Science Watch

Problem Solving and Learning

John R. Anderson

Newell and Simon (1972) provided a framework for understanding problem solving that can provide the needed bridge between learning and performance. Their analysis of means–ends problem solving can be viewed as a general characterization of the structure of human cognition. However, this framework needs to be elaborated with a strength concept to account for variability in problem-solving behavior and improvement in problem-solving skill with practice. The ACT theory (Anderson, 1983) is such an elaborated theory that can account for many of the results about the acquisition of problem-solving skills. Its central concept is the production rule, which plays an analogous role to the stimulus–response bond in earlier learning theories. The theory has provided a basis for constructing intelligent computer-based tutoring systems for the instruction of academic problem-solving skills.*

Thorndike's (1898) original learning experiments involved cats learning to solve the problem of getting out of a puzzle box. As most introductory psychology texts recount, Thorndike concluded that his cats managed to get out of the puzzle box by a trial and error process. In Thorndike's conception there was really nothing happening that could be called problem solving. What was happening was the gradual strengthening of successful responses. Thorndike's research is often cited as the beginning of the analysis of learning that occupied American psychology for much of this century. It could also be cited as the beginning of the neglect of problem solving as a topic worthy of analysis.

Although Köhler (e.g., 1927) and the other Gestalt psychologists used problem-solving tasks to demonstrate the inadequacies in the behaviorist conceptions of learning, they failed to offer an analysis of the problem-solving process. Tolman (1932) saw the critical role of goals in learning and behavior but failed to put that insight into a coherent theory, leaving him vulnerable to Guthrie's (1952) famous criticism that he left his rat buried in thought and inaction.

Problem solving finally was given a coherent program of analysis by Newell and Simon (1972) in a line of research that culminated in their book *Human Problem Solving*. The basic conception of problem solving they set forth continues to frame research in the field. Their conception had its foundation in artificial intelligence and computer simulation of human thought and was basically unconnected to research in animal and human learning.

Research on human learning and research on problem solving are finally meeting in the current research on the acquisition of cognitive skills (Anderson, 1981; Chi, Glaser, & Farr, 1988; Van Lehn, 1989). Given nearly a century of mutual neglect, the concepts from the two fields are ill prepared to relate to each other. I will argue in this article that research on human problem solving would have been more profitable had it attempted to incorporate ideas from learning theory. Even more so, research on learning would have borne more fruit had Thorndike not cast out problem solving.

This article will review the basic conception of problem solving that is the legacy of the Newell and Simon tradition. It will show how this conception solves the general problem of the relationship between learning and performance that has haunted learning theory. In particular, it provides a concrete realization of Tolman's insights. I will also present the case for problem solving as the structure that organizes human thought and means–ends analysis as the principal realization of that structure. I will argue, however, that this research has been stunted because of its inability to deal with variability and change in behavior.

Then I will turn to the more recent research on acquisition of cognitive skills. I will discuss the critical role of the production rule, a computational improvement over the stimulus–response bond, in organizing that research. I will show how the acquisition of complex skills can be accounted for by the separate acquisition of these rules, thus realizing the goal of learning theory to account for complex learning in terms of the acquisition of simple units. I will close by discussing the implications of this analysis for education, one of Thorndike's great concerns. Here I will describe my own research on intelligent tutoring systems, which has been based on the recent insights into problem solving and learning. We have been able to greatly accelerate and improve the acquisition of

Donald J. Foss served as action editor for this article.

This research was supported by National Science Foundation Grant BNS-8705811 and Office of Naval Research Contract N00014-90-J-1489. I would like to thank Allen Newell and Lynne Reder for their comments.

Correspondence concerning this article should be addressed to John R. Anderson, Department of Psychology, Carnegie Mellon University, Pittsburgh, PA 15213.

Copyright 1993 by the American Psychological Association, Inc. 0003-066X/93/$2.00
Vol. 48, No. 1, 35–44

complex skills, such as proof skills in geometry or computer programming skill. This serves to illustrate the powerful practical applications that can be achieved if only the fields of problem solving and learning listen to each other.

Canonical Conception of Problem Solving

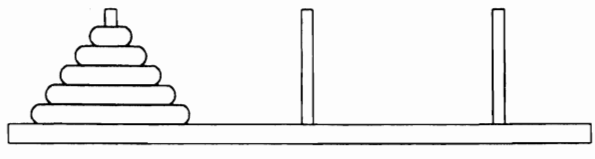
In this section I will try to sketch the canonical conception of problem solving that has its origins with the work of Newell and Simon.

Problem Space

The concept of a problem-solving *state* is probably the most basic term in the Newell and Simon characterization of problem solving. A problem solution can be characterized as the solver beginning in some initial state of the problem, traversing through some intermediate states, and arriving at a state that satisfies the goal. If the problem is finding one's way through a maze, the states might be the various locations in the maze. If the problem is solving the Tower of Hanoi problem (see Figure 1), the states would be various configurations of disks and pegs.[1] The actual reference of *state* is ambiguous. It could mean either some external state of affairs or some internal coding of that state of affairs. Newell and Simon, with their emphasis on problem solving by computer, typically took it to mean the internal coding.

