Computational Intelligence, Volume 13, Number 4, 1997

SIMULATED ANNEALING AND THE GENERATION OF THE OBJECTIVE FUNCTION: A MODEL OF LEARNING DURING PROBLEM SOLVING

JONATHAN CAGAN*

Department of Mechanical Engineering, Carnegie Mellon University, Pittsburgh

KENNETH KOTOVSKY*

Department of Psychology, Carnegie Mellon University, Pittsburgh

A computational model of problem solving based on significant aspects of human problem solving is introduced. It is observed that during problem solving humans often start searching more or less randomly, becoming more deterministic over time as they learn more about the problem. This two-phase aspect of problem-solving behavior and its relation to learning is one of the important features this model accounts for. The model uses an accelerated simulated annealing technique as a search mechanism within a real-time dynamic programming-like framework upon a connected graph of neighboring problem states. The objective value of each node is adjusted as the model moves between nodes, learning more accurate values for the nodes and also compensating for misleading heuristic information as it does so. In this manner the model is shown to learn to more effectively solve isomorphs of the Balls and Boxes and Tower of Hanoi problems. The major issues investigated with the model are (a) whether such a simulated annealing-based model exhibits the kind of random-to-directed transition in behavior exhibited by people, and (b) whether the progressive discovery of the objective function, even when given very little or poor initial information, is a plausible method for representing the learning that occurs during problem solving and the knowledge that results from that learning.

Key words: problem solving, human problem solving, simulated annealing, reinforcement learning, Tower of Hanoi Puzzle, Balls and Boxes Puzzle, Chinese Ring Puzzle.

1. INTRODUCTION

This paper presents a model of problem solving that simulates significant aspects of human problem-solving behavior while also functioning as a computational model for problem solving. The model is best described as a hybrid model in that it combines computational approaches that are similar to a number of real-time reinforcement machine learning approaches (Sutton 1988; Barto, Sutton, & Watkins 1990; Barto, Bradtke, & Singh 1995) while at the same time attempting to model some important aspects of human performance and to do so while making cognitively plausible assumptions about many of the basic information processing mechanisms and limitations that operate in the model. The model proposes that in many problem situations, search within a new situation is initially largely random, but becomes more deterministic as problem solving progresses and, in particular, as one learns more and more about the search space. The model assumes that some landmarks might exist that can reasonably be expected to represent task-domain knowledge the problem solver brings to the task. As one moves through the search space, the indicators help one adjust their knowledge about the space, thereby learning an increasingly more accurate evaluation function that is operated on during the search process.[1] The model thus progressively develops a representation of the problem space while also searching it for the goal using a stochastic optimization technique. In particular, the computational model employs an accel-

*Order of authorship is alphabetic.

Address correspondence to the authors at Carnegie Mellon University, Pittsburgh, PA 15213; e-mail: jon.cagan@cmu.edu; kotovsky@cmu.edu

[1] Properly, this evaluation function should be called a “subjective” function because it represents the model’s current and evolving understanding of the value of states in the space. An instance of this subjective function at any time is the “objective” function that is used within the optimization framework of the simulated annealing algorithm. We will refer only to the objective function in the remainder of this paper, in keeping with the standard usage of the term in the optimization literature.

erated version of the stochastic optimization technique of simulated annealing (Kirkpatrick, Gelatt, & Vecchi 1983), used in conjunction with objective function modifications based on knowledge obtained via moves between two neighboring states. In regard to the objective function modifications, the model is similar to the temporal difference technique described by Sutton (1988). This paper in particular investigates the following issues:

- the extent to which simulated annealing is a useful description of significant aspects of human problem-solving behavior, and in particular, the transition from randomness to more directed behavior
- the extent to which the progressive discovery of the objective function during search is a plausible and useful technique for acquiring and representing knowledge gained during problem space exploration
- the extent to which the technique can work to gradually enable more and more efficient solutions of the problem without giving the system large amounts of a priori knowledge of the task-domain within which it will operate. One question addressed here is whether such a simple model of learning and knowledge representation can learn to solve problems that are reasonably difficult for humans.

The model operates in a task environment defined as a graph of linked nodes, the links radiating from any node representing the set of legal moves available from that node. Additionally, each node has an associated objective function value that initially may be unrelated to the actual value (proximity to the goal) of that state. The model attempts to move from a start state to a goal, updating as it does so the objective function of each node it encounters. The updating is based on its discovering adjacency relations among the nodes it traverses.

This work represents an attempt to determine the functionality of a model with a modest and cognitively realistic degree of knowledge about the domain it is operating in, i.e., the minimal amount of knowledge that any problem solver attempting to solve that particular problem might reasonably be expected to know about the environment being searched. Examples of such knowledge might include information attainable from the problem statement, such as major subgoals that exist on the way to a solution or, possibly, reasonable expectations about states in the immediate vicinity of the goal. This knowledge is represented in the form of particular values of the objective function at those states. In addition, a problem solver brings some general strategies or heuristics that might lead to expectations about the value of particular states. One example would be the heuristic of downhill search (or hill-climbing), whereby a value is placed on the superficial appearance of progress toward the goal. That people possess such a heuristic is a reasonable assumption about the initial state of human subjects' general heuristics, even if those heuristics are often erroneous and lead to a misleading metric of progress. In these latter, erroneous cases, which usually arise in what are called *detour problems*, an interesting issue for the model is whether it can recover from the false downhill search-induced movement to local minima,[2] and find a path to the goal. In any case, the problem description presented to the model at times includes some initial knowledge state values that are either known or assumed, and that might be expected to either help or impede the solution process. We show that in either situation the model consistently learns a correct objective function such that search for the goal becomes increasingly quick and efficient.

[2] Note that this paper focuses on problem *minimization*; the inverse discussion would hold true for maximization. *Downhill searching* and *hill-climbing* are thus equivalent heuristics in the respective representations of search as minimization or maximization.

The approach taken in this paper represents search as an optimization problem and uses a standard optimization method as the search technique. This approach is somewhat minimalist in that not only does it use a simple optimization mechanism but also a knowledge representation system that is not dependent on the strategic use of a highly elaborated knowledge base. In contrast, other methods of solving optimization problems such as monotonicity analysis (Papalambros & Wilde 1988; Agogino & Almgren 1989; Cagan & Agogino 1987), activity analysis (Williams & Cagan 1996), as well as standard numerical gradient-based approaches (Vanderplaats 1984) take more strategic approaches. The knowledge representation used in the current work contrasts with PDP approaches (McClelland & Rumelhart 1986; Rumelhart & McClelland 1986) in relying on a uniquely assigned value for each state in the problem space that is only affected by interactions with immediately adjacent states rather than a more distributed representation. The work further stands out in not relying on knowledge-rich representational systems (e.g., Feigenbaum 1977) nor powerful AI search techniques (e.g., Rich 1983).

This work uses on-line machine learning techniques to learn about the evaluation space. In particular, the method is a type of real-time dynamic programming (Barto, Bradtke, & Singh 1995), employing a $\lambda = 0$ temporal difference process (Sutton 1988) where state value updates are based on a one-step look ahead mode. The limited one-step look ahead mode is critical to this work, not for the efficiency of the machine learning technique, but rather for its relevance to cognitive processes. Our model invokes processes that are at least broadly similar to or compatible with those seen in the behavior of human subjects, in particular by not positing an unrealistic (for humans) ability to store short-term information. Other approaches, such as $\lambda = 1$ mechanisms that approach more off-line methods where no reinforcement is applied until the goal state is found, place very large demands on memory for the pathway taken to reach the goal on each iteration. These methods include those that depend on the retention of complete move records and state visit frequencies to update knowledge of the environment (Fulcher 1992). Similarly, counter-based, recency-based, and error-based methods (e.g., Thrun 1992; Sutton 1990; Thrun & Möller 1992), while very effective from a machine learning perspective, are also viewed as too demanding of cognitive resources. These methods, although able to take advantage of the computational power of the machine to be quite efficient in their ability to learn the evaluation functions, differ in not being focused on human problem solving. Although humans may be able to learn more than pairwise relations, it is extremely unlikely that they can learn and store complete move pathways while engaged in active problem solving.

Our learning mechanism differs from the literature in that our approach is meant to be consistent with human cognitive processes, not solve the machine learning problem with maximal efficiency. We do, however, borrow features from the machine learning literature. Simulated annealing is a Markovian process; our method is most similar to, and takes many features from, Q-Learning (Watkins 1988; Watkins & Dayan 1992), which works within controlled Markovian domains. Like Q-Learning, the value at each state represents the current approximation of the correct objective (or evaluation) function, the evaluation function is updated during each move, and the experimentation strategy uses probabilities to determine the move to take.

However, we introduce a unique simulated annealing method for control of the experimentation strategy. Not only is the method used to make decisions as to whether to make a move or not, but the aggressiveness of the technique is coupled with the learning framework. We introduce an *accelerated* annealing schedule that, over successive iterations, becomes more aggressive in its transition from random to deterministic search. We show that learning about the problem space is not enough; the probabilistic tendency to violate that acquired knowledge and move to higher energy (or evaluation) level states, seen to be useful early in problem solving, must be sharply curtailed through the accelerated annealing in order for the learning to be maximally effective. However, without the learning of the objective function the acceleration of the annealing does not result in an improvement in performance, and can even be detrimental to the model's behavior. Further, a typical use of simulated annealing optimizes a fixed objective function; in this work the technique is searching over a *constantly changing* objective function. We argue and support in this paper that although a pure simulated annealing mechanism is unlikely to model the complete cognitive process, the concept behind the accelerated annealing mechanism could very well be a part of the human cognitive mechanism in problem solving.

Another difference between our method and the literature is that, rather than updating the current state to the new state in some predetermined way, both the current and new states are updated based on percentages of the difference between the values of the states. This aspect of the model is thus similar to the delta rule used in parallel distributed processing models of cognition (McClelland and Rumelhart, 1986) but with the absence of an overt teacher. The perspective is that humans learn proximity of states, and thus our model learns that two states are near *each other*. Since the initial value of the states may be incorrect or approximate, both states are adjusted toward each other rather than assuming that the new state has a more correct value and thus the current state updated toward it. Rather than using the Bellman equation, the updates do not represent the number of moves to the goal but rather the relative position of one state with respect to the other; while search is taking place the updates occur independent of where the goal state lies.[3]

To the extent that the work reported herein attempts to model significant aspects of human behavior, and to represent its learning from experience without reliance on cognitively unrealistic amounts of constantly revised knowledge, or strategic decision making at each step, even at the sacrifice of computational efficiency, it represents something of a hybrid between a machine learning/AI based model and a cognitive simulation. To illustrate our approach, we focus on two problems within which human performance has been thoroughly explored in the problem-solving literature: the Tower of Hanoi Puzzle and the Balls and Boxes Puzzle (an isomorph of the Chinese Ring Puzzle). We will first describe each of these problems, discuss their generality, and examine previous empirical studies of human subjects' approaches to their solutions. From these results we propose a model of problem solving. To simulate this model we introduce our computational model using simulated annealing and objective function adjustments that progressively refine the objective function. We illustrate the computational model on both the Tower of Hanoi and the Balls and Boxes Puzzles, where the program consistently learns an effective objective function over several iterations while at the same time decreasing the number of moves needed to reach the goal state. This is true even when it is given no information or even misleading information about the true evaluation function.

Beyond showing generality of the model for solving various problems, we choose both problems because they illustrate different aspects of the model. The Balls and Boxes Puzzle has the simpler problem space of the two but is nonetheless quite difficult for people to solve. Due to the linear nature of its space, we are able to compare our model's solution to move records from humans solving the problem; as well this problem readily illustrates an important aspect of people's problem-solving behavior—the *final path behavior* discussed below. The Tower of Hanoi problem has a larger, more complicated solution space. This

[3] Other work (e.g., Sabes, 1993; Baird 1995) has updated states toward each other by adjusting both sides of the Bellman equation to guarantee convergence, not to improve a cognitive-based model.

FIGURE 1. Tower of Hanoi four-disk problem. Four disks start on the leftmost peg and are to be moved to the rightmost peg.

problem better illustrates the learning performance of the model in larger spaces and provides a framework to test the learning and search aspects of the model. The Tower of Hanoi results will be shown first to explore the capabilities of the model, followed by the Balls and Boxes results to better compare the model to records of the performance of humans solving the same problem.

2. TOWER OF HANOI PUZZLE

The first problem used in the modeling work reported here is the four-disk Tower of Hanoi problem (see Figure 1). The problem consists, in the external representation that is presented to subjects, of a series of disks placed in descending size order, on one of three pegs (the “start peg”). The goal of the problem is to move the stack of disks to another of the three pegs, the “goal peg.” Moves are subject to the restrictions that only one disk may be moved at a time, if a peg contains more than one disk only the smallest may be moved, and a larger disk may never be placed on a smaller disk.

The problem can be defined in terms of the possible “states” of the problem or puzzle and the legal moves that convert one state to another. In these terms, a problem is represented by a graph in which problem states are nodes and legal moves are links joining these nodes. This representation of the problem space as a series of nodes connected by links designating legal moves is the form of the problem space that is searched by our computer model. This mapping can be thought of as an external search space that depicts all possible legal problem configurations or knowledge states, and the connections between them. It is in this search space that the computer models we report on search for a solution. The “objective function” (see note 1) is the set of values associated with the nodes in this space. Unlike the human subjects, the computer does not have any representation of the surface features of the problem in the form of disks on pegs or, in the Balls and Boxes problem, balls in boxes.

