate what can be accomplished when, departing from this preference, heuristic principles are used freely to supplement the limited procedures of logic in building theorem proving programs.)

When we examine human reasoning, especially as it is applied to substantive affairs, it proceeds in quite a different way. There are not just a few inference procedures but many; and these are not all logical rules, but generally incorporate important domain-specific knowledge. If we watch a good student solving a problem in kinematics, we find the law of uniform acceleration is being used not as an axiom but as a computational procedure for inferring, say, distance from time and acceleration. The human processes in situations like these are readily modeled by production systems with relatively little use of declarative knowledge.

Human reasoning is a mixed bag which serves many purposes. It is used to a much greater extent to discover than to verify, and we know that discovery often requires heuristic search, taking long jumps at the expense of guarantees either of completeness or validity. The lack of these guarantees is not a virtue - it is the price we pay for living in a world where completeness and guaranteed correctness of search are computationally infeasible. Better to find an answer sometimes than to be assured that you will eventually find it (in eons?), and that if you do, it will not be a mirage. Better to check AFTER you have found a candidate than to refuse to hazard possibly false steps.

The principles I have just announced are not laws of logic, but empirical generalizations from human experience. In most real-life situations, human reasoning is, and must be, heuristic search. If powerful inference rules, even vulnerable ones, can be incorporated in the search, it will be more likely to reach its goal in a tolerable time.

Now there is no reason why logic programming cannot be carried on in the spirit of these principles, just as there is no reason why a language like PROLOG cannot be extended to equivalence with a Turing machine. But if the principles are followed, then logic programming loses its special rationale and claim to preference. Contrary to the underlying justification for logic programming, procedures will then be substituted for declarative statements, and flexible best-first search control will replace depth-first backtrack search.

My problem is not with a programming language. My problem is with what seems to me a misconception of the central principles that underlie intelligence, and that should guide the design of intelligent programs for AI and cognitive science. Among those central principles is the idea that problem solving is heuristic search.

One of the oldest issues in cognitive science, along with the competition between seriality and parallelism, is the issue of whether knowledge should be represented declaratively or procedurally. I suspect that here, as in the serial/parallel issue, the answer is "both." There is probably good reason to believe that much of our knowledge of the world is stored in declarative form, but that much of our capability for using that knowledge is stored procedurally, as sets of productions. We need to be suspicious of proposals to place the whole of intelligence, or nearly the whole, in one or the other of these forms of representation.

4.4 Nonverbal Representation

In my plea for a balance between declarative and procedural knowledge, I have defined the latter in terms of productions, but haven't said exactly what I mean by the former. "Declarative" should not be equated with "propositional." One important form of declarative representation, widely used in AI, consists of list structures and description lists (property list) structures - often called schemas, scripts, or frames.

Of course, such structures can be interpreted as sets of interrelated propositions. But though list structures and propositions may be logically equivalent, they are not computationally equivalent. In propositional representations, variables play the role that is played by common linked nodes in list structures. List structures can be used to build representations that are computationally equivalent to diagrams (see Novak, and Larkin and Simon), allowing in many cases far more efficient computation than can be achieved with sets of propositions.

List structures are not the only form of storage of picture-like information. One important alternative is the raster of pixel arrays. The computational convenience and power of such rasters of course depends heavily on the hardware and software processes that are available for manipulating them and reading information from them. The same is true of other representations, which describe figures in terms of the equations of their boundaries, or the like.

If it is true, as seems probable, that much human reasoning uses picture-like and diagram-like mental and external representations, then research on computer hardware and software for implementing such representations will be of great value. There has been substantial research activity of this kind in connection with CAD, but to the best of my knowledge, it has not been closely linked with research in artificial intelligence or cognitive science. A closer linkage could lead to very interesting and useful ideas about how to represent knowledge that is declarative, but not explicitly propositional.