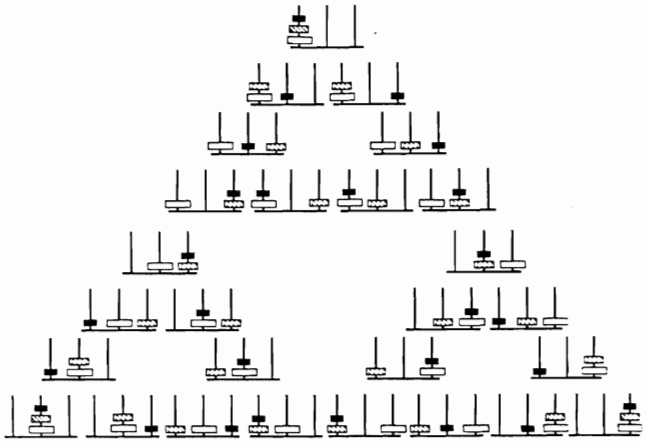
The second key construct is that of a problem-solving *operator.* An operator is an action that transforms one state into another state. In the maze the obvious operators are going from one location to another, whereas in Tower of Hanoi they are various movements of disks. An operator can be characterized by what must be true for it to apply and what change it produces in the state. In the case of the maze, there must be a path between the two locations for the move operator, and its effect is to change the location of the organism. In the case of Tower of Hanoi, the disk to be moved must be on top of the source peg and must be smaller than the smallest disk at the destination peg. Its effect is to change the location of the disk. Newell and Simon conceived of the problem solver as having an internal representation of the operators, their preconditions, and their effects.

Figure 1
Tower of Hanoi Problem



Note. The goal is to move all the disks from the start peg to the finish peg. Only one disk may be moved at a time, and one cannot place a larger disk on a smaller disk.

Figure 2
Problem Space for the Three-Disk Tower of Hanoi Problem



Note. Adjacent configurations can be reached by a single, legal move of the disk.

Together the concepts of state and operator define the concept of a *problem space.* At any state some number of operators apply, each of which will produce a new state, from which various operators can apply producing new states, and so forth. Figure 2 illustrates the complete problem space for the three-disk Tower of Hanoi problem, one of the smaller of the problem spaces. As can be seen, many problem spaces are closed with only a finite set of reachable states and loops among those states. Within the problem-space conception, the problem in problem solving is *search,* which is to find some sequence of problem-solving operators that will allow traversal in the problem space between the current state and a goal state.

In contrast to states and operators, Newell and Simon did not hold that there is an internal representation of an entire problem space. Rather problem solvers can dynamically generate paths in this space by applying their operators. This generation process can either be done externally, in which case direct actions are taken, or internally, in which case the problem solver imagines some sequence of actions to evaluate them.

Problem-Solving Methods

Whether one is performing operators externally or imagining them, the critical issue is how to select the next

[1] The Tower of Hanoi task is one of a number of "toy" tasks that had an important role in the early development of ideas about problem solving. Studies of problem solving have now extended to complex and important problem-solving tasks. However, the Tower of Hanoi task and others like it remain useful both for exposition of the basic concepts and as paradigms for studying these concepts in relative isolation.

operator. The term *problem-solving method* refers to the principles used for selecting operators. The method chosen can vary from blind search to executing an algorithm that is guaranteed to find a minimum-step solution. Problem solvers' behavior in a particular situation can be understood by knowing which method is being used. Artificial intelligence textbooks (e.g., Nilsson, 1971) frequently recount a large array of often exotic methods. Anderson (1990b) can be consulted for evidence that humans at various times use some of the simpler methods. For instance, people tend to select operators that create states more similar to the goal state (this method is called *hill climbing*). The next subsection discusses in some detail the method of means–ends analysis, which seems to be the premier human problem-solving method.

Although problem solving can be typically understood as some method applying in a fixed problem space, occasionally problem solving can progress by changing the problem space by re-representing the problem states or the operators or by adding new operators. These tend to be thought of as the more insightful problem solutions. Research on functional fixedness (e.g., Duncker, 1945) can be thought of in these terms, as can research on problem-solving representation (e.g., Kaplan & Simon, 1990).

Newell and Simon in their 1972 monograph showed how to apply their method of analysis to a number of problem-solving situations. By characterizing a subject's representation of states, his or her operators, and the problem-solving method, one is able to simulate the behavior of subjects down to the point of predicting every (or nearly every) move they make in a complex problem-solving episode. One can walk away from such an analysis with the claim of having understood the episode in a fairly rich and detailed way. Although the issue of evaluating the fit of such a simulation model to the episode has always been a sore point, often the qualitative fit can be quite compelling.

It is of interest to consider the outlines of the application of this analysis to some classic learning task, such as an animal learning to run a maze. Under this analysis the learning that takes place is effectively operator learning—learning that moving along a path will get the animal from one location to another. The performance that takes place would use this operator knowledge through some problem-solving method to achieve the goal. Thus, as Tolman (1932) insisted, learning is separate from performance, and it is goals that trigger the conversion of what has been learned into performance. Tolman was criticized for not unpacking how that conversion took place. It is the problem-solving method that converts what is learned into performance in service of a goal. Thus, the rat is no longer left lost in thought, and there is nothing nonmechanical guiding the animal through the maze.

Means–Ends Analysis

Two key features often observed of human problem solving are *difference reduction* and *subgoaling*. Difference reduction refers to the tendency of problem solvers to select operators that produce states more similar to the goal state. People are very reluctant to pursue paths that temporarily take them in the direction of states less similar to the goal (see Anderson 1990b). One of Köhler's (1927) interests was to understand the difficulties various species of animals have with detour problems that require them to take a nondirect path to the goal. So the reliance on similarity is hardly unique to humans. Anderson (1990a) can be consulted for arguments that this reliance on similarity is adaptive in that most problems can be effectively solved by moving in the direction of the goal. Of course, how one measures similarity can be tricky, and some kinds of problem-solving learning take the form of developing more useful ways of assessing similarity to the goal state. This is often characterized as problem solvers going beyond the surface features of a problem to its deep features (e.g., Chi, Feltovich, & Glaser, 1981).

Subgoaling can be nicely illustrated in the Tower of Hanoi problem. For instance, consider the following protocol of one of Neves's (1977) subjects who was faced with the Tower of Hanoi problem in Figure 3:

> The 4 has to go to the 3,
> But the 3 is in the way.
> So you have to move the 3 to the 2 post.
> The 1 is in the way there.
> So you move the 1 to the 3.