The number of moves in the minimum solution path is $2^n - 1$, where n is the number of disks in the Tower of Hanoi. Similar relations between number of disks (balls) and minimum solution path length hold for the Balls and Boxes problem to be described below. The structure of the problem space in the Tower of Hanoi problem is, however, more complex due to the

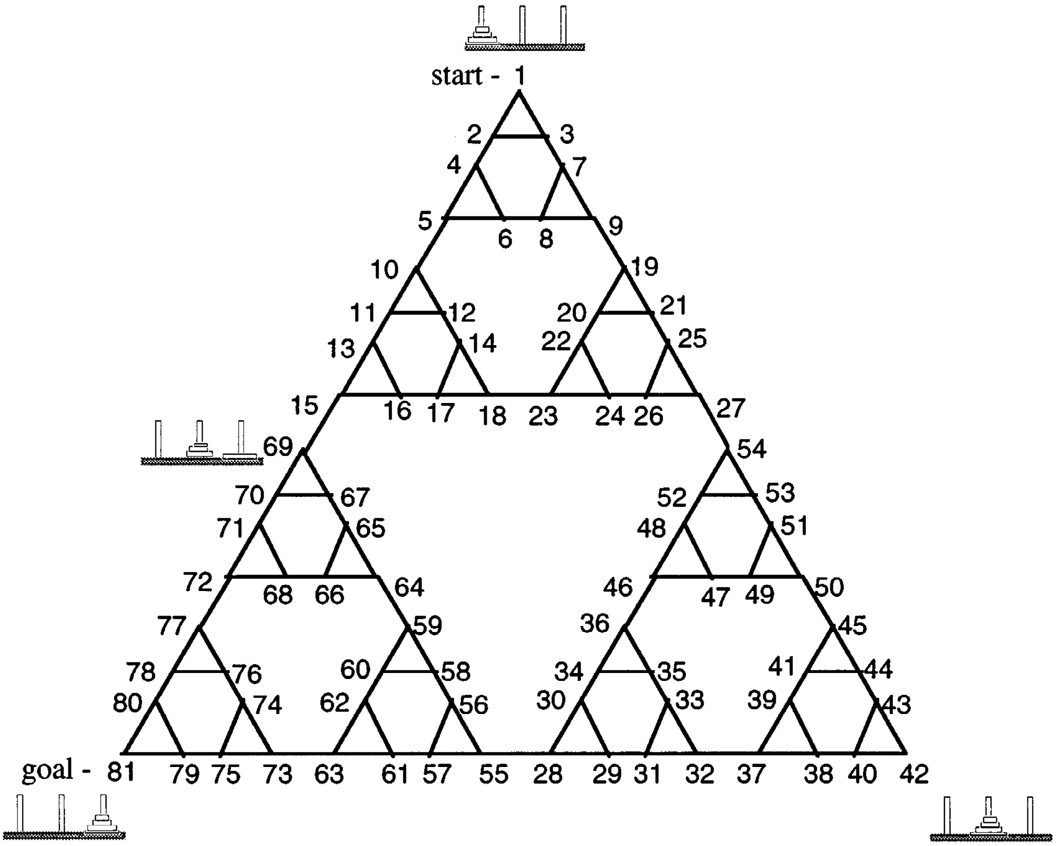


FIGURE 2. Tower of Hanoi four-disk problem problem space. Numbers label the states for identification. The typical starting position is position 1, with the goal position at 81. The most direct path traverses positions 1, 2, 4, 5, 10, 11, 13, 15, 69, 70, 71, 72, 77, 78, 80, and 81.

fact that there is a choice of three moves at all but three of the nodes in the search space (the three nodes where all the disks are piled on one peg), rather than two choices at each choice point as will be seen with the Balls and Boxes/Chinese Puzzle. The problem space of the four-disk version of the Tower of Hanoi problem is depicted in Figure 2. The space consists of 81 different states and the minimum solution path is 15 moves. As we can see from that figure, the move choices at almost all nodes consist of two possible new moves and the retraction of the most recent move (the return to the immediately prior state). The only exceptions to this are the three "corner" states where all of the disks are stacked on one peg and only two move choices are possible. The problem can be solved for any number of disks, and has the property that an n-disk problem can be viewed as consisting of an $n-1$ disk problem (moving all but the largest disk off the start peg), a move of the largest disk to the goal peg, and then another $n-1$ disk problem as the stack is moved back on top of the largest disk to complete the solution. The solution of the typically represented Tower of Hanoi Puzzle requires violations of downhill search in that it is not possible to solve the problem by simply moving disks from the start to the goal peg; it is thus termed a "detour" problem.

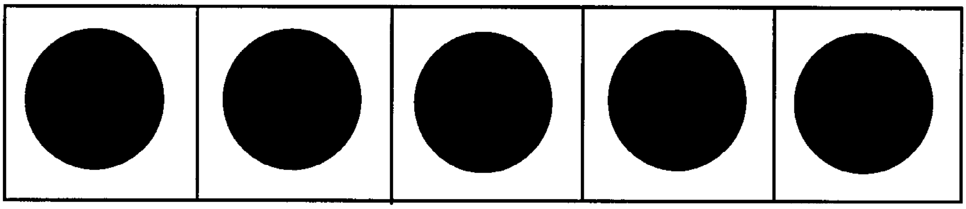


FIGURE 3. Balls and Boxes five-ball problem. The problem is shown as depicted at the start with all five balls in their respective boxes, with the goal being to remove them.

3. BALLS AND BOXES PUZZLE

The second problem explored, the Balls and Boxes Puzzle, has been used in a number of previous empirical studies (Kotovsky & Simon 1990; Reber & Kotovsky 1992) as has its much more difficult isomorph, the classic Chinese Ring Puzzle, which has been described by Ruger (1908), Afriat (1982), and Kotovsky and Simon (1990).[4]

The task, as presented to experimental subjects, is to remove five balls from five boxes. (The balls correspond to the disks of the Tower of Hanoi problem.) A move consists of inserting or removing a single ball to or from its respective box. The rules governing moves are that a ball is only free to move if the ball to its immediate right is in its box and none of the balls farther to the right are in theirs. The only exception is the rightmost ball which is always free to move. At the usual start of the problem all five balls are in their boxes and the goal is to get them all out. The situations are displayed on a CRT, and when presented to subjects, the moves are made by manipulating a mouse. This particular problem has five balls and boxes, with the usual start state having all five balls inside their boxes, the goal being to remove all of the balls. The problem is depicted in Figure 3.

The problem space for the puzzle is shown in Figure 4, which depicts the ball array that corresponds to each state in the problem space. To illustrate the move contingencies, the last five states illustrate the moves required to remove the final three balls from their boxes under the standard problem rules.[5] As always, the rightmost ball is free to move. In state 5, it is removed, producing state 4. In state 4, the leftmost ball is free to move given that only the ball to its immediate right is in its box, and it is removed, producing state 3. Next, the rightmost ball is replaced, producing state 2 from which the middle ball can be removed, given that it then has only the ball to its immediate right in the box. Notice that the first of this pair of moves (replacing the rightmost ball) could be misinterpreted as moving the subject in the wrong direction since it adds balls whereas the goal is to remove them. This necessary violation of downhill search is discussed below when the subjects' move records are analyzed. The last two moves consist of removing the middle ball and finally the rightmost one which achieves the goal (state 0). The minimum number of moves required to solve the

[4] Although the Chinese Ring Puzzle, a manipulation or "tavern" puzzle in which a metal bar must be extricated from five confining rings, is much more difficult than the Balls and Boxes Puzzle, the problem spaces are identical; thus the isomorphism.

[5] The same discussion and move sequence can be used for the rings on the bar by substituting "ring" for "ball" and "on or off the bar" for "in or out of the box".

entire problem is 21 or 31, depending on the start state as illustrated in the figure[6] (state 21 is typically used as the start state as it has all five balls in their boxes).

4. HUMAN PROBLEM SOLVING RESULTS

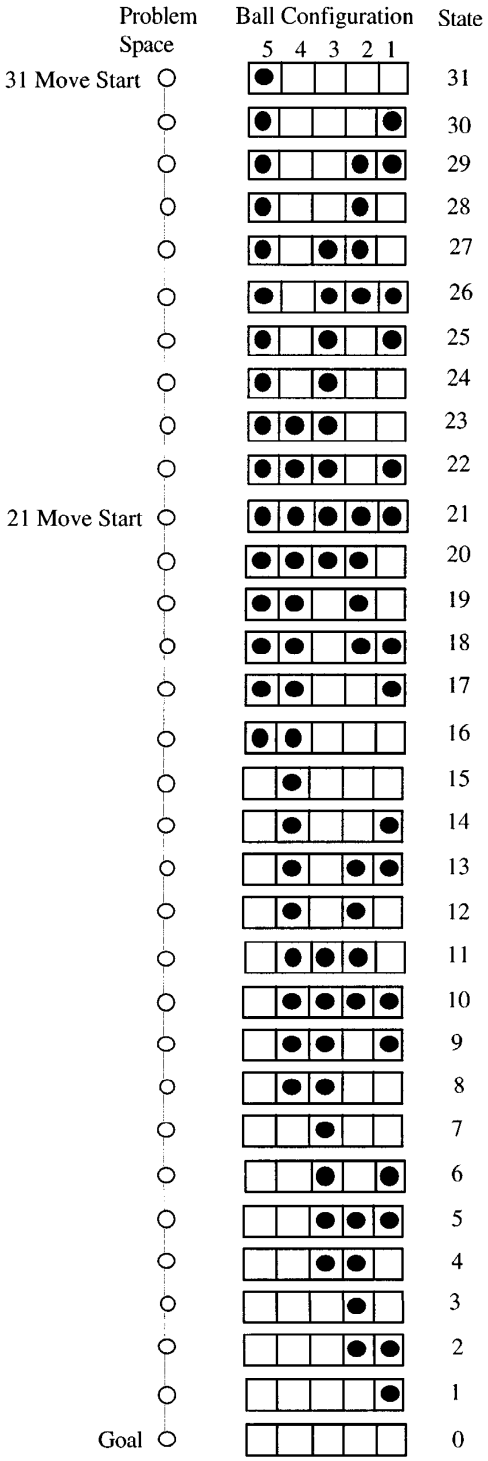
Typical solution paths (from state to state) obtained from human subjects solving the Balls and Boxes problem are illustrated in Figure 5. In that figure we present a set of move records for human subjects solving the puzzle. As can be seen in that figure, the subjects typically make a large number of moves with little or no net progress toward the goal, and then suddenly move rapidly and accurately to the goal. This dichotomous behavior has been labeled “exploratory” and “final path” (Kotovsky & Simon 1990). Interestingly, the seemingly insightful transition to the final path generally is not accompanied by significant verbalizable knowledge or true insight into the problem, and if given the same problem again, subjects again resort to problem solving including another exploratory and final path period. Learning is evidenced by a shortened exploratory period, but nothing approximating total understanding is evidenced on the second solution attempt (Reber & Kotovsky 1992; Kotovsky & Simon 1990). Another finding is that the linearity of the search space is not discovered by the subjects (even when they solve the puzzle multiple times) and thus the problem is not trivialized by that feature.

A similarly dichotomous exploratory/final path solution process has been found for subjects solving some difficult isomorphs of the shorter three-disk Tower of Hanoi Problem, with people typically solving the problem by wandering in the problem space for some time and then traversing the entire solution path length and solving the problem in the last 15% of the time (Kotovsky, Hayes, & Simon 1985). However, in the Tower of Hanoi problem, there are additional mechanisms that are likely to lead to the final path behavior beyond those found in the Balls and Boxes problem. The Balls and Boxes problem is characterized by subjects’ inability to verbalize or even recognize strategic information, even after successful solution, while the Tower of Hanoi problem isomorphs yield much more verbalizable information about higher-level strategic knowledge such as move planning and subgoaling (Kotovsky et al. 1985; Kotovsky & Simon 1990; Reber 1993). The implicit or nonconscious nature of the learning that occurs in the Balls and Boxes problem that allows movement onto the final path within a problem and faster solution of a second problem is an important characteristic of the problem and is particularly well suited to the type of model we propose. The additional, higher level processes found in the Tower of Hanoi are not the focus of the current modeling effort, although any implicit learning that occurs there (or in other problems—see Broadbent & Berry 1988) is.

Major empirical results include the following (Kotovsky et al. 1985; Kotovsky & Simon 1990; Reber 1993):

- Despite the simple linear structure of the problem search space, the problems can be very difficult. The Balls and Boxes problems often took hundreds of moves to solve. The Chinese Ring isomorph was even more difficult. In one study, only 7% of the subjects were able to solve the problem in 1.5 hours.

[6] In general, for an n-ring problem, there are 2^n possible states in the search space, with a resultant minimum solution path of $2^n - 1$ moves. There are a number of similarities between this problem and the Tower of Hanoi problem (Section 2). Both problems are infinitely expandable. In both problems, the most restricted piece (ring or disk) is moved half as frequently as the next most restricted, which is moved half as frequently as the next, and so on. Finally, the minimum solution path length is the same as in the n-disk Tower of Hanoi problem, although the size of the search space is not.



Problem Space	Ball Configuration 5	4	3	2	1	State
31 Move Start ○	●					31
○	●				●	30
○	●			●	●	29
○	●			●		28
○	●		●	●		27
○	●		●	●	●	26
○	●		●		●	25
○	●		●			24
○	●	●	●			23
○	●	●	●		●	22
21 Move Start ○	●	●	●	●	●	21
○	●	●	●	●		20
○	●	●		●		19
○	●	●		●	●	18
○	●	●			●	17
○	●	●				16
○		●				15
○		●			●	14
○		●		●	●	13
○		●		●		12
○		●	●	●		11
○		●	●	●	●	10
○		●	●		●	9
○		●	●			8
○			●			7
○			●		●	6
○			●	●	●	5
○			●	●		4
○				●		3
○				●	●	2
○					●	1
Goal ○						0

FIGURE 4. Balls and Boxes five-ball problem problem space. The linearity of the space as well as the 21-move starting position and goal are illustrated.

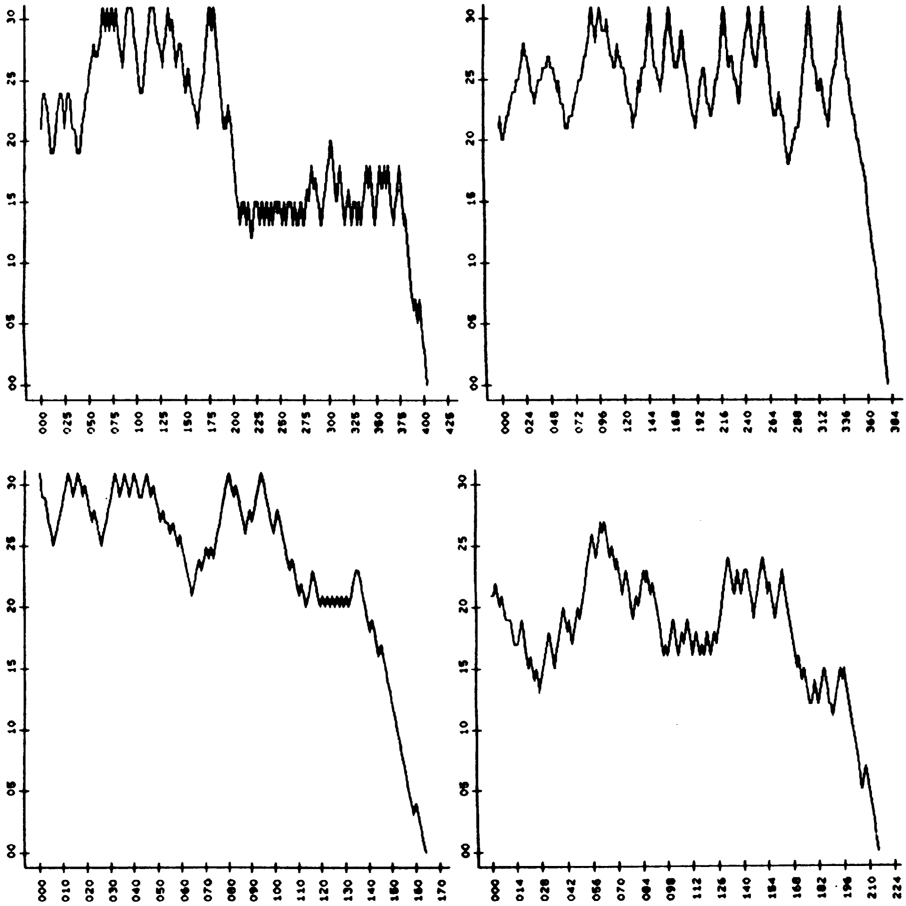


FIGURE 5. Balls and Boxes problem typical human move records. The distance from the goal is indicated on the ordinate and the move number on the abscissa (from Kotovsky & Simon 1990).