As in this case, subgoaling can involve creating a stack of such subgoals. Simon (1975) discussed the difficulty in remembering these subgoals. Anderson and Kushmerick (in press) showed that the time to make a move in the Tower of Hanoi task is strongly correlated with the number of subgoals that must be set before that move.

Means–ends analysis provides a way of understanding why difference reduction and subgoaling are so pervasive in human problem solving and how they relate to one another. Figure 4 illustrates the logic of means–ends analysis. The basic cycle of the problem solver is to look for the biggest difference between the current state and the goal state and try to reduce that difference. The problem solver makes a subgoal of eliminating that difference. Thus, if a problem solver correctly perceives the Tower of Hanoi problem, he or she would consider the biggest difference to be the largest disk out of place, as did Neves's (1977) subject. The problem solver searches for some operator relevant to removing that difference. If the operator can be applied, it is, and problem solving progresses for-

Figure 3
State of Tower of Hanoi Problems Facing the Subject Whose Protocol is Reported in the Article

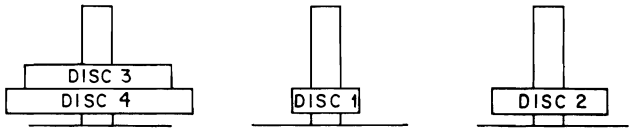
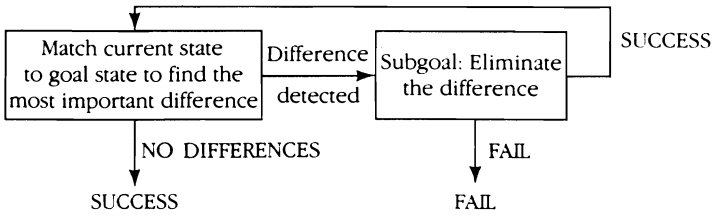
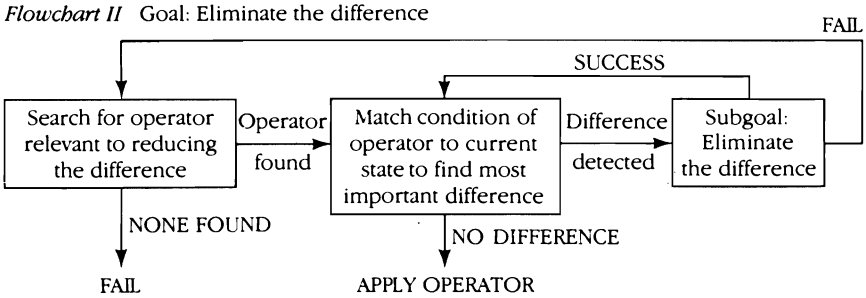


Figure 4
Application of Means–Ends Analysis

Flowchart I Goal: Transform current state into goal state



Flowchart II Goal: Eliminate the difference



Note. Flowchart I breaks a problem down into a set of differences and tries to eliminate each. Flowchart II searches for an operator relevant to eliminating a difference.

ward. However, if it cannot (as when a disk blocks the move of another disk in Tower of Hanoi), the problem solver sets the subgoal of eliminating the blocking condition. Thus, for instance, Neves's subject set the subgoal of removing Disk 3, which was blocking the move of Disk 4. The problem solver no longer is working on the original goal but is working on a subgoal, which is only a means to the ultimate end. The three key features of means–ends analysis are the focus on eliminating a single large difference, the selection of operators by what differences they reduce, and the subgoaling of the preconditions of the operator if they are not met in the current state. Anderson (1990a) can be consulted for a general analysis of why this problem-solving method can lead to optimal problem solving in novel situations.

Means–ends analysis does not just apply to exotic laboratory puzzles. Newell and Simon (1972) emphasized that it is found in all aspects of life. Consider, for instance, their following example:

> I want to take my son to nursery school. What's the difference between what I have and what I want? One of distance. What changes distance? My automobile. My automobile won't work. What is needed to make it work? A new battery. What has new batteries? An auto repair shop. I want the repair shop to put in a new battery; but the shop doesn't know I need one. What is the difficulty? One of communication. What allows communication? A telephone . . . and so on. (p. 416)

Whereas it would be incorrect to assert that all human problem solving is organized by means–ends analysis, this problem-solving method has played the largest role in accounting for behavior in puzzles like Tower of Hanoi, academic problem solving (Larkin, McDermott, Simon, & Simon, 1980), and everyday problem solving (Klahr, 1978). Often because of the structure of the problem, all the aspects of the underlying means–ends method do not manifest themselves. Thus, problem solving on certain puzzles may look like hill climbing (e.g., Jeffries, Polson, Razran, & Atwood, 1977) because the operators for the problem do not have the kind of prerequisite structure that leads to subgoaling, and so we only see difference reduction. Conversely, a problem may look like pure subgoal decomposition (Anderson, Farrell, & Sauers, 1984) because there is no similarity structure to guide the choice of subgoals.

It is of interest to speculate how far means–ends analysis is found down the phylogenic scale and developmental scales. Klahr (1978) has argued that children are quite capable of means–ends analysis. Their problem solving is often ineffective because of inadequate representation of the problem, and they become more effective means–ends problem solvers when their representations of the problem and the operators become sophisticated enough to enable means–ends problem solving to apply. Köhler's (1927) characterization of chimpanzee problem solving would seem to imply a means–ends capacity for them, even as his more dismal characterization of lower organisms would imply they do not have a means–ends capacity. There should be a very strong connection between tool manufacture and use and means–ends problem solving. A tool is a concrete means to an end. My own

belief is that the means–ends problem-solving method is an innate part of the cognitive machinery of humans and other primates.

Central Role of Problem Solving in Cognition

The remark above about the possible innate status of the means–ends method raises the issue of how to conceive of the place of problem solving in cognition generally. There is a tendency of some psychologists to view research on problem solving as a narrow domain approximately equivalent to research on mathematical behavior. That is, it is an intellectual activity that we may engage in a few times a day and that can be understood in terms of principles of cognition more general than problem solving. This is far from how some researchers on problem solving (e.g., Newell, 1980) have viewed the matter. For them, all higher level cognition is problem solving. This is an implication of the proposal made above for how problem solving provides the bridge between learning and performance. The problem-solving methods provide the mechanisms for converting knowledge into behavior, including cognitive behavior. They provide this bridge everywhere and not just with esoteric puzzles.

One problem with the claim for the central role of problem solving is that much of human cognition does not feel like problem solving. Some activities, like solving a Tower of Hanoi problem or solving a new kind of physics problem, feel like problem solving, whereas other more routine activities, such as using a familiar computer application or adding up a restaurant bill, do not. This reflects the difference between the reference of problem solving in everyday speech and its use by researchers. In everyday speech the term *problem solving* refers to activities that are novel and effortful. The theorist's claim is that the underlying organization of these activities is no different from the underlying organization of the more routine.

Newell (1980) argued that the dimension of difference between routine problem solving and real problem solving is the amount of search involved. When we become familiar with a problem domain, we learn which operators apply without having to search among them. The experience of effort is correlated with the amount of problem-solving search. Newell argued that we are always in a search space, as witnessed by what happens when we hit on some novel problem state in an otherwise routine problem space. Newell claimed that we transit smoothly into problem-solving search and indeed that much of human cognition is a mixture of routine problem solving and problem solving that involves search. This claim is realized in his Soar model of cognition (Newell, 1990).

Complications With the Canonical Conception

In this section, I consider problems with the canonical conception of problem solving that arise because of its failure to incorporate the perspective of a learning theory.

Variability in Problem Solving

One of the things that is apparent when human problem solving is considered is that the solutions produced vary across replications of the problem with different individuals or indeed for the same individual on different occasions. This variability shows up in subjects taking different paths of solutions to solve a problem and in terms of their making occasional errors in their problem solving. It is not much noted, but if one looks at the latencies one sees considerable variability in the times required to perform the same step of a solution (Anderson, Kushmerick, & Lebiere, in press). Such variability has been observed by many researchers in human problem solving but is perhaps best documented in our research on LISP programming where we observed more than 100 students solving more than 100 LISP programming problems (Anderson, Conrad, & Corbett, 1989). The canonical problem-solving framework with its emphasis on deterministic behavior is not well prepared to handle this variability.

There are two basic ways that such variability has been approached within the canonical framework. One is to attribute the differences to differences among the cognitive models of different people (and sometimes among the cognitive models of the same person at different times). In the standard framework, this comes down to differences in problem-solving representations, operators, and methods. This leads to a style of theorizing in which separate models are proposed for each subject, which creates a frustrating problem of generality in the claims that can be made.

Perhaps the most hopeful effort of this sort has been the attempt to account for errors in problem solving in terms of bugs or misconceptions about the problem domain (e.g., Brown & Van Lehn, 1980). In one notable effort, Burton (1982) accounted for a large fraction of subtraction errors by assuming over 100 different bugs. The term *bugs* comes from analogy to programming where a program can have an error that leads to a systematic mistake. It was hoped that we could come up with a theory of the origins of these bugs in terms of the learning history of the students (e.g., Van Lehn, 1989). A learning account of variability would be a way to achieve generality. Unfortunately, subsequent research has cast doubt on the systematicity of these errors (Anderson & Jefferies, 1985; Anderson & Reder, 1992; Katz & Anderson, 1988; Payne & Squibb, 1990). Often students are best characterized as doing the right thing most of the time and, when they make errors, being unsystematic in the errors they make.

The second approach is simply to assume a certain randomness in which alternative operators (perhaps some buggy) are indiscriminately chosen among. This has not been a popular move but can be found in some attempts to deal with the statistical distribution of solutions across subjects (e.g., Atwood & Polson, 1976; Jeffries et al., 1977). This approach certainly has a grain of truth to it, but it fails to reflect the systematicity that does exist in

the choices that are made. Anderson et al. (in press) were able to show that the distribution of choices among operators was strongly correlated with the optimality of the operators. Also, the frequencies of erroneous choices decrease gradually (within a single subject) with experience.

Variability in behavior and the gradual improvement of the distribution of responses with experience are, of course, the bread and butter of typical learning theories. This suggests that problem-solving approaches would do well to incorporate into their analyses some of the standard ideas from learning theory. The trick is to do this and maintain the computational power of existing approaches that is clearly needed to deal with the complex, coordinated structure of a problem-solving sequence.

Learning in Knowledge-Rich Domains

In the last decades, there has been a surge of research on how the transition is made from novel to routine problem solving as one gathers experience with a problem domain. This reflects a shift in research interest both toward learning and toward knowledge-rich, real problem-solving domains, such as physics, and away from knowledge-lean toy tasks like the Tower of Hanoi. This effort has identified both strengths and weaknesses in the canonical theory.

A great deal of this research has taken the form of comparing subjects who are relative experts at a problem-solving task with subjects who are relative novices at the task. Inferences are made about learning on the basis of the comparisons. Perhaps the most significant single observation is that no one achieves a high level of performance in any domain without a great investment of time. Hayes (1985) estimated that it takes 10 years to achieve master's levels of performance in most professional domains. This indicates that problem-solving expertise does not come from superior problem-solving ability but rather from domain learning.

Not surprisingly, there are great differences between problem-solving experts and novices as a function of the extensive learning experiences of the experts. These differences are reviewed in Anderson (1990b) and Van Lehn (1989).