- The behavior of people solving the puzzle exhibits two phases, a long “exploratory” period in which subjects make many moves with no net progress toward the goal, followed by a rapid “final path” dash to a solution during which they move directly and unerringly to the goal. The exploratory behavior is characterized by frequent reversals of direction, a great deal of revisiting of previous visited states, minimal progress toward the goal, and, where the problem solver is not constrained from making them, a great number of illegal (problem rule-violating) moves.
- While the transition from exploratory to final path behavior seems to indicate a sudden insight into the solution, subjects are able to report very little understanding of the problem and are not able to articulate any of the rules that describe the problem or the structure of the problem space.
- Although the subjects are not able to report on or articulate the rules that describe the problem or state structure, when they solve the problem from the same initial state again, they solve it with the same characteristic move patterns but in fewer steps due to a shortened exploratory period.
- Isomorphs of the Tower of Hanoi problem that involve monsters passing globes back

and forth or changing their sizes (rather than disks moving from peg to peg as in the Tower of Hanoi) exhibit a similar exploratory–final path dichotomy. Subjects typically solve Monster problem isomorphs (traversing the total distance from start to goal after much nonprogress making "exploratory" behavior) in the final minute or two of a ten- to fifteen-minute solution attempt.

Note that the exploratory phase can be viewed as almost random search. However, the prototypical abrupt change to the final path indicates that some learning has occurred that allows efficacious move-making. We propose that this happens through subjects learning pairwise move contingencies during the exploratory period in the Balls and Boxes Puzzle.[7] In addition, there is a further decrease in the randomness of the move choices that possibly results from a decreased reliance on the downhill search heuristic which leads them to prefer moves that remove balls from their boxes (Reber & Kotovsky 1992). The subjects thus become less stochastic in their approach to solving the problem, eventually achieving an error-free and unwavering final path to the goal. Finally, on subsequent solutions of the problem, although this learning allows them to eventually perform in a less random manner and thus more rapidly solve the problem, they do not start out that way, but rather exhibit another exploratory period, albeit a shortened one, a somewhat surprising result given the evidence of understanding provided by the preceding final path. Examples of the speedup from the first solution of a problem to the second solution of the same problem range from 420/205 (first solution/second solution) at the high end to 116/49 at the low end depending on the difficulty of the particular problem isomorph used (Kotovsky & Simon 1990; Reber 1993).

5. A MODEL OF PROBLEM SOLVING

We propose a computational model of problem solving based on the preceding observations. The model has the following characteristics: It is given the basic node-link structure of the problem space with minimal information about the relation of nodes in that space to the goal. Search within that space begins stochastically, becoming deterministic over time. When the model moves between two states, it updates the objective function values of those states to reflect the acquired knowledge that each is easily reachable from the other and thus must have similar values. When the problem is re-solved, the updated understanding of the objective function space is used and search progresses in a less stochastic manner. These operating characteristics of the model are broadly based on the behavior exhibited by human subjects, in particular the fact that over time the search becomes less exploratory and moves to final path—i.e., that people start out moving more or less randomly (although they might not feel that they are moving randomly) but after some time become much more directed toward the goal. The learning of pairwise contiguity relations (here represented by moving the energy levels toward each other when a move occurs between the two states) is based on the plausible assumption that people can learn that two states are close to each other when it is possible to move from one to the other in one move. Note again that this contiguity is bidirectional; both state values are adjusted rather than just the one from which the algorithm moves as is typically done in machine learning.

[7] In Monster problem isomorphs of the Tower of Hanoi, an additional mechanism comes into play that involves the working memory load imposed by surface representational features of the different isomorphs, and how they impact on planning pairs of moves. These mental model representational features that account for difficulty differences among the various isomorphs are not the focus of the current model, which operates by evolving a representation of the underlying structure of the problem space.

Our model uses the stochastic optimization technique of simulated annealing as a method of exploration. Simulated annealing is a zero-order optimization technique that begins in an essentially random manner but quickly becomes more deterministic while still maintaining some stochastic characteristics. The model uses simulated annealing to search for the goal in the problem space by moving between connected states. At the start of a model run the states are assigned values based on estimates of the a priori knowledge that humans solving the same problem might be expected to have. Thus each such state is initially described as being at one of three levels of certainty: “known”, “assumed”, or “unknown”. A known state has a known and fixed evaluation value; an assumed state has a weakly determined value that is adjustable; an unknown state has an unknown value (which is arbitrarily set to be relatively high and is readily adjustable). Unless there is some plausible reason to assume that a human would have a priori knowledge about a state, it is set at the default initial value of “unknown”. As the model moves between states, the values of the two states move toward each other, indicating that the two states are easily reachable from each other and thus must have evaluations near to each other. The relative amount of movement of the two states is based on their relative certainty level. Exhaustive search of the entire space is not desirable; however, if the search finds a state not previously visited then an unexplored region is indicated in which it may be desirable to search. This increase in uncertainty that results from being in an unfamiliar region is accomplished by a transitory increase in annealing temperature. After the goal is found, the program is restarted at the same initial point with the updated values of the objective function at each state maintained. Additionally, the rate of reduction in the annealing temperature (as discussed below) is accelerated, thereby reducing the randomness in the search of the now more familiar space. Each of the model’s mechanisms is now more fully described.

5.1. Simulated Annealing

Introduced by Kirkpatrick et al. (1983), simulated annealing is a stochastic numerical optimization technique used to solve continuous, ordered discrete and multimodal optimization problems. The algorithm selects an initial state, which is evaluated through the objective function. In an annealing algorithm the result of the evaluation is known as the *energy*. A state is selected within some predefined neighborhood, here defined as the set of nodes linked to the current state, and it is evaluated. The energy of the new state is compared to that of the original state. If it is better in its evaluation then it is selected as the current state (a “move” occurs); if it is worse then there is still a probability that it will be selected as the current state. The probability of accepting a worse state is initially high (emulating random search) and progressively decreases to zero (emulating downhill search). This probability, Pr{new}, takes the form:

$$\Pr\{\text{new}\} = \frac{e^{-\frac{E_{\text{new}}-E_{\text{current}}}{T}}}{e^{-1}},$$

where E_{new} and E_{current} are the new and current evaluations (energies), T is a variable called *temperature*, and the denominator (e^{-1}) is used to normalize the initial probability to be 1.0.

The variable T (temperature) is defined so that Pr{new} is initially 1.0 and decreases after a large number of moves to 0.0, unless the problem is solved first. The decrease in temperature is laid out in a function called the annealing schedule. Simulated annealing is often used to solve problems where 100,000 or more iterations are required; sophisticated self-adaptive annealing schedules can be defined which adjust themselves based on a statistical analysis of their behavior (e.g., Huang, Romeo, & Sangiovanni-Vincentelli 1986). In our application between 15 and 1,000 moves are all that is required and reasonable and so a fixed “natural” or vanilla schedule is used: Here the temperature is multiplied by a constant reduction factor that is less than, but close to, 1.0 at each temperature reduction. In our implementation, the temperature is reduced at every iteration. Thus the temperature at any iteration, i, is defined as:

$$T = T_{\text{initial}} * (\text{reduction\_factor})^{i},$$

where T_{initial} is the temperature at the start of the run. In this implementation, the initial temperature is set to 1.0 and the reduction factor is initially set to .99 unless otherwise stated.

The state space in the current implementation is represented by a connected graph where each node represents a feasible configuration of balls in boxes or disks on pegs and the links between nodes represent the legal moves. Such a move between any two nodes consists of moving one ball or one disk. A random move is selected from the available links and the move is taken based on the annealing criteria.

Note the characteristic of simulated annealing: the search starts with a large random component and then progressively becomes downhill search. Thus, wide exploration of the search space becomes focused into a local region where the solution is found. These same characteristics are seen in the solution paths from humans while solving the Balls and Boxes and Tower of Hanoi problems where the move-making starts out seemingly random and progresses into a delimited pathway. Our model attempts to capture this characteristic human behavior in two ways: one is simply the annealing temperature reduction and the other is the progressive smoothing of the search space as the relative proximity and values of nodes are learned. The learning in our model is due to the combination of these two mechanisms, fairly random behavior (a wide range of acceptance of candidate moves) early in problem solving due to the high annealing temperature, coupled with a decrease in temperature as more is learned about the structure of the problem space through alterations of the energy levels as moves are made. Both of these mechanisms represent reasonable assumptions about how humans may modify their behavior as they progress in a problem-solving episode.

5.2. Evaluation Adjustments (Learning about the Problem Space)

The exploration-learning capability of the model assumes three types of values for the objective function: known values, assumed values, and unknown values. This trifurcation was chosen to represent the likely variation that would be expected to exist in a human problem solver's level of certainty about different loci in the problem space. Although it is possible that people have an even more highly graded set of certitude levels, the above partitioning is a reasonable first assumption. Each state has associated with it an initial estimation of the value of the state. We assume that a problem solver typically knows he or she is starting at some distance from the goal; therefore the starting state (as well as most others) have a relatively high initial energy value, while that of the goal, which we assume is recognizable by the problem solver, has a very low energy value. In addition there might be a small number of "landmark" states between the start and the goal (such as a state known to be close to the goal) where the solver can be expected to know (or believe he or she knows on the basis of some general heuristic) either the approximate or exact value. This knowledge is instantiated by assigning an energy to each state and marking the state as "known", "assumed", or "unknown", depending on the level of certainty about its value, i.e., its energy. Known information is exact and does not change when that state is encountered; assumed and unknown values have increasing flexibility in the amount that they can change. An example of such a landmark state initial value assignment is implemented in one of the problems investigated, one version of the four-disk Tower of Hanoi problem. In that problem the only initial intermediate problem space energy function knowledge is that it is "good" to get the largest disk on the goal peg. The value of that state is initially set as "assumed"; all other states are set at "unknown" except the goal state which is always set at "known".

TABLE 1. Transition Matrix Representation of Learning (alteration of energy levels) When a Move is Made between Two States; Ratios Indicate Source/Destination.

		Destination		
		Known	Assumed	Unknown
Source	Known	0/0	0/.8	0/.9
	Assumed	.8/0	.25/.25	.1/.8
	Unknown	.9/0	.8/.1	0/0

The model learns about the space as it moves between states, changing both the value and classification of the states it encounters as well as the value of the states it moves from. In this learning, known states are more powerful than assumed states, which in turn are more powerful than unknown states. What this means is that the certitude of the value assigned to a state grants that state the power to modify other states and itself resist modification. Thus if there is a move from a known state to an unknown or assumed state, it modifies the value of the target state to bring it closer to the value of the known state. The complement is also true; i.e., a move from an assumed or unknown state to a known state moves the assumed or unknown state toward the known state. Finally, if a move is made from an assumed or unknown state to another assumed or unknown state, it not only modifies the value of the target state, but also modifies the value of the state it moves from. The assumption here is that the model learns that the two states are contiguous by moving from one to the other, and thus links them with similar energy values, given that it is easy (one move) to get from one to the other. In this way the model learns about the search space, representing its knowledge as a set of energy values associated with the states it has visited from previously known or assumed states. It thus progressively expands its knowledge of the space outward from states whose value it knows (partially or wholly) to states it encounters during the solution of the problem. This learning of pairwise relations is well documented in human behavior (Cohen, Ivry, & Keele 1990).

Again, the power of a state to change another contiguous state depends on the level of certainty of its knowledge of the energy function values for the states the model moves to and from. The difference between the energies associated with the source and destination states is reduced by a fractional amount, determined by the level of knowledge of the two states, according to the values given in Table 1. In this table the pair of numbers indicates what percentage the source (top) moves toward the destination and what percentage (of the energy difference separating them) the bottom (destination) moves toward the source. Thus, an assumed state moving to an unknown state moves the assumed source 10% closer to the unknown destination and the unknown destination moves 80% closer to the assumed source.

Initial unknown states are set to a high value, the premise being that start states are likely to be far from the goal. Once an unknown state is visited from a known or assumed state and its value adjusted, its classification is changed to "assumed". This is because there is now some information about its value, albeit incomplete or inexact information. The differences between adjacent states' values are thus reduced as the model makes moves. Note that the actual numbers used in the table are somewhat arbitrary; what is important is the relative certitude of the information for the model (or, correspondingly, for the human problem solver) indicated by the relative magnitudes of the values in the table.

Again note that the policy (i.e., learning method) used in this model contrasts with that typically used in machine learning. Here, the values of both the current and new states are modified. As well, rather than adjusting the values by some a priori quantity, the states are adjusted by a percentage based on the type of states involved.

This modification of the objective function within a simulated annealing framework is a significant departure from typical annealing. Typically, a fixed objective function is annealed to determine its minimum value; here the objective function itself is constantly changing to represent knowledge acquired during solution of the problem.

5.3. Finding New States

Given that this model uses simulated annealing rather than exhaustive search, we have included a mechanism that encourages the exploration of previously unvisited portions of the problem space when and if they are encountered. If a previously unvisited state is encountered and has a lower energy value than the source state, the annealing algorithm will move to the new state. If the new state evaluates to a higher value than the source state, then the annealing probability, Pr{new}, dictates whether or not to move to the new state. As the temperature decreases, Pr{new} will decrease, making it less likely that the model will move to the new state, thus mitigating against the exploration of new regions. To give the algorithm the opportunity to explore these new regions, if the probability is low the annealing temperature is temporarily raised to moderately increase the probability of exploring the new region. In particular if the probability of accepting the unvisited state is less than some threshold (in our implementation that threshold is 25%), then, based on the probability function, the temperature is increased so as to give a higher probability (50% in our implementation) of accepting the unvisited but higher energy state. In this implementation, when a state is visited, a record of that visit is maintained in the structure for the node (i.e., in human terms, it is thereafter recognized that it is familiar or previously visited). As the iterations continue the temperature continues to drop by being multiplied by the reduction factor. This "energy kick" mechanism increases the chance of the model engaging in the type of opportunistic behavior used by humans by exploring previously unvisited areas of the search space without necessarily engaging in exhaustive search of that space.

5.4. Accelerated Annealing (Increasingly "Confident" Move-Making)

As the model learns more about the search space and smoothes the energy function, it is less likely to be caught in a local minimum and can thus anneal faster; this is analogous to the person learning more about the space and moving more confidently (rapidly) to solve the problem faster on successive iterations (Kotovsky & Simon 1990; Reber 1993). We implement this concept by accelerating the annealing schedule. In particular, after each trial[8] the reduction factor is numerically reduced by a constant amount, thereby increasing the rate of annealing. We call this *accelerating* the annealing process. In this implementation, the acceleration factor is conservatively set to .001 for the first 24 iterations after which the energy function is assumed to be sufficiently smoothed. Then the rate of annealing is significantly increased by setting the acceleration factor to a default value of .010 for the remaining 16 iterations. This means that if 40 trials are run and the initial reduction factor is .99, then in the next trial it will be .989, accelerating the annealer by .001; the next trial will be .988, and so on until the last iteration (of the 40) where, after the cumulation of 24 accelerations of .001 and 16 of .010, the decrement is .806, or $.99 - (24 \times .001 + 16 \times .010)$.

The accelerations that we employ in each of the examples and the transition state after trial 24 is a conservative model for the system. Many runs could have a faster acceleration from

[8] A trial is considered a completely annealed solution attempt after which the problem is reset to the initial start state while maintaining the modified energy values.

the start and a much earlier transition state to the more aggressive acceleration. However, the numbers were chosen here to be applicable across all of the examples and illustrate the power of the acceleration even with this conservative testbed. When we use the model with more aggressive numbers on some of the less difficult isomorphs, the model solved the problem in fewer moves, much closer to those representative of human subjects.

6. APPLICATION OF THE MODEL TO PROBLEM EXAMPLES

6.1. Initializing Assumptions

This section describes the results of applying the model to two exemplary problems, the four-disk Tower of Hanoi and five-ball Balls and Boxes Puzzles, in order to determine how it would perform. In each of these applications the initial conditions were as follows: The model operated in a problem space that had an initial energy associated with each node in the space. Within each problem type a varied set of isomorphs were investigated. At one extreme, problems had all initial states except the goal state classified as unknown with identically high energy values. In other problems, a small number of other states were given values on the basis of either downhill search-based inferences about differences in the value of particular nodes, or the kind of general preexisting knowledge that could reasonably be assumed to be held by a human problem solver, without special knowledge of the puzzle. This latter knowledge was treated as assumed, rather than known, and thus could be modified. Although often helpful, these assumptions could also be harmful in those cases where downhill search must be violated, such as in detour problems (which applies to both of the problems used here) where one must move away from the goal in order to get to it. Another extreme type of problem was one wherein all states except the goal state were given values predicated on downhill search-based estimates of their distance from the goal. In all cases the goal was classified as "known" with energy value zero, thus instantiating the assumption that the solver could recognize the goal state.

In the discussion below, we present the results of the model's exploration of the space in which it solved the same problem 40 times (i.e., 40 trials or iterations), learning more and more about the search space as it did so. This large number of repeated solutions or solution attempts was chosen in order to allow for a thorough analysis of the model's performance, including the eventual total smoothing of the problem space, and not because it was necessarily a realistic number of solution attempts for a human subject to try. The performance of the model was compared with those aspects of human performance we wished to focus upon. These include in particular the viability of the basic incremental learning mechanism, and whether it can overcome incomplete, or even misleading, initial assumptions, as well as the transition after reasonable amounts of problem exploration from exploratory to final path behavior. The model started each time (on each iteration) with the previously learned knowledge about state values retained. In order to reduce the variance and obtain a stable estimate of each result, each experiment was repeated 50 times (corresponding to an experiment using 50 subjects), and the curves plotted each represent the average and standard error of 50 runs (with each run comprising 40 learning trials). In addition, a typical run is identified to illustrate details of how the algorithm behaves in any one run.

6.2. Four-Disk Tower of Hanoi Puzzle—Problem Isomorphs

We examine first the performance of the model on the four-disk Tower of Hanoi problem. The problem space of that problem is depicted in Figure 2. As can be seen in that figure, there are a total of 81 different knowledge states in the problem space. The model is started at state number 1, which corresponds to all the disks being on the leftmost peg, and the goal the model is given is to move to state 81, which corresponds to all the disks being on the rightmost peg. Moving from state to state corresponds to following the standard rules for that puzzle which restrict moves to (1) moving one disk at a time, (2) never placing a larger disk on a smaller one, and (3) moving only the topmost (smallest) disk from the disks stacked on any peg. There were four different versions of the problem, each differing by the amount of preexisting knowledge the program was given about the problem. The four problems and their initial energy level configurations are listed in Table 2. In all cases, the goal state energy level was set at zero. The problem types defined in Table 2 are listed in order of increasing amounts of useful information that are initially provided to the model via the energy levels.

We define another feature of the problem space that is used in some of the learning analyses to be reported below. This is the “primary path”, which is the most direct path of 15 moves from start to goal, i.e., the pathway down the left side of the largest triangle in Figure 2. Examination of these adjacent states allows us to observe the relative “smoothing” of the energy function as the model solves the problem. This smoothing is due to the transitions from one state to another pulling the energy values of the adjacent states together as the model attempts to traverse between adjacent states.

6.2.1. Misleading Problem. The first problem we consider is the four-disk Tower of Hanoi problem, with misleading energies initially supplied; i.e., a set of energies that corresponds to a “naive” problem solver’s assumption that the more disks that are on the goal peg the better, and with assigned energy levels corresponding to this downhill search strategy. The initial and subsequent energy levels are depicted in Figure 6 for the initial assignments, and the 1st, 15th, and 40th learning trials. As previously described, the learning (modification of the energy levels assigned to different positions in the problem space) cumulates over the 40 iterations. The points are the averages obtained from 50 repetitions (runs) of the experiment, where the energy values are reset after each run (but not within the 40 iterations or trials that constitute a single run). Unless otherwise defined, this general approach to trials and repetitions holds over all the work described below. In the “misleading” case, the initial energy levels for each run are set at a value equal to four minus the number of disks on the goal peg, regardless of whether they are there in correct order or not. The initial energy levels thus range over the integer values 0 to 4, as depicted in Figure 6.[9]

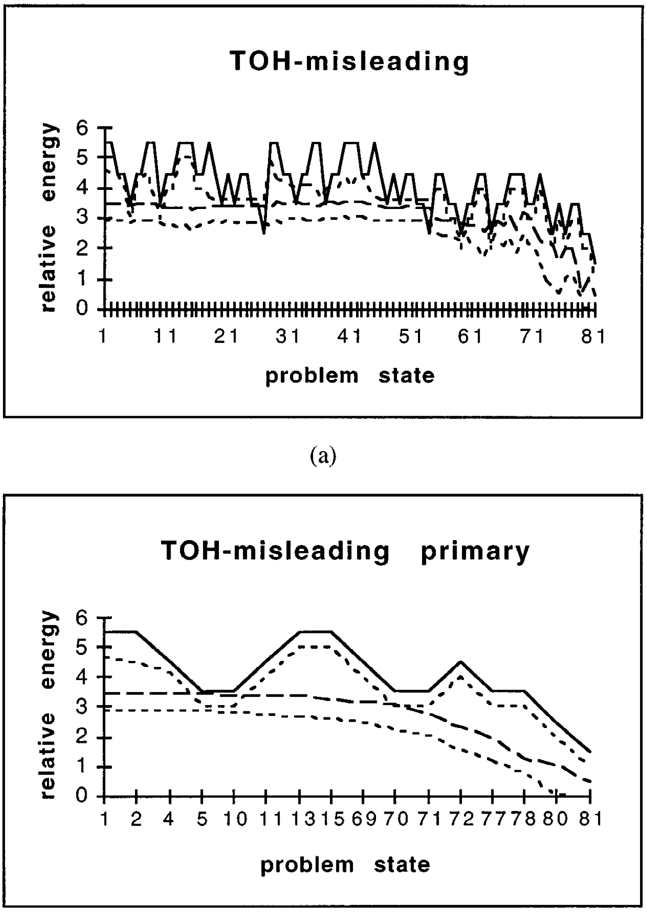
The function describing the energy values becomes progressively smoother as the model moves through the space, searching for the goal and adjusting the energy values as previously described in Table 1. Figure 6 shows this progressive smoothing for all the states in the problem space (Figure 6a) and for the set of adjacent states that represent the most direct path from the start to the goal (Figure 6b) for a typical run. The “misleading” nature of the initial information is most easily seen in Figure 6b where the initial energy values show states 5 and 10 and also states 70 and 71 to be at lower energy levels than most of the remaining states on the way to the goal, creating local minima. As the figure shows, the energy function smoothes out over trials, with some smoothing having taken place during the first trial, a great deal by the fifteenth, and the process having progressed almost to completion by the fortieth. This progressive smoothing is a demonstration of the significant learning that the model is accomplishing.

[9] In this and subsequent figures describing relative energy values, the energy curves have been offset from each other for illustrative purposes. In particular, the curve of the initial energy assignments is increased by a value of 1.5, while that of the first iteration is increased by 1.0, that of the 15th iteration by .5, and that of the 40th iteration shows exact values. This treatment thus creates a family of curves for ease of depiction, rather than a set of overlaid and thus hard-to-decipher curves.

TABLE 2. Types of Four-Disk Tower of Hanoi Problems.

Problem	Energy levels configuration
Misleading	All nodes set at an energy level (0 to 4) corresponding to the number of disks on the goal peg, with increasing numbers of disks being equated to smaller energy levels (i.e., energy = 4 − number of disks on the goal peg). This is often misleading in that putting a wrong disk on the goal peg, even though it decreases the energy level, increases the distance to the goal rather than reducing it. These initial assumptions are those that might be expected from a naive problem solver who initially follows a downhill search heuristic that assumes that moving any disk to the goal is desirable.
No Information	All nodes at the same energy level (4), except the goal. This corresponds to the problem solver initially recognizing only when it reaches the goal state.
Large disk	All states corresponding to configurations wherein the largest disk is on the goal peg are given lower energy levels (3), with all other states except the goal state at energy level 4. These states (55 through 81) comprise the bottom or left triangle in Figure 2. This corresponds to a problem solver assuming that getting the most restricted disk to the goal is good. It thus incorporates that single subgoal in the initial energy assignments.
First three disks	This problem assigns an energy level to all states based on the number of disks correctly placed on the goal peg. These energies take on the integer values from 0 to 4 (i.e., energy = 4 − number of disks on the goal peg in *correct order*). This corresponds to a problem solver starting with a good assessment of the problem in the form of three useful subgoals.

The progressive smoothing of the energy function with experience is not as readily ascertainable from Figure 6a as it is from Figure 6b because the two-dimensional shape of the problem space mandates that in any linear portrayal there are included some nonadjacent states that result from the mapping of a two-dimensional array onto a single dimension. For example, in Figure 2, see the leftmost sequence that, as the solver approaches the goal, moves from state 15 to 69 or from states 70–72 and then to state 77. As well, states 27 and 28, and 54 and 55, are eight steps apart. We would not expect nonadjacent, though numerically consecutive, states to attain similar energy function values over time, and, as seen in Figure 6, they don't.



(a)

(b)

FIGURE 6. Tower of Hanoi Misleading problem isomorph: Energy function for the initial conditions (solid line) and iterations 1 (top short dash), 15 (long dash), and 40 (bottom short dash). (a) Energy for each node in the problem space. (b) Energy values for only those nodes on the primary path. Note that the energy values for the different iterations are displaced by 0.5 each for purposes of display.

Figure 7 displays the number of moves (state to state transitions) made by the model in solving the problem. The mean number of moves (for the set of 50 runs) is plotted vs. the trial number. The shading indicates confidence limits, ±2 times the standard error of the mean. The "standard" annealing schedule used in most of the work reported here was, as described earlier, to start the annealing at .99, and decrease that multiplier by .001 on each of the first 24 iterations, and then by .010 on each of the remaining 16 iterations. In Figure 7a we depict the number of moves until the model stops and in 7b the number of moves to a solution.[10] The difference is that 7a includes instances where the model does not solve the problem but stops because the annealing temperature reaches zero. It thus depicts a mixture of the number of moves taken by the model to solve the problem and the number taken until

[10] Note that the ordinate scales for the moves to solution and moves to stop figures may differ.

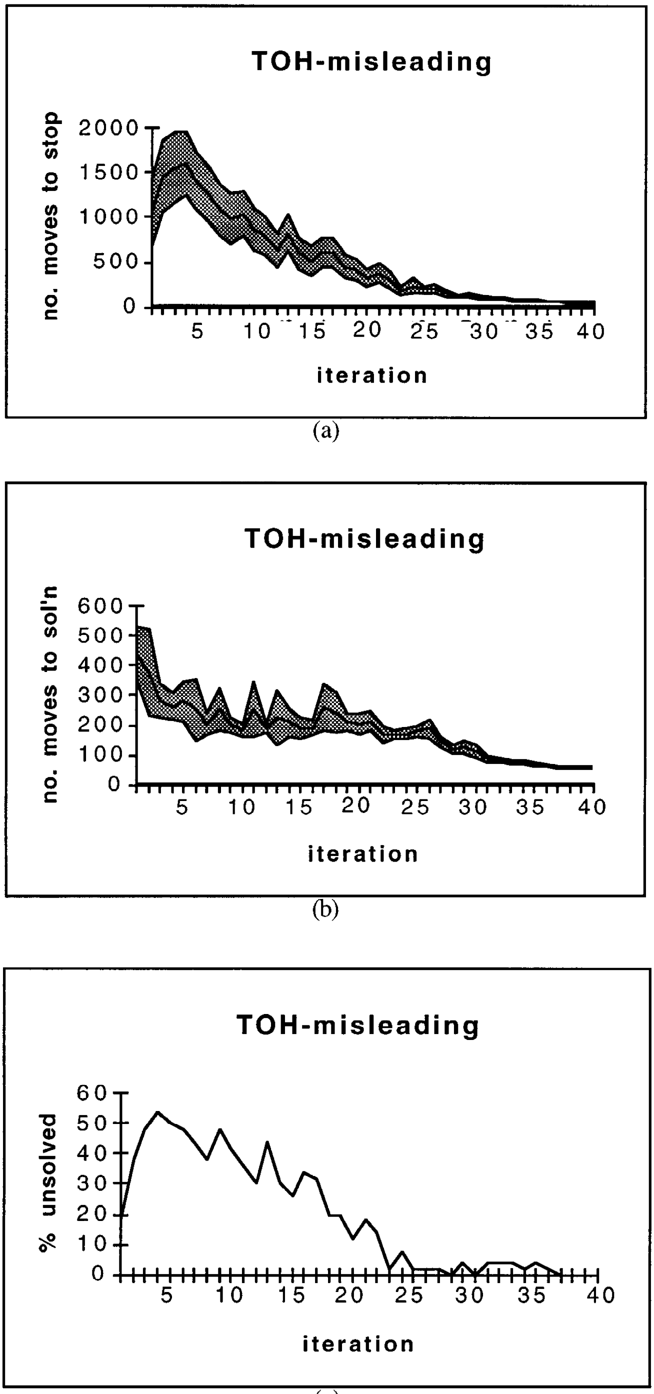
the annealing temperature reaches zero. This failure to solve is unique to this problem (due to the misleading nature of the initially provided information which causes local optima). In Figure 7c we depict the proportion of the time the model does not solve the problem (the contribution to 7a of nonsolved cases) which is seen to decrease to zero over time. Inspection of Figure 7b shows that on runs where the model does solve the problem it does so more rapidly with experience. Figure 7c shows that the proportion of times the model solves the problem increases dramatically over iterations. The model thus does indeed learn to solve the problem more quickly and efficaciously with experience. The average number of moves to completion is initially greater than 1000 (and a significant number of attempts fail) and with experience this decreases to less than 60 moves with no failures.