Some of these differences appear to be nicely captured within the canonical model. For instance, there are changes in how experts go about solving problems. It is possible to separate these changes into what has been called *tactical learning* and *strategic learning*. Tactical learning refers to the acquisition of new, often more complex problem-solving operators. So, for instance, with practice geometry students learn to recognize vertical angle configurations involving triangles they are trying to prove congruent (e.g., Anderson, 1990b). Strategic learning refers to wholesale changes in the methods students use to organize their problem solving. So, novice problem solvers in physics work backwards from what they are trying to find to the givens of the problem, whereas experts work in the opposite direction (e.g., Larkin et al., 1980). In programming, more expert students will use top-down, breadth-first progressive refinement, whereas novices will not (Jeffries, Turner, Polson, & Atwood, 1981). In all cases, the expert is adopting approaches that are effective for that problem domain. In the case of programming, the strategy is explicitly taught as the structured programming methodology in programming courses; in the case of physics, it appears to be induced.

Experts also appear to use better problem representations. In particular, experts appear to represent problems in terms of deeper features, which are connected to problem-solving success, rather than superficial features. For instance, Chi et al. (1981) found that novices sorted problems on superficial features, such as whether they involved inclined planes, whereas experts sorted them according to Newton's laws.

Increased Problem-Solving Capacity

In contrast to these improvements that seem to be captured by changes in the problem space, other changes seem to reflect a fundamental increase in capacity for solving problems within a fixed problem space. For instance, there is evidence for improved memory for problem states. This was first well documented with respect to chess, where it was shown that chess experts were able to reproduce much more of a chessboard given a brief exposure than were chess novices (Chase & Simon, 1973). The same phenomenon has been shown subsequently in a large number of domains. It was first thought that this could be accounted for by the fact that experts had learned a great many complex problem patterns and so could store in a single chunk information that novices required many chunks to store. That is, it was thought it could be accounted for by changes in problem-solving representations. However, newer evidence and analysis now indicate that experts can store more information (more chunks) in long-term memory (Charness, 1976; Van Lehn, 1989). This increased long-term memory capacity is something outside the canonical theory. It does not contradict the canonical theory, but the canonical theory does not provide the terms to explain it.

One of the most straightforward effects of increased practice of a particular skill is that it is performed more quickly and more accurately. The form of the reduction of time or errors with practice can be shown (Newell & Rosenbloom, 1981) to be a power function of the form

$$P = AN^{-b},$$

where P is the performance measure (time or errors), A is a scaling constant, N is the number of trials of practice, and b is a constant usually less than one that reflects learning rate. The fact that learning satisfies this functional form is not altogether trivial. The typical learning function that has been proposed in most learning theories is exponential:

$$P = Ab^{N},$$

where b is again less than one. The exponential learning function has the intuitively appealing property that for each unit of practice, performance improves by a constant fraction b. This predicts much more rapid learning than

what is observed. The fact that power-law learning is ubiquitous creates an interesting connection between learning theory and problem solving because the power law also describes simple learning situations, such as learning paired associates, as well as extremely complex problem solving, such as learning to do proofs in geometry. Newell and Rosenbloom (1981) developed a theory of power-law learning that holds this result derives from learning more and more complex operators. However, this explanation applies only in the case of combinatorially complex tasks, and it does not seem to apply to simple tasks like paired-associate learning. Rather, this learning appears to reflect general associative strengthening mechanisms. Anderson (1982) argued that it is a simple strengthening process that accounts for all power-law learning including that which is occurring in combinatorially complex problem-solving tasks.

The ACT* Theory of the Acquisition of Problem-Solving Skills

The list of changes that occur with experience (only partially reviewed above) is probably too challenging to account for with a single theoretical proposal. Certainly, no one-factor theory has been forthcoming. I describe here my ACT* theory (Anderson, 1982, 1987, 1989) of the learning process, which captures some of the major empirical trends and offers some straightforward connections to more traditional research on human learning. This section concludes with a description of the application of this theory to the development of intelligent tutors.

Basic Concepts in the ACT Theory*

The ACT* theory of cognition (Anderson, 1983) makes a distinction between declarative knowledge, which encodes our factual knowledge, and procedural knowledge, which encodes much of cognitive skill including problem-solving skill. The theory assumes that problem solving takes place basically within a means–ends problem-solving structure. ACT* is a theory of the origin and nature of the problem-solving operators that feed the means–ends engine. It assumes that when a problem solver reaches a state for which there are no adequate problem-solving operators, the problem solver will search for an example of a similar problem-solving state and try to solve the problem by analogy to that example. There is substantial evidence that a subject's early problem solving is strongly influenced by analogy to similar examples (e.g., Pirolli, 1985; Ross, 1984). Anderson and Thompson (1989) have developed a simulation model of this analogy process.

This initial stage of problem solving is called the *interpretative stage.* It often requires recalling specific problem-solving examples and interpreting them. The memories retrieved are declarative memories. However, there is no necessary long-term memory involvement. For instance, students use examples in a mathematics section to guide solution to a problem given at the end of the section without ever committing the examples to memory. It is interesting in this regard to consider how amnesia patients who suffer serious deficits to long-term declarative memory might acquire a problem-solving skill. Phelps (1989) has argued that this can happen only when the examples from which they work are present in the environment and do not have to be recalled from long-term memory.

The interpretive stage can involve substantial verbalization as the learner rehearses the critical aspects of the example from which the analogy derives. There is a dropout of verbalization that is associated with the transition from this interpretive stage to a stage where the skill is encoded procedurally. *Knowledge compilation* is the term given to the process of transiting from the interpretive stage to the procedural stage.

Procedural knowledge is encoded in terms of production rules that are condition–action pairs, such as the following two from geometry:

IF the goal is to prove two triangles congruent,
THEN try to prove corresponding parts are congruent.

IF segment AB is congruent to segment DE, and segment BC is congruent to segment EF, and segment AC is congruent to segment DF.
THEN conclude triangle ABC is congruent to triangle DEF because of the side-side-side postulate.

These rules are basically encodings of the problem-solving operators in an abstract form that can apply across a range of situations. The Anderson and Thompson (1989) model shows how one can extract such problem-solving operators in the process of doing problem solving by analogy. Knowledge, once in production form, will apply much more rapidly and reliably.

Strength of Knowledge Encoding

According to the ACT* theory, a critical factor that determines both the accessibility of declarative knowledge and the performance of procedural knowledge is the strength of encoding of this knowledge, which basically reflects amount of practice. According to the ACT* theory, this strength grows as a power function of practice. (For an in-depth analysis of why it is a power function, see Anderson & Schooler, 1991.) It is this growth of strength that controls the power-function improvements occurring in skill learning. Anderson (1982) showed that, although other learning processes such as knowledge compilation are at work, the factor that controls rate of learning is strength. For instance, to compile a production rule from an example, the example has to be retrieved and maintained in working memory, which will depend on its strength of encoding. Thus, according to ACT* the ubiquitous power law of learning reflects the ubiquitous growth of strength of knowledge with practice. It is curious to note that the growth of strength in ACT* is just a particular instantiation of Thorndike's law of exercise, which he later rejected. However, there is good evidence for a law of exercise with respect to dependent measures

like speed of performance of a problem-solving skill (even in the absence of external feedback).

The concept of strength in ACT* is much like other strength concepts that have appeared in other theories of learning and memory over this century. In particular, the probability of a particular production rule applying is a function of its strength. This probabilistic manifestation of strength accounts for the gradual disappearance of errors and for the variation in how people solve problems. There can be multiple productions (some correct, some not) that might apply at a particular time, and the probability of each will reflect their strength. Thus, the ACT* theory has no problem dealing with the phenomenon of variability in problem-solving behavior. More recently, Anderson et al. (in press) reported considerable success applying the theory to the specific distribution of problem choices.

Intelligent Tutoring Research

I conclude with a discussion of the work we (Anderson, Boyle, Corbett, & Lewis, 1990) are doing on intelligent tutoring, both as an indication of the application of this approach and as a source of further evidence for the theory of problem-solving skill outlined above. Work on intelligent tutoring (for a review see Polson & Richardson, 1988) refers to efforts to create computer-based systems for instruction using artificial intelligence approaches. The approach to development of intelligent tutors that we take is called the *model-tracing approach.* It involves developing a cognitive model of the skill that should be learned (e.g., doing proofs in geometry or writing computer programs in the language LISP). This model takes the form of a set of production rules that can solve the class of problems the student is being asked to solve in the same way that the student should solve the problems. Our approach is relatively unique in the field in terms of the strong emphasis it places on use of a real-time cognitive model in instruction.

Our tutors interact with the students while they try to solve a problem on the computer. It is assumed that the student is taking an overall means–ends approach and that learning involves acquiring production rules that encode operators to use within this problem-solving organization. The tutor tries to interpret the student's problem solving in terms of the firing of a set of production rules in its cognitive model. The instruction and help it delivers to the student is determined by its interpretation of the student's problem-solving state; furthermore, its choice of subsequent problems to present to the student is determined by its interpretation of which rules the student has not mastered. One of the major technical accomplishments of our work has been the development of a set of methods for actually diagnosing the student's behavior and attributing segments of the problem-solving behavior to the operation of specific production rules. The various evaluations of the tutor have been generally positive, and we attribute our success at instruction to our success at interpreting the student's behavior. Typical evaluations have students performing approximately one standard deviation better than control classrooms (if given same amount of time on task) or taking one half to one third the time to reach the same achievement levels as control students. Currently we are working with the Pittsburgh Public Schools (Anderson, 1992) to revolutionize and greatly accelerate their high school mathematics curriculum on the basis of our model-tracing approach.

The ability to attribute segments of the student's problem-solving behavior to specific production rules has also enabled us to monitor the performance of these rules. We can measure how many errors students make on specific rules and how that error rate decreases with practice on that specific rule. Figure 5 shows some data on this issue from the LISP tutor (Anderson et al., 1989). That figure displays mean number of errors (where the maximum possible is three). We can also see, when students make no errors on a specific rule, how their time to perform the rule decreases with practice. This is displayed in Figure 6 for the LISP tutor. Both figures display average data and data from specific lessons to give a sense of variability. The dependent measure, opportunities, in these figures refers to the number of times that rule has been used in solving problems within that lesson. We look only at production rules new to that lesson.

These learning curves have a number of interesting features. First, they are plotted on log–log coordinates so that a power function should appear as a linear relationship. There appears to be a dramatic improvement in performance from the first use of a production rule to the second. After that, improvement is quite slow and apparently satisfies a power-law function. Similar data have been obtained with the geometry tutor. This dramatic first-trial improvement may reflect the compilation

Figure 5
Errors per Production Made by Students as a Function of Amount of Practice in Lesson in Which Productions Were Introduced