Note in Figures 7a and 7c the "bump" during the early iterations indicating that the model starts reasonably well but in the early iterations becomes somewhat less consistent in solving the problem, taking more moves before it stops as compared to its performance on the initial and later iterations. We believe that as the model modifies its objective function values in the early iterations, it spreads out the local minima while also accelerating the rate of annealing. The conjunction of these events makes it more likely that the model gets caught in these broader local minima, and thus increases the number of nonsolutions. As the model continues to learn, it smoothes out these minima, allowing it to consistently solve the problem.[11]

The data of Figures 6 and 7 together demonstrate the learning that occurs during problem solution. Figure 6 shows the changes in the energy function that result from the model's learning what states are near what other states and adjusting the objective function as it makes moves. Figure 7 shows the result of these changes together with the effect of accelerating the annealing schedule on the number of moves taken in subsequent trials. These two sources (learning and annealing acceleration) will be disentangled below. The fact that the initially misleading objective energy function does not prevent the model from learning is the first major result of this modeling work, and serves as proof that the model can effectively learn about the problem space even given the obstacle of initially being "fed" inaccurate information about the goodness of various states.

Note that although the model starts out requiring more than 400 moves to solve the problem (or more than 1000 moves to stop if we include nonsolutions), this number rapidly decreases over iterations (Figure 7b), although it continues to fail to solve a significant proportion of the time until the 25th iteration. On this "misleading" problem, we would have to conclude that the model is considerably less powerful a problem solver than are humans. The misleading information (based on a naive downhill-search heuristic) provides a significant impediment to the model. It has been suggested (Reber & Kotovsky 1992) that humans give up or modify that heuristic earlier in their problem-solving episode, thus escaping its major consequences. This difference between the performance of the model and of people in this instance requires further examination. In an initial exploration we combined the energy values from the misleading and first three disks isomorphs. This in effect creates a model of a solver who combines a naive downhill search heuristic with some recognition that *correctly* placed disks should not be removed. The results demonstrated significant improvement in the algorithm's ability to solve this problem over that from the misleading isomorph alone. Further research may indicate that a similar combination of heuristics or transition between heuristics occurs in humans.

[11] Further support for this explanation can be found in the results of a subsequent experiment (Figure 17b) where the annealing was not accelerated. The number of moves to stop dropped more rapidly after the bump, indicating a quicker recovery from the local minima.



(a)

(b)

(c)

FIGURE 7. Tower of Hanoi Misleading problem isomorph. Number of moves vs. iteration: (a) number of moves until the model stops due to either success (i.e., finding a solution) or due to having the annealer stop because the temperature had decreased to near zero, (b) number of moves to a solution for successful trials only, and (c) the percentage of trials that were unsuccessful at each iteration.

6.2.2. No Information. The second condition of the four-disk Tower of Hanoi problem is the no-information condition. In this condition, all of the energies are set at the same level (4) with the exception of the goal state which is set at 0. This represents the condition where subjects are assumed to know nothing about the problem space except for being able to recognize when they have reached the goal. The results are plotted in Figures 8 and 9. This lack of initial knowledge is depicted in the initial energy[12] configuration of Figure 8 (iteration 0), which shows all states at energy level 4 with the sole exception of the goal state which, as always, is at energy level 0. Figure 8 shows that there is a rapid smoothing of the energy function as the model learns via the exploration that occurs in the early iterations. This is most easily seen in Figure 8b, which shows the energy levels on the direct path to the goal, whereas Figure 8a shows the energy levels for the entire space. As before, the energy levels are shown for a sampling of the iterations: the initial energies and the 1st, 15th, and 40th iterations. We conclude from this smoothing that the model is rapidly learning from its experience in moving between states, representing this learned knowledge in the form of altered energy levels.

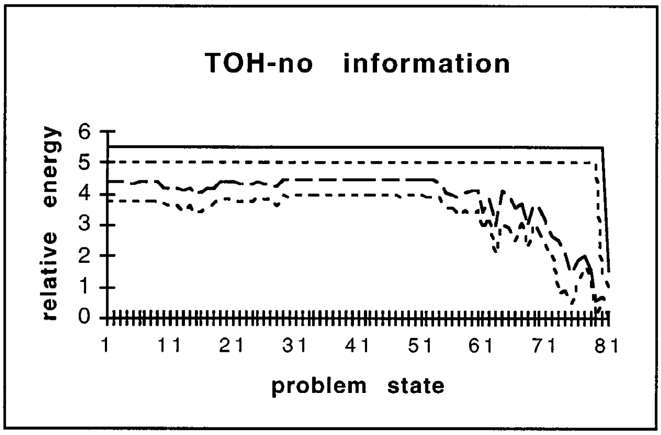
The effectiveness of the learning in supporting a solution to the problem is depicted in Figure 9, which shows the average (for 50 runs) number of moves to solve the problem on each of the 40 iterations that comprise each run.[13] As before, the learning (alteration in energy levels) that has been acquired is cumulated across the 40 iterations. In addition, the annealing schedule is accelerated as it was before, starting at a value of .99 and accelerating by .001 on each of the first 24 iterations, and by .010 on each of the remaining iterations. The learning in this version of the problem is much more rapid even in the earliest iterations than in the misleading problem, and the smaller standard error of the mean shows the relative uniformity of the model's behavior on different solution attempts, in comparison to the previous problem. In addition, the model was able to consistently solve the problem on all iterations of each, unlike the previous case where there was a relatively high initial failure-to-solve rate.

6.2.3. Large Disk Problem. We next explored a problem where the model is given a significant subgoal indicating proximity to the goal. The Large Disk problem includes the initial knowledge that states in which the large disk is on the goal peg are desirable, i.e., has lower energy levels. This represents useful information that is also realistic in that a problem solver could be expected to recognize the importance of this landmark near the start of the problem. The results of providing this knowledge are discernible in Figures 10 and 11, where the typical energy function and average moves to solution are presented. While the energy function (Figures 10a and 10b) shows the usual learning about the problem space, the moves to solution (Figure 11) show the usefulness of the information that was provided. The problem was relatively easy to begin with (approximately 500 moves on the first iteration) and the difficulty decreased to approximately 50 moves, with a relatively small standard error.

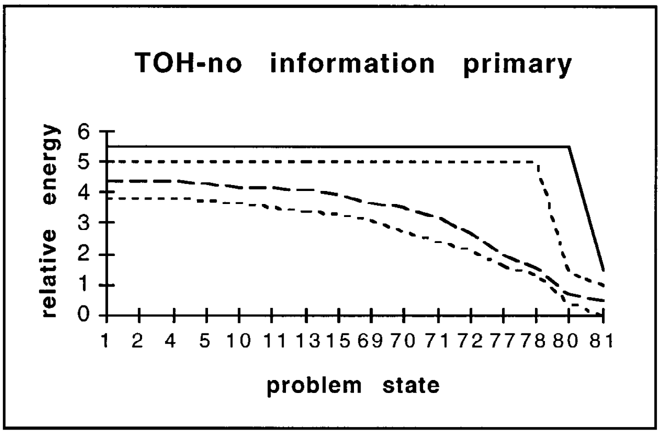
6.2.4. First Three Disks Problem. Given the impact of the information about a single landmark—the correct placement of the large disk—we increased the amount of information we initially provided the model in the First Three Disks problem, indicating all major landmarks. We did this by informing the model about the correct placement of each of the three largest disks on the goal peg via the energy assignments described in Table 2. The

[12] The initial energy value actually reads 5.5 instead of 4 in Figure 8 because it is offset by 1.5 for diaplay purposes, as described in note 9.

[13] Again, shading in this and subsequent graphs represents ±2 times the standard error of the mean.



(a)



(b)

FIGURE 8. Tower of Hanoi No Information problem isomorph: Energy function for the initial conditions (solid line) and iterations 1 (top short dash), 15 (long dash), and 40 (bottom short dash). (a) Energy for each node in the problem space. (b) Energy values for only those nodes on the primary path. Note that the energy values for the different iterations are displaced by 0.5 each for purposes of display (and thus the caption reads "relative" energy).

results obtained with the model solving this even more informative problem are presented in Figures 12 and 13. As examination of those figures shows, the problem was easier with the energy function smoothing rapidly and the number of moves to solution starting at approximately 400 (indicating the initial ease of the problem) and declining fairly rapidly. After the first few iterations, however, there were no differences between this and the Large Disk problem in moves to solution, despite the added initial information provided in this problem. The likely cause of this is that the added information was about states near to the goal state (the small triangles within the large bottom left triangle of the problem space), and after a few iterations of spreading information back from the goal state the effect of the initially provided information became irrelevant (see Figure 12). Thus, given that states very close to the goal state can be expected to be rapidly learned to be "good" states on early iterations of

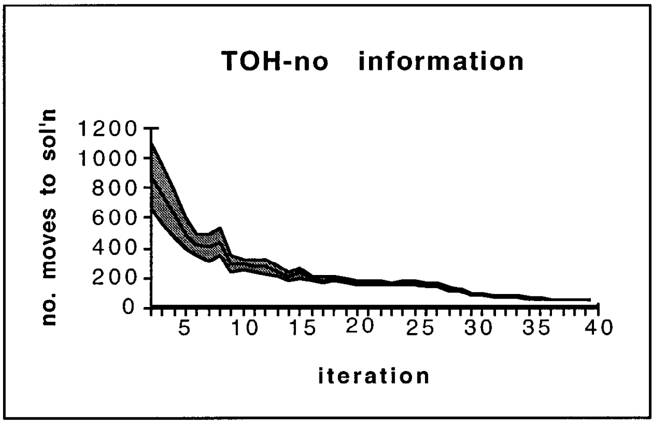


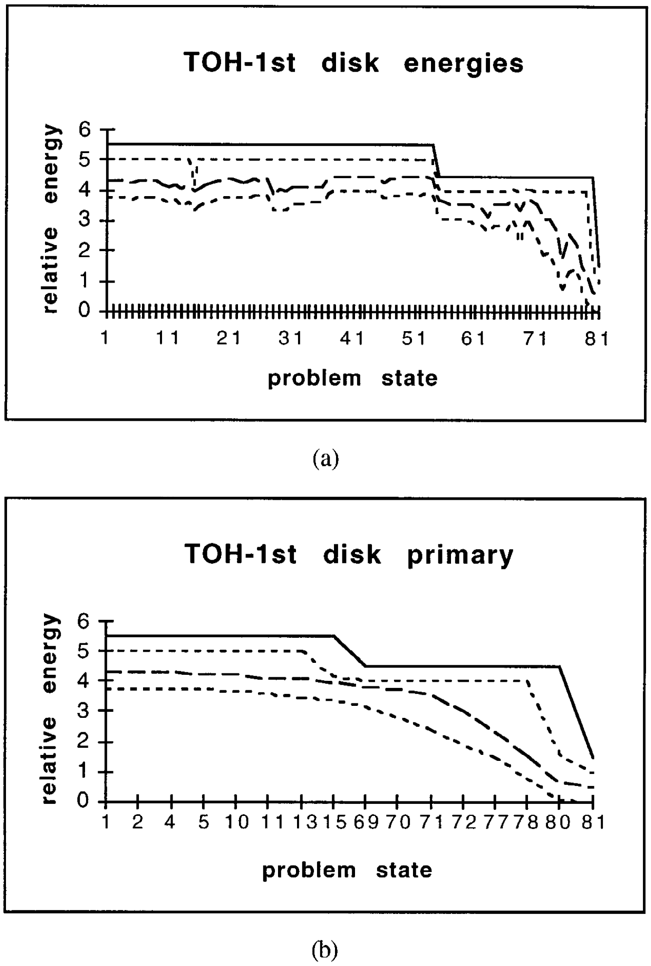
FIGURE 9. Tower of Hanoi No Information problem isomorph: Number of moves to solution vs. iteration.

the problem solving, providing information about these states most benefits initial trials with the benefit disappearing on later trials due to the learning capabilities of the model.

6.2.5. Random Walk Control. In examining the move records we have seen that the number of moves on the initial and final iterations has ranged from 1058 and 57 (Misleading problem) to 878 and 51 (No Information problem), 489 and 52 (Large Disk problem) and 432 and 48 (Three Disk problem). In order to better interpret these numbers, which are the basis for the argument that the modified energy levels in the objective function do represent real learning about the problem space, we need to compare these move numbers with the number of moves the model would take to solve the problems with no knowledge about the relative values of nodes in the problem space. We ascertained this via a random walk solution of the problem, a totally random choice of moves at each node in the problem space with no energy level/annealing apparatus operative. In Figure 14 we show the results of doing this. The plot is of the standard 40 iterations, 50 runs per iteration. It should be pointed out that the iterations are not meaningful in this experiment in that nothing changes from iteration to iteration, i.e., there is no learning mechanism, so each iteration is exactly like every other one, subject to random variation. Examination of Figure 14 shows that the number of moves to a solution averaged 832, and that, as expected, there was no decrease over the iterations. This stands in stark contrast to all of the conditions considered previously, where the number of moves, although sometimes starting high, nonetheless always decreased to an average value well below that obtained by random search. As indicated above, the Misleading problem initially yielded a higher number of moves than the random model due to the nonsolutions caused by the annealing settling into local minima, until the model's learning took effect.

6.3. Model Learning and Solution Characteristics:

6.3.1. Rate of Learning. If we plot the model's move record in a manner that reveals its rate of progress toward a solution, an interesting regularity emerges. This can be seen in Figure 15 where we show, for each of the four versions of the Tower of Hanoi problem, the progress through the problem space (in the form of visited states, with the start at 0 and the goal at 81) plotted against move number. This is shown for the 1st, 15th, and 40th iteration with the standard annealing conditions described previously. As can be seen in that figure,



(a)

(b)

FIGURE 10. Tower of Hanoi Large Disk problem isomorph: Energy function for the initial conditions (solid line) and iterations 1 (top short dash), 15 (long dash), and 40 (bottom short dash). (a) Energy for each node in the problem space. (b) Energy values for only those nodes on the primary path. Note that the energy values for the different iterations are displaced by 0.5 each for purposes of display.

while the model often fails to solve a problem on the first iteration for the Misleading problem, it always begins to solve it by the later iterations and does so in a fairly characteristic manner. On the other isomorphs, the model exhibits final-path behavior much earlier. There is usually an acceleration of progress toward the goal late in the problem solving, almost always in the period immediately prior to the model's reaching the goal.[14] This goal acceleration has been termed "final path behavior" in studies with human subjects (Kotovsky et al. 1985), and this

[14] In the panels of Figure 15 this goal acceleration is exaggerated by the large vertical jumps in the solution path which are due to discontinuities in neighboring node numbers as depicted in Figure 4. This problem does not occur in the Balls and Boxes results described in Section 6.4.3 because there are not discontinuities in node numbering due to the linear problem space. Note also that the scale of the abscissa may vary from panel to panel of Figure 15.

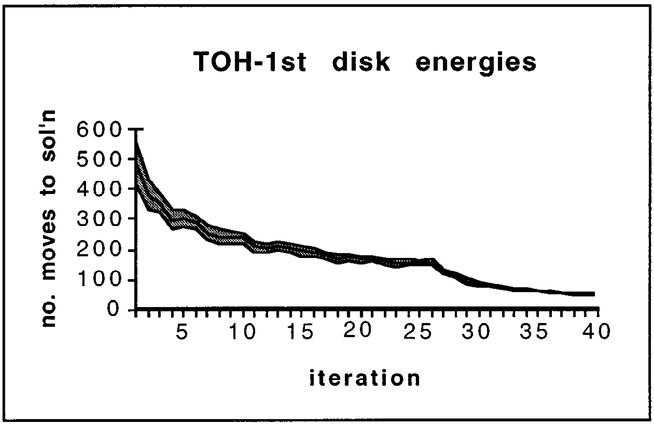
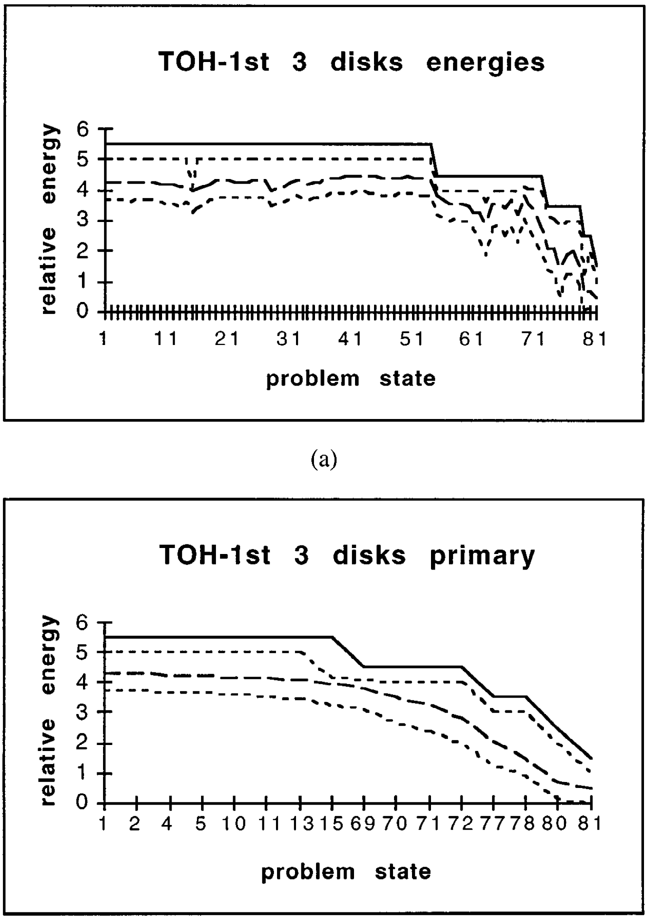


FIGURE 11. Tower of Hanoi Large Disk problem isomorph: Number of moves to solution vs. iteration.

feature of the model's behavior, the mechanism that produced it, and its similarity to human data will be discussed further below.

6.3.2. Exploration Encouragement. One feature of the model that has been briefly described above addresses the issue of the value of exploration, i.e., a wide latitude of move acceptance early in the problem solving so as to allow the model to explore the space. The manner in which this is implemented is that the temperature is increased (to a value that yields a probability of move acceptance of .5) when a previously unvisited state in the problem space is encountered and the probability of accepting a higher energy move is less than .25. This was done in order to "encourage" exploration of previously unexplored areas of the problem space even if the annealer had decremented to a low temperature; it corresponds to a person increasing the randomness of the behavior/search when encountering unfamiliar terrain. This device tends to operate only in early runs of the problem solving and is not so frequently invoked that it subverts the operation of the annealer in both initially allowing and progressively discouraging moves to higher energy states. Figure 16 shows that this feature of the model did meet these criteria of only being occasionally invoked and of dropping out with experience. As the figure shows, there are a few occasions where the "kick" in the temperature happens on the first iteration, and there are none by the fifth. This feature of the model thus represents a potentially interesting exploration enhancement, but is not a major determinant of the model's behavior in the problems reported here.

6.3.3. Asymptotic Behavior and Annealing Temperature Acceleration. Although the model does exhibit a great deal of learning on these problems, as described above, it appears to asymptote in the vicinity of 50 moves, improving very slowly on the last few iterations. This stands in sharp contrast to the substantial learning that occurs in the early iterations. Comparing its performance to the random walk model yields the conclusion that an impressive amount of learning is occurring. However, if the comparison is made to the minimum solution path length for this problem, 15 moves, the question arises as to why the model does not exhibit even more learning. When we remember that the model is based on simulated annealing, which has a probabilistic component that allows the acceptance of seemingly non–progress-making moves, then the answer suggested is that the excess moves might be due to the model



(a)

(b)

FIGURE 12. Tower of Hanoi Three Disk Energy problem isomorph: Energy function for the initial conditions (solid line) and iterations 1 (top short dash), 15 (long dash), and 40 (bottom short dash). (a) Energy for each node in the problem space. (b) Energy values for only those nodes on the primary path. Note that the energy values for the different iterations are displaced by 0.5 each for purposes of display.

staying “too hot” for too long; i.e., that the annealing, which is intended to represent the initial, more random part of problem solving, by not diminishing rapidly enough, is leading the model to engage in a fair amount of excess move-making.

In order to determine the likelihood of the annealing temperature being the explanation, we conducted the following experiment. For one of the problem types, the No Information problem, we parameterized the acceleration in the rate of temperature reduction used by the annealer, in order to explore the effect of allowing the model to cool faster. The “standard” temperature reduction rate reported in all the problems above was .99 with an acceleration of .001 on each subsequent iteration up to iteration 24, after which the acceleration factor was increased to .010. Alternative versions within this experiment kept the acceleration factor at .001 throughout, or increased it after iteration 24 to either .005 or .020 for the next 16

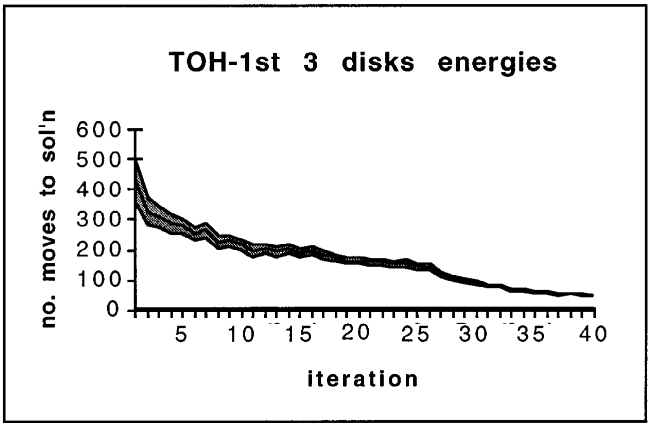


Figure 13. Tower of Hanoi Three Disk Energy problem isomorph: Number of moves to solution vs. iteration.

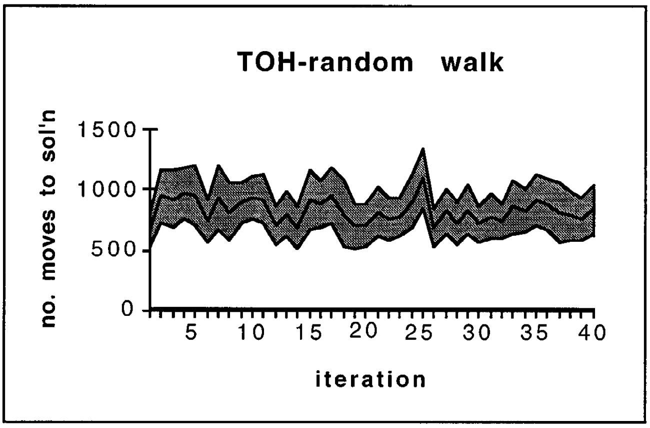
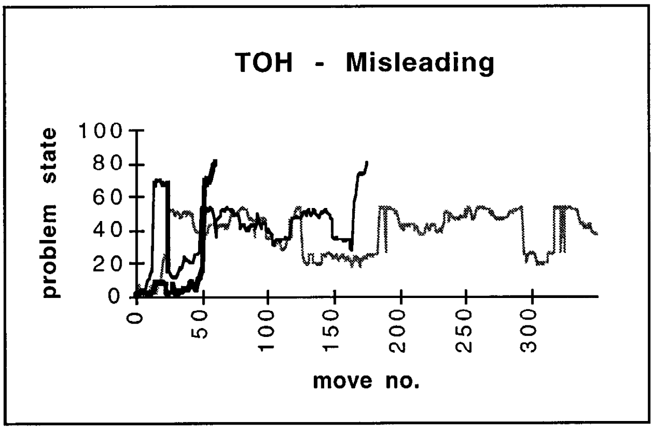


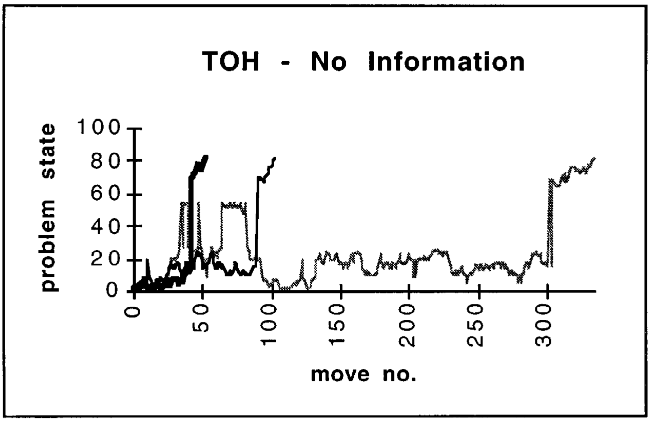
Figure 14. Tower of Hanoi problem: random walk solution. Number of moves to solution vs. iteration.

iterations. Finally, as a benchmark, we ran the annealer with the .99 temperature reduction rate and no acceleration.

With no acceleration, there is no change in the annealing reduction factor and any change in performance over the set of iterations is due entirely to the learning of the problem space (the modified objective function). Under these conditions, the model exhibits a significant amount of learning. As inspection of Figure 17a shows, it takes 897 moves to solve the puzzle on the initial iteration, and this decreases to 337 moves over the 40 iterations. This contrasts with the results with acceleration of annealing depicted in Figure 18, where we see that the initial iterations are similar (as we would expect given that the acceleration has not had time to occur) but the learning is much greater with acceleration than with no acceleration. With no acceleration, the slow rate of temperature reduction results in a large number of moves to solution, even after the model has had the chance to learn the value of many places in the problem space. As we allow the annealer a modest acceleration of .001 per iteration,



(a)

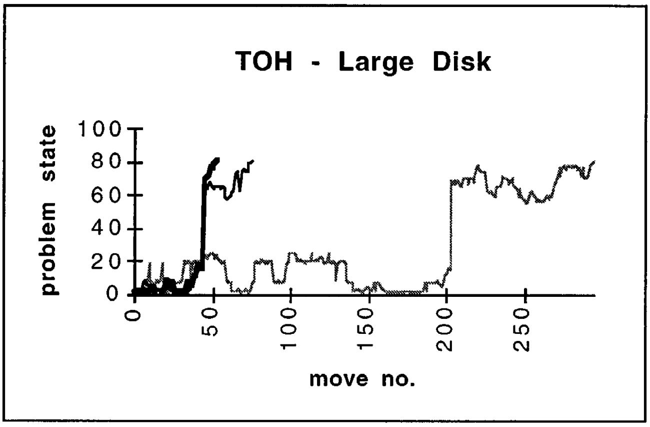


(b)

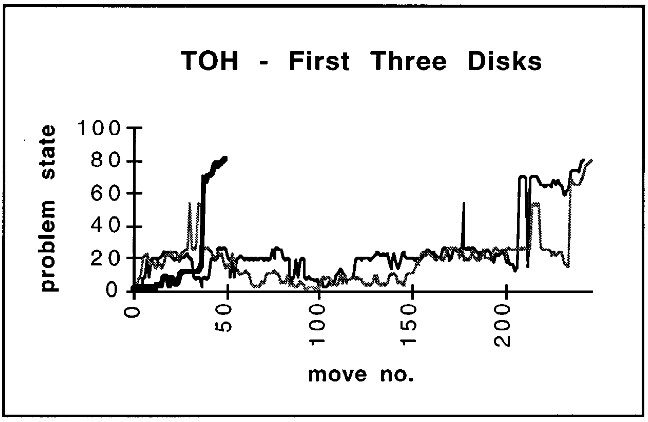
FIGURE 15. Tower of Hanoi problem isomorphs: Visited states (problem space state number) vs. move number during problem solution for iterations 1 (gray line), 15 (thin black line), and 40 (thick black line) for each of the four problem isomorphs: (a) Misleading, (b) No information. *Continued next page.*

the initial iteration stays about the same (as expected) but the number of moves to solution drops considerably to a final value of 120 at iteration 40. Thus, separating the two changes that occur during the model's exploration of the problem space—the rate of cooling of the temperature and the modification of the objective function values—demonstrates that the large preponderance of excess moves on the final iteration are the result of the relatively slow rate of temperature reduction. (A similar result is depicted in Figure 17b for the Misleading problem and will be referred to in Section 6.3.4.)