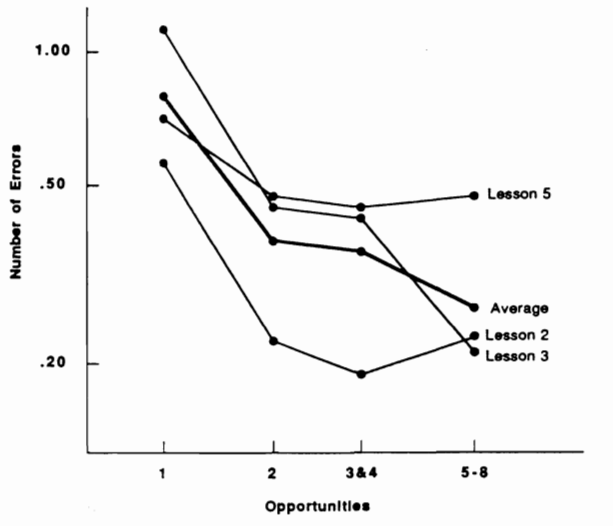
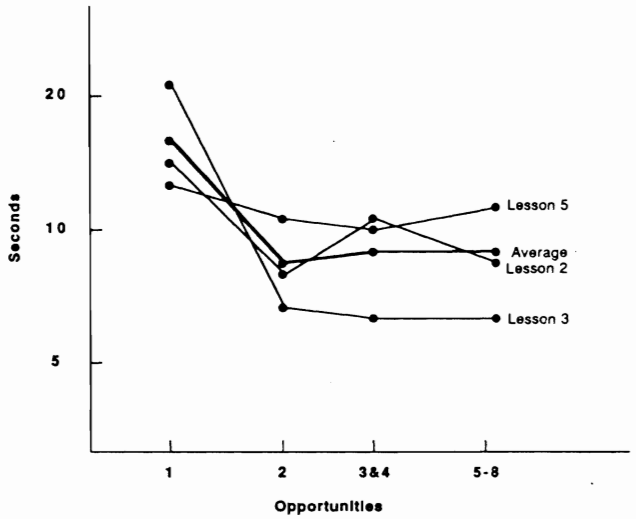


Figure 6
Time for Correct Coding per Production as a Function of Amount of Practice of Production in Lesson in Which Production Was Introduced



of domain-specific production rules. There are problems with advancing this interpretation too forcefully because it rests on the exact way the data are averaged and on relatively strong assumptions about scale. So, a first trial discontinuity remains as an intriguing possibility awaiting further research and analysis.

There are a number of additional points to make about the learning curves found in Figures 5 and 6. If one analyzes the data on the basis of surface-level categories of behavior such as writing variables, the improvement in performance does not seem orderly. Systematic learning functions show up only when defined in terms of production rules. A second point is that these rules appear to be learned independently. We do not find evidence that similar types of rules tend to be learned at the same rate as would be shown by intercorrelations in the learning rates of thematically related productions. Thus, the production rule does appear to be the right unit of analysis.

We have been able to identify some general factors that determine how well subjects perform within the tutor: In the case of LISP, these factors turn out to be (a) the speed with which subjects acquire new rules and (b) the degree to which they retain old rules. In the case of geometry these factors turn out to be (a) the success students have with algebraic rules and (b) the success they have with rules that involve spatial relations (see Anderson, in press, for a review). However, with remedial practice, students of differing abilities can be brought to equivalent levels of performance on these rules. Students brought to equivalent levels perform equally well on various nontutor posttests of ability. Thus, it would appear that acquiring a skill is basically learning each of the individual rules. The number of such rules can be large. For a modest semester's course in LISP, we estimate that approximately 500 separate production rules must be acquired.

Thus, the production rule is serving much of the same function that had been assigned to the stimulus–response bond in past theories. The skill appears to be nothing more than the sum of these rules. Each rule is learned independently, and individual differences are reflected in the learning of these rules and not the performance of these rules once acquired. Complex cognitive skill reflects the accretion of many specific pieces of knowledge.

Conclusion

I think we are beginning to see rapid and important progress being made with respect to understanding how complex problem-solving skills are learned. This progress has depended on bringing together ideas from problem-solving theory and learning theory. We can understand acquisition of complex problem-solving skills only when we recognize the problem-solving structure that organizes their performance while recognizing the rather simple learning that governs the acquisition and strengthening of the individual problem-solving operators.

REFERENCES

Anderson, J. R. (Ed.). (1981). *Cognitive skills and their acquisition.* Hillsdale, NJ: Erlbaum.

Anderson, J. R. (1982). Acquisition of cognitive skill. *Psychological Review, 89,* 369–403.

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

Anderson, J. R. (1987). Skill acquisition: Compilation of weak-method problem solutions. *Psychological Review, 94,* 192–210.

Anderson, J. R. (1989). A theory of human knowledge. *Artificial Intelligence, 40,* 313–351.

Anderson, J. R. (1990a). *The adaptive character of thought.* Hillsdale, NJ: Erlbaum.

Anderson, J. R. (1990b). *Cognitive psychology and its implications* (3rd ed.). New York: Freeman.

Anderson, J. R. (1992). Intelligent tutoring and high school mathematics. In *Proceedings of the Second International Conference on Intelligent Tutoring Systems* (pp. 1–10). Montreal, Quebec, Canada.

Anderson, J. R. (in press). *Rules of the mind.* Hillsdale, NJ: Erlbaum.

Anderson, J. R., Boyle, C. F., Corbett, A., & Lewis, M. W. (1990). Cognitive modelling and intelligent tutoring. *Artificial Intelligence, 42,* 7–49.

Anderson, J. R., Conrad, F. G., & Corbett, A. T. (1989). Skill acquisition and the LISP tutor. *Cognitive Science, 13,* 467–506.

Anderson, J. R., Farrell, R., & Sauers, R. (1984). Learning to program in LISP. *Cognitive Science, 8,* 87–130.

Anderson, J. R., & Jeffries, R. (1985). Novice LISP errors: Undetected losses of information from working memory. *Human Computer Interaction, 1,* 107–131.

Anderson, J. R., & Kushmerick, N. (in press). Tower of Hanoi and goal structures. In J. R. Anderson (Ed.), *Rules of the mind.* Hillsdale, NJ: Erlbaum.

Anderson, J. R., Kushmerick, N., & Lebiere, C. (in press). Navigation and conflict resolution. In J. R. Anderson (Ed.), *Rules of the mind.* Hillsdale, NJ: Erlbaum.