To further investigate the effect of the acceleration of the annealing on the improvement in performance, three different accelerated annealing rates were applied on iterations 25 to 40, as mentioned above. These were .005, .010 (standard run), and .020, resulting in final values of moves to solution of 71, 51, and 38, respectively (Figure 18). Thus the model's performance improved substantially as the rate of annealing was more rapidly accelerated. This result



(c)



(d)

FIGURE 15. *Continued.* Tower of Hanoi problem isomorphs: Visited states (problem space state number) vs. move number during problem solution for iterations 1 (gray line), 15 (thin black line), and 40 (thick black line) for each of the four problem isomorphs: (c) Large disk, and (d) Three-disk energy.

highlights an important aspect of the functioning of the model: the interaction between the progressively smoother energy distribution that results from the model learning about the space, and the annealing, which at any given temperature tends to produce unnecessarily random activity in the smoother energy spaces. There is thus an interaction between learning about the space and the optimal rate of annealing, suggesting that a dynamic adjustment in the annealer as the model learns would be effective at reducing the number of moves needed to solve the problem on later iterations. This might correspond in humans not only to the random exploratory phase giving way to more directed search, but to its doing so under the control of cues from the environment. In fact, we believe this factor may be the major source of the sizable differences in number of moves and rate of learning between the behavior of the model and that of human subjects. Thus, as the problem solver becomes aware of the problem space becoming more recognizable and familiar, the solver's behavior

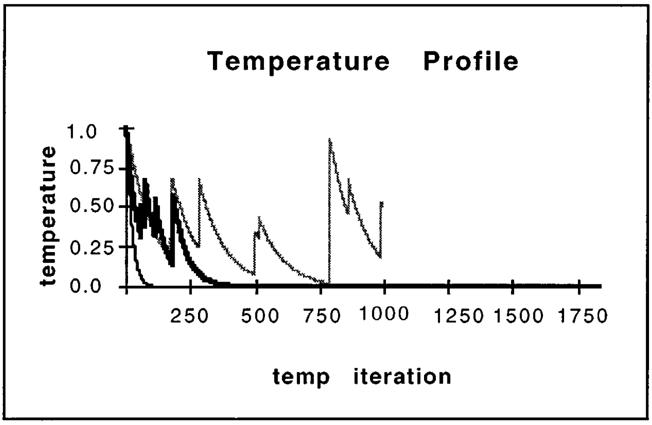
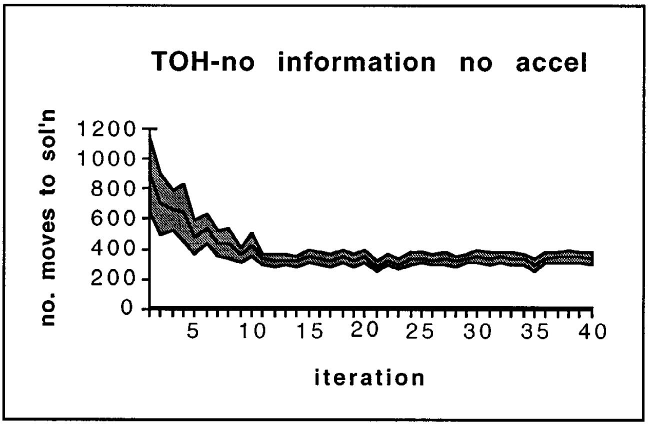


FIGURE 16. Tower of Hanoi problem: Temperature profile for iterations 1 (gray line), 2 (thin black line), and 5 (thick black line).

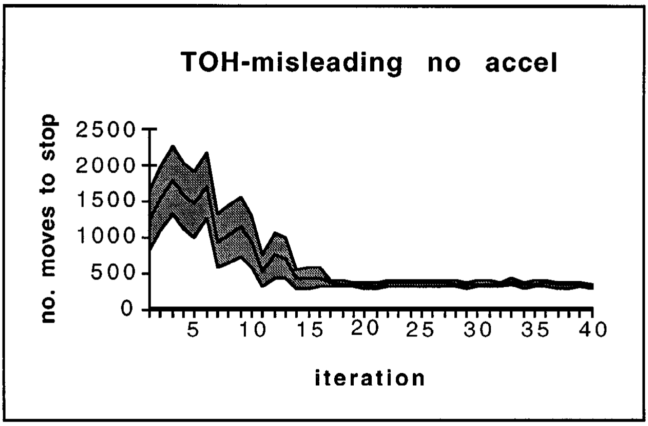
might become less random—an effect opposite to the upward adjustment of the annealing temperature when the model encounters unfamiliar portions of the problem space that was described above. Our "tweaking" of the annealing function acceleration in the standard run from .001 to .010 on the 24th and subsequent iterations is simply a first approximation to what needs to be done in this regard with respect to modeling this function. Our overall conclusion from this experiment with annealing acceleration is that learning about the problem space is not enough; the probabilistic tendency to violate that acquired knowledge and move to higher energy level states, so useful early in the problem solving, must be sharply curtailed in order for the learning to be maximally effective.[15]

6.3.4. Components of Learning: Annealing Acceleration vs. Energy Modification. In an effort to further disambiguate the contributions made to the model's learning by the changes in the energy levels as compared with those made by the acceleration of the temperature reduction in the annealing schedule, we conducted a pair of experiments that separated the two contributions to learning. The first used the Misleading problem, where the model solved the problem with normally accelerated annealing (initial reduction .99 and acceleration of .001 on each iteration up to the 24th and .010 from the 25th to the 40th) but with no learning of the problem state space energy values. This allowed us to determine the effect of annealing acceleration alone on learning, and thus indirectly test our assumption that the "real" learning is in the form of the alterations of the energy function as a result of experience. The result of turning off that learning while allowing the accelerated annealing reduction to occur was that the model almost never solved the problem but rather became stuck in a local minimum as the temperature dropped to zero. Thus the learning was, as expected, in the alteration of the energy levels that represented acquired knowledge of the problem space, or possibly resulted

[15] Note, however, that in a similar experiment with the Misleading problem, introducing the increased reduction factor in the early iterations became detrimental to the performance of the model. Here, the annealer became too cool before the objective function smoothed out, allowing the model to become trapped in a local minimum. (This may be similar to the often observed phenomenon of human solvers getting "stuck" in a wrong part of a problem space and having trouble extricating themselves.) This further supports our argument that a dynamic adjustment strategy needs to be investigated.



(a)



(b)

FIGURE 17. Tower of Hanoi problem isomorphs: Number of moves to solution with no acceleration of the annealer for the No Information (a) and Misleading (b) isomorphs.

from an interaction between the alteration of the energy levels and the acceleration of the annealing.

The second experiment involves the data from Section 6.3.3 where the annealing acceleration was removed from the learning process while alterations of the energy function were allowed to proceed. The Misleading and No-Information problems were solved with a constant reduction factor of .99 on each iteration. The results are depicted in Figures 17a (No Information) and 17b (Misleading), which show that there is improvement in performance with experience (iterations), as evidenced by the decrease in the total moves to solution. This is true for both the Misleading and the No-Information problems. As before, this improvement is due to two sources: increased speed in finding the goal when the problem is solved and decreased number of failures to solve the problem (in the Misleading problem). Thus the learning of the energy function alone produces a significant improvement in performance; however, a comparison of the number of moves to solution in this energy function learning

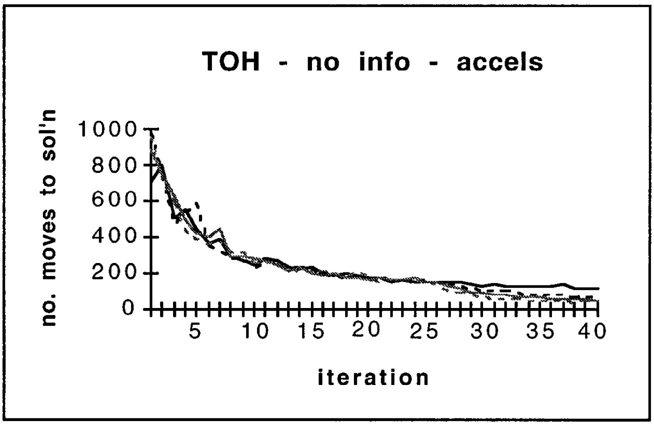


FIGURE 18. Tower of Hanoi problem No information isomorph: Number of moves to solution vs. iteration number for trials with varying annealing accelerations. Accelerations were varied starting with the 24th iteration and the set of accelerations used were .001 (solid black line), .005 (dashed black line), .010 (solid gray line), and .020 (dashed gray line). The acceleration used up to the 24th iteration was the standard .001.

case with the number of moves where both energy function learning and annealing acceleration are present (Figures 7 and 9) demonstrates that the final performance in this case with no acceleration is worse. The model reaches an asymptote on fairly early iterations with no acceleration of the annealing, while with acceleration the model continues to improve through the final iteration.

The results of this pair of experiments show that although there is a marked improvement in the model's performance as a result of the learning of the energy function without accelerated annealing, and no improvement or even a decrement in performance with annealing acceleration alone, the greater learning shown in the earlier results is due to an interaction between the two learning mechanisms: the learning of the energy function and the acceleration of the annealing. These results support the conclusion that optimal performance depends on an interaction between the smoothing of the energy function and the acceleration of the annealing. Without the learning of the energy function the acceleration of the annealing did not result in an improvement in performance, and with no acceleration of the annealing there was improvement, but even after many iterations it was very limited. We surmise that the reason for this limited improvement on later iterations is that without annealing acceleration, the temperature remains too hot for the increasingly smooth energy function of the learned space (as justified above). As a further experiment, we ran the standard model (combined accelerated annealing and energy modification) for 40 iterations, then turned off all annealing (accepting only improved moves) and ran the model for a 41st iteration. The result was that the goal consistently was found in approximately 22 moves for all problems, demonstrating the efficacy of the model's learning, i.e., that the smoothing of the space provided a learned downhill path to the goal and that the excess moves found under our standard annealing conditions are due to the still warm, if not hot, annealing temperature on the final iteration.[16]

[16] Note that the number of moves in the direct solution is 22 and not the minimum of 15. As the model moves toward the goal, it encounters nodes that have more than one branch which improves its position. The model, not having a fully optimized move selection policy, randomly picks any one of these branches, not necessarily the optimum one, resulting in its sometimes taking short side excursions off the minimum path. A similar type of finding has been reported in some three-disk isomorphs

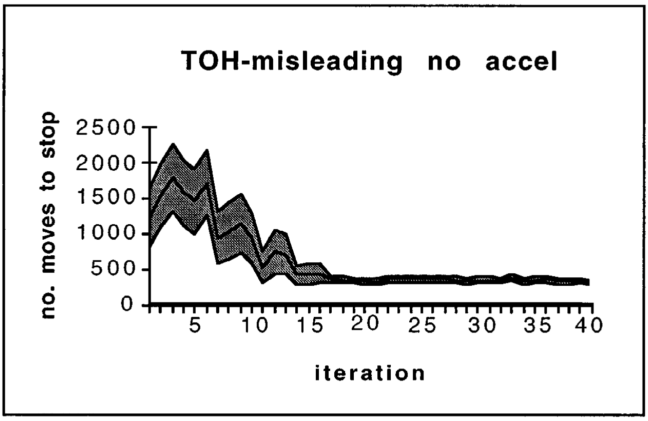
Upon further analysis, the initial difficulty in the case of the Misleading problem with no acceleration described above is found to be due to the large number of times the model became trapped in a local minimum caused by the initially erroneous downhill search assumption provided to the model on that problem. Over subsequent iterations, the model learned about the problem space, and the frequency of successful solutions rapidly increased to 100% (Figure 19c). If we plot only those solution attempts that were successful—those that resulted in the model finding the goal—the number of moves remains essentially constant across all 40 iterations (Figure 19b). Thus the entire improvement seen in Figure 19a is due to the disappearance of the nonsolutions. The fact that the number of moves remains high even on the final iteration (324 moves) in the no-acceleration case indicates that the annealing temperature remains too high.[17]

6.3.5. Summary: Tower of Hanoi Results. What we learned from experimentation with the model solving a number of versions of the Tower of Hanoi problem is, first, that the model can learn to represent the problem space in the form of its energy function, and in so doing learn to modify that function, thus developing an increasingly accurate and useful representation of that problem space so as to increase the efficiency of its performance. Second, it can do so without being provided in advance with significant information about the structure of that space (which states or move directions take one farther from or closer to the goal) and can even recover from the kind of misleading strategy reasonable problem solvers relying on the weak method of downhill search often use. It takes some extra iterations in this misleading case but the model recovers and learns to solve more effectively (in fewer moves). Third, we determined the effect of providing the model with some useful initial information a problem solver might be expected to have and, as anticipated, found it to be effective in reducing the initial difficulty of the problem and in facilitating more rapid learning. It also resulted in an increase in the consistency of the model's performance on multiple runs as demonstrated by lower standard errors. Fourth, we disentangled the effects of two modifications of the model that occur during iterations: the learning of the energy function that we claim is primarily responsible for the increased efficacy of the model's behavior, and the acceleration of the cooling of the annealing function with additional iterations. The major finding is that the learning is primarily accomplished by changing the energy levels with experience, that accelerating the annealing without recording those changes in energy levels does not yield learning, but that the two interact; accelerating the annealing does significantly increase problem-solving speed if the energy levels have been smoothed. Finally, the model has been shown to learn a complete downhill characterization of the path to solution.

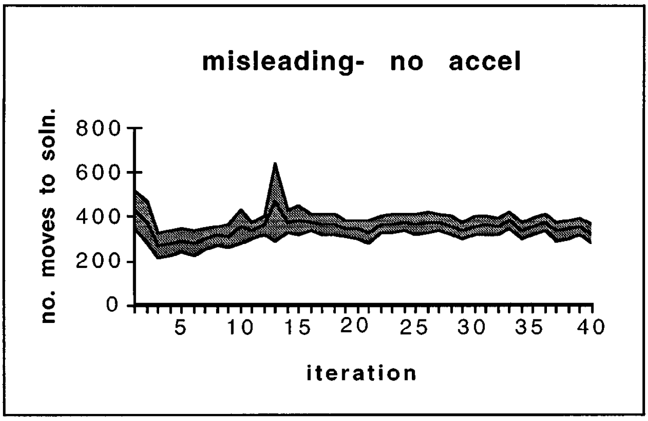
In exploring the model's behavior on the first problem we were mainly interested in testing whether the model can progressively learn an effective representation and solve various isomorphs of the problem, and how the various mechanisms of the model contribute to its performance. However, although we have constructed the model to be reasonably compatible with what we know about human cognitive mechanisms and performance, we did not perform any explicit comparison of the model's behavior with that of human subjects except in the most general way. We turn to that issue next as we consider the performance of the model on a second problem and a more detailed consideration of the exploratory–final path transition.

of the Tower of Hanoi problem where a short "side-excursion" or alternative pathway to the goal was taken by a sizable subset of subjects (Kotovsky et al. 1985).

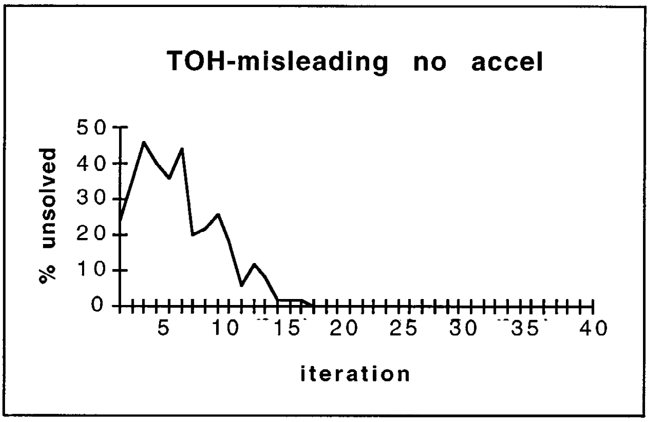
[17] We did not have to perform a similar analysis for the No Information problem because the absence of local minima meant that there were zero nonsolutions to that problem.



(a)



(b)



(c)

FIGURE 19. Tower of Hanoi problem Misleading isomorph with annealer turned off on final iteration: (a) number of moves to stop, (b) number of moves to solution on successful solution attempts, and (c) frequency of unsuccessful solution attempts.