Anderson, J. R., & Reder, L. M. (1992). *Working memory load and performance in algebra.* Manuscript in preparation.

Anderson, J. R., & Schooler, L. J. (1991). Reflections of the environment in memory. *Psychological Science, 2,* 396–408.

Anderson, J. R., & Thompson, R. (1989). Use of analogy in a production system architecture. In S. Vosniadou & A. Ortony (Eds.), *Similarity and analogical reasoning* (pp. 267–297). Cambridge, England: Cambridge University Press.

Atwood, M. E., & Polson, P. G. (1976). A process model for water jug problems. *Cognitive Psychology, 8,* 191–216.

Brown, J. S., & Van Lehn, K. (1980). Repair theory: A generative theory of bugs in procedural skills. *Cognitive Science, 4,* 379–426.

Burton, R. R. (1982). Diagnosing bugs in a simple procedural skill. In D. Sleeman & J. S. Brown (Eds.), *Intelligent tutoring systems* (pp. 157–183). San Diego, CA: Academic Press.

Charness, N. (1976). Memory for chess positions: Resistance to inference. *Journal of Experimental Psychology: Human Learning and Memory, 2,* 641–653.

Chase, W. G., & Simon, H. A. (1973). The mind's eye in chess. In W. G. Chase (Ed.), *Visual information processing* (pp. 215–281). San Diego, CA: Academic Press.

Chi, M. T. H., Feltovich, P. J., & Glaser, R. (1981). Categorization and representation of physics problems by experts and novices. *Cognitive Science, 5,* 121–152.

Chi, M. T. H., Glaser, R., & Farr, M. (Eds.). (1988). *The nature of expertise.* Hillsdale, NJ: Erlbaum.

Duncker, K. (1945). On problem-solving (L. S. Lees, Trans.). *Psychological Monographs, 58,* (Whole No. 270).

Fitts, P. M., & Posner, M. I. (1967). *Human performance.* Monterey, CA: Brooks/Cole.

Guthrie, E. R. (1952). *The psychology of learning.* New York: Harper & Row.

Hayes, J. R. (1985). Three problems in teaching general skills. In S. Chipman, J. Segal, & R. Glaser (Eds.), *Thinking and learning skills* (pp. 391–406). Hillsdale, NJ: Erlbaum.

Jeffries, R. P., Polson, P. G., Razran, L., & Atwood, M. E. (1977). A process model for missionaries-cannibals and other river-crossing problems. *Cognitive Psychology, 9,* 412–440.

Jeffries, R. P., Turner, A. A., Polson, P. G., & Atwood, M. E. (1981). The processes involved in designing software. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 255–284). Hillsdale, NJ: Erlbaum.

Kaplan, C. A., & Simon, H. A. (1990). In search of insight. *Cognitive Psychology, 22,* 374–419.

Katz, I. R., & Anderson, J. R. (1988). Debugging: An analysis of bug-location strategies. *Human Computer Interaction, 3,* 351–399.

Klahr, D. (1978). Goal formation, planning, and learning by pre-school problem solvers, or: My socks are in the dryer. In R. S. Siegler (Ed.), *Children's thinking: What develops?* (pp. 181–212). Hillsdale, NJ: Erlbaum.

Köhler, W. (1927). *The mentality of apes.* New York: Harcourt, Brace.

Larkin, J. H., McDermott, J., Simon, D. P., & Simon, H. A. (1980). Models of competence in solving physics problems. *Cognitive Science, 4,* 317–345.

Neves, D. (1977). *An experimental analysis of strategies of the Tower of Hanoi* (C.I.P. Working Paper No. 362). Unpublished manuscript, Carnegie Mellon University.

Newell, A. (1980). Reasoning, problem-solving, and decision processes: The problem space as a fundamental category. In R. Nickerson (Ed.), *Attention and performance VIII* (pp. 693–718). Hillsdale, NJ: Erlbaum.

Newell, A. (1990). *Unified theories of cognition.* Cambridge, MA: Harvard University Press.

Newell, A., & Rosenbloom, P. (1981). Mechanisms of skill acquisition and the law of practice. In J. R. Anderson (Ed.), *Cognitive skills and their acquisition* (pp. 1–55). Hillsdale, NJ: Erlbaum.

Newell, A., & Simon, H. A. (1972). *Human problem solving.* Englewood Cliffs, NJ: Prentice-Hall.

Nilsson, N. J. (1971). *Problem-solving methods in artificial intelligence.* New York: McGraw-Hill.

Payne, S. J., & Squibb, H. R. (1990). Algebra mal-rules and cognitive accounts for error. *Cognitive Science, 14,* 445–481.

Phelps, E. A. (1989). *Cognitive skill learning in amnesiacs.* Unpublished doctoral dissertation, Princeton University.

Pirolli, P. L. (1985). *Problem solving by analogy and skill acquisition in the domain of programming.* Unpublished doctoral dissertation, Carnegie Mellon University.

Polson, M., & Richardson, J. (Eds.). (1988). *Handbook of intelligent training systems.* Hillsdale, NJ: Erlbaum.

Ross, B. H. (1984). Remindings and their effects in learning a cognitive skill. *Cognitive Psychology, 16,* 371–416.

Simon, H. A. (1975). The functional equivalence of problem solving skills. *Cognitive Psychology, 7,* 268–288.

Thorndike, E. L. (1898). Animal intelligence: An experimental study of the associative processes in animals. *Psychological Review, Monograph Supplement, 2* (2, Whole No. 8).

Tolman, E. C. (1932). *Purposive behavior in animals and men.* New York: Appleton-Century-Crofts.

Van Lehn, K. (1989). Problem-solving and cognitive skill acquisition. In M. Posner (Ed.), *The foundations of cognitive science* (pp. 527–580). Cambridge, MA: MIT Press.