6.4. Balls and Boxes Puzzle

6.4.1. Problem Isomorphs. We next examine the performance of the model on isomorphs of the five-ball Balls and Boxes/five-ring Chinese Ring Puzzle. The problem space of this problem is linear in that the only choices at all states other than state 31 and goal state 0 are to move either toward or away from the goal.[18] Despite the apparent simplicity and small size of the problem space, the puzzle can be quite difficult for human subjects. The Chinese Puzzle version is extremely difficult for people to solve within a two-hour time period. In one study only 1 of 14 college students was able to solve it. The Balls and Boxes problem is easier, but versions of it can take an average of up to 420 moves to solve (Kotovsky & Simon 1990). The linear portrayal of the space gives us the advantage that we can readily compare move records between the model and humans. The types of problems presented to our model were similar to the set used for the Tower of Hanoi problem, as described in Table 3.

6.4.2. Model Learning and Solution Characteristics: Balls and Boxes Problem. The results obtained with these problems are depicted in Figures 20 through 23. Panel a in each of these figures shows for each state the typical energy configuration for the initial conditions, the 1st, 15th, and 40th iteration. The figures show in each case progressive and rapid learning, visible as the smoothing of the energy curves. The modification of the energy states was made according to the matrix presented in Table 1, the same transition matrix used for the Tower of Hanoi problem. The result of this learning is also depicted in Figures 20 through 23, panels b, which display the model's performance, plotting the number of moves taken to solve the problem versus iteration for the 40 iterations of each learning trial. As before, the annealing started at .99 on the first iteration and was accelerated by .001 on each of the first 24 iterations, and by .010 on each of the remaining 16. Each point in the figure again represents the average of 50 runs, corresponding to 50 subjects. As the figures show, the model learned under all of the problem conditions.

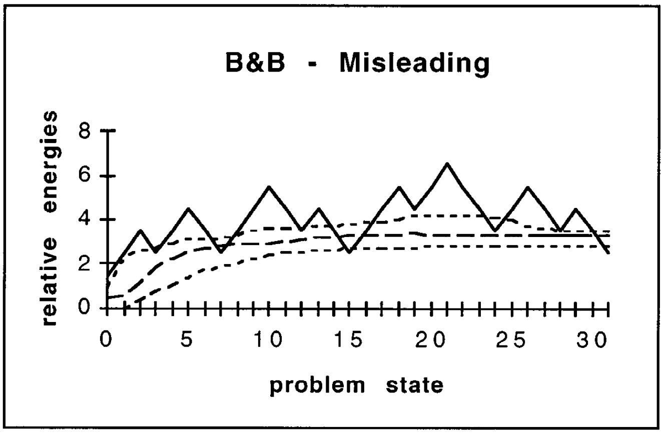
Particular outcomes are that in the Misleading condition, the model learned fairly steadily, the number of moves decreasing from an initial 1078 moves to 56 on the final iteration. As with the Tower of Hanoi Misleading condition, the model frequently failed to solve the problem on the early iterations (Figure 20d) with the failure rate decreasing to zero or near zero by the end as the learning took effect. Reporting the number of moves for successful solutions only (Figure 20e), the initial moves were 369 decreasing to 56.

The No Information problem was solved with the standard annealing schedule (initially .99) and acceleration (.001 for first 24 iterations, and .010 on the remaining 16). As depicted in Figure 21, the initial iteration required 792 moves to solve, decreasing to 48.

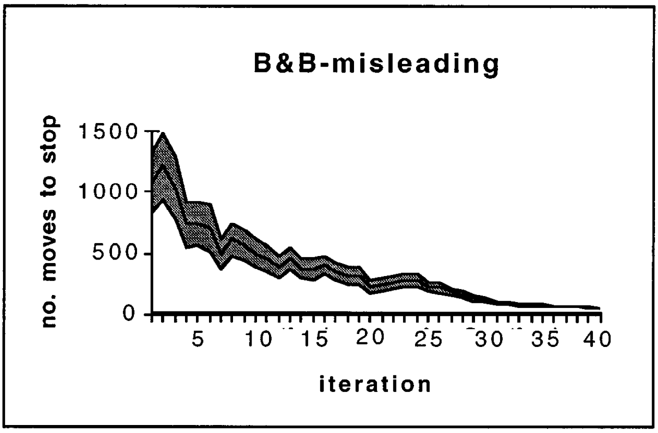
The next problem type was the First Ball Out (corresponding to the Large Triangle problem in the Tower of Hanoi set of problems). Here the model is potentially given a great deal of help by the fact that the initial energy levels of all the states below state number 15 are given a lower energy level (3) than all the other states that are at the default level of 4 except for the goal state that is always assumed recognizable and given the level 0. In this case, the initial solution is very fast, taking 478 moves, and by the final iteration solving the problem in 48 moves (Figure 22).

The final problem solved was the First Four Balls problem that gives the model the maximum potential help provided in any of the Balls and Boxes problems. Here, each ball's correct removal was indicated by a reduced initial energy level. The results were that the

[18] State 31 and the goal state represent the endpoints of the linear problem space and therefore have only one move choice each. At each other point in the problem space, there are two move choices: a solver can either continue in the direction of the last move or else rescind it, or, viewed alternatively, move one step closer to the goal or one step away.



(a)



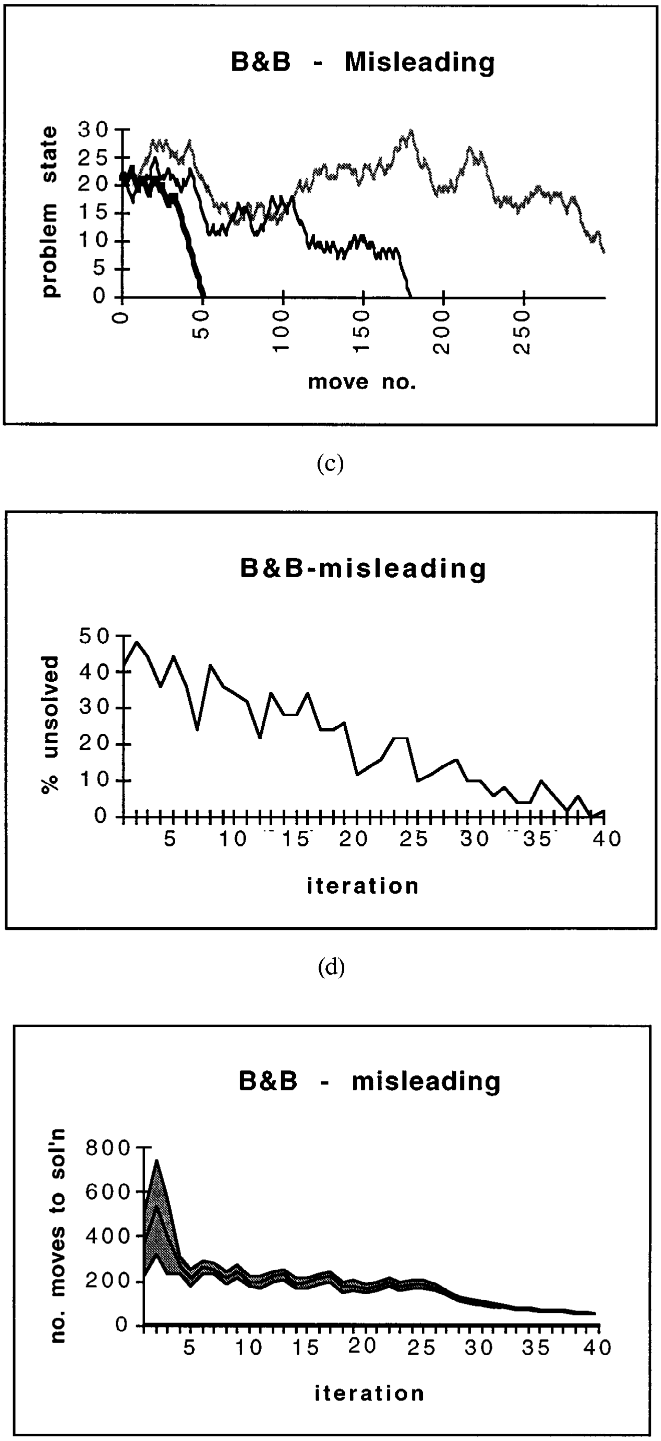
(b)

FIGURE 20. Balls and Boxes problem Misleading isomorph: (a) Energy function for the initial conditions (solid line) and iterations 1 (top short dash), 15 (long dash), and 40 (bottom short dash). Note that the energy values for the different iterations are displaced by 0.5 each for purposes of display. (b) Number of moves vs. iteration. *Continued next page.*

initial solution (again, under standard annealing conditions) was obtained in 265 moves, with the model learning to solve it in 47 moves by the final iteration. The results for this problem are depicted in Figure 23.[19]

The final experiment carried out with this problem was to determine the number of moves that would be required by the model to solve the problem via a random walk, i.e., with no differentiation of the states via differing energy levels and no annealing. The model in this

[19] As in the TOH, the high number of moves in the later, postlearning iterations are due to the nonoptimized annealing schedule. A faster cooling results in solutions much closer to the minimum solution path of 21 moves; when the annealing is turned off on later iterations the model consistently took exactly 21 moves to solve.

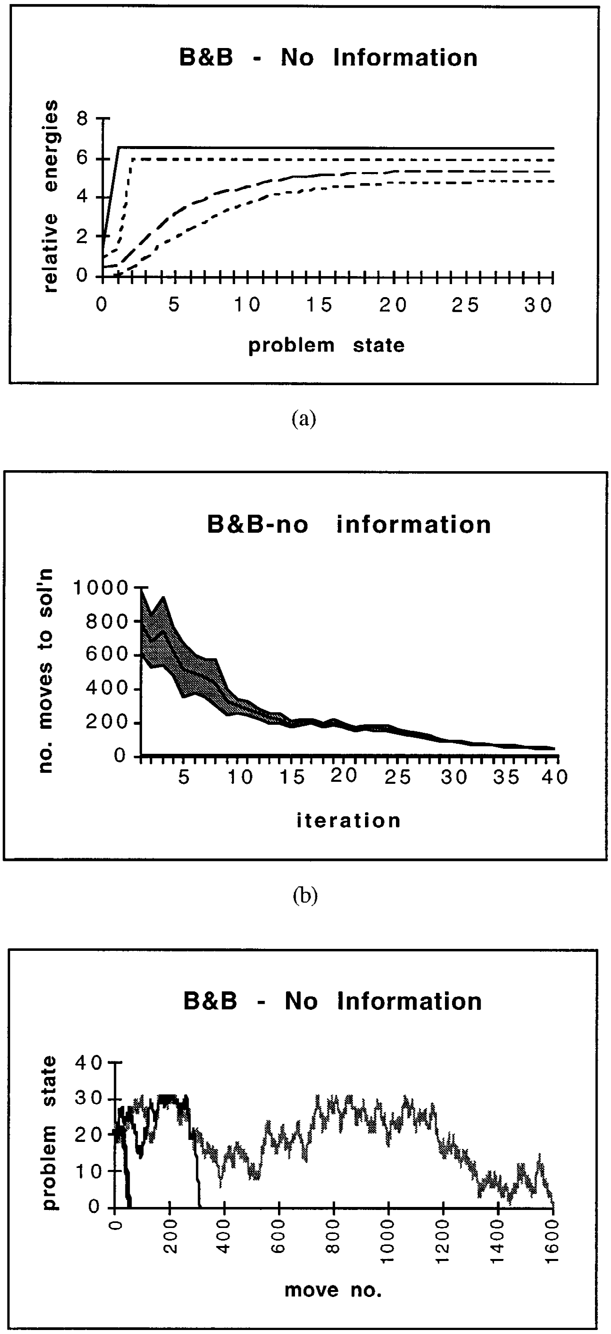


(c)

(d)

(e)

Figure 20. *Continued.* Balls and Boxes problem Misleading isomorph: (c) Visited states (problem space state number) vs. move number during problem solution for iterations 1 (gray line), 15 (thin black line), and 40 (thick black line). (d) Percent successes vs. iteration number. (e) Number of moves to solution, successful solutions only.

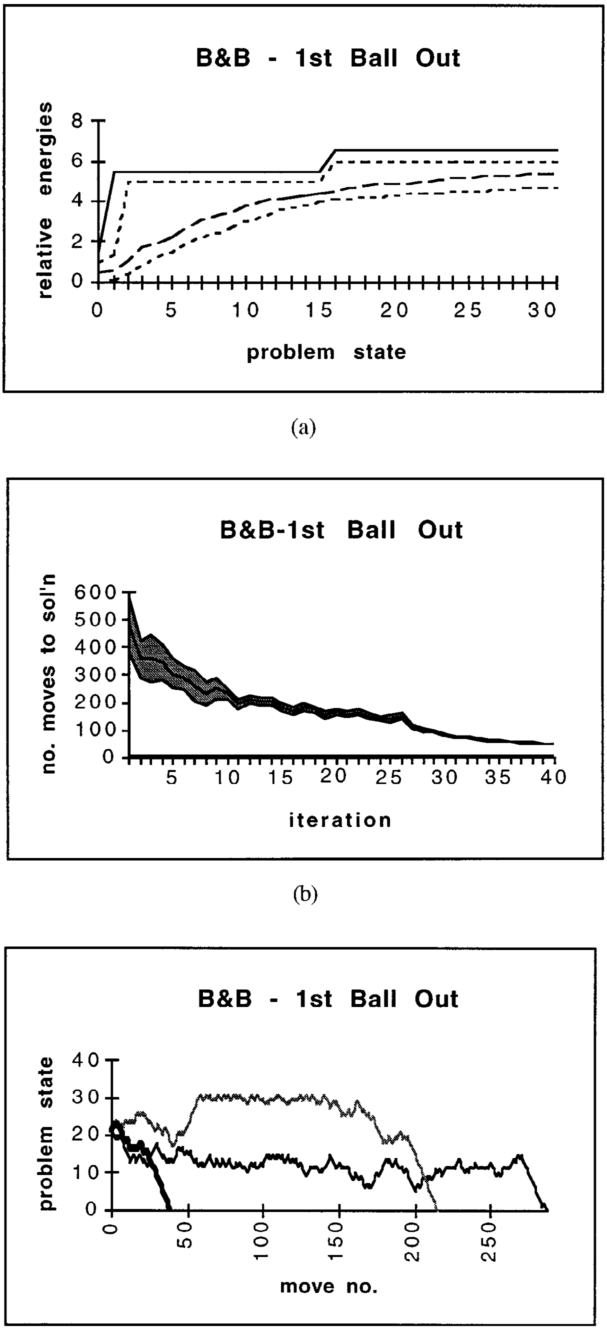


(a)

(b)

(c)

FIGURE 21. Balls and Boxes problem No Information isomorph. (a) Energy function for the initial conditions (solid line) and iterations 1 (top short dash), 15 (long dash), and 40 (bottom short dash). Note that the energy values for the different iterations are displaced by 0.5 each for purposes of display. (b) Number of moves vs. iteration. (c) Visited states (problem space state number) vs. move number during problem solution for iterations 1 (gray line), 15 (thin black line), and 40 (thick black line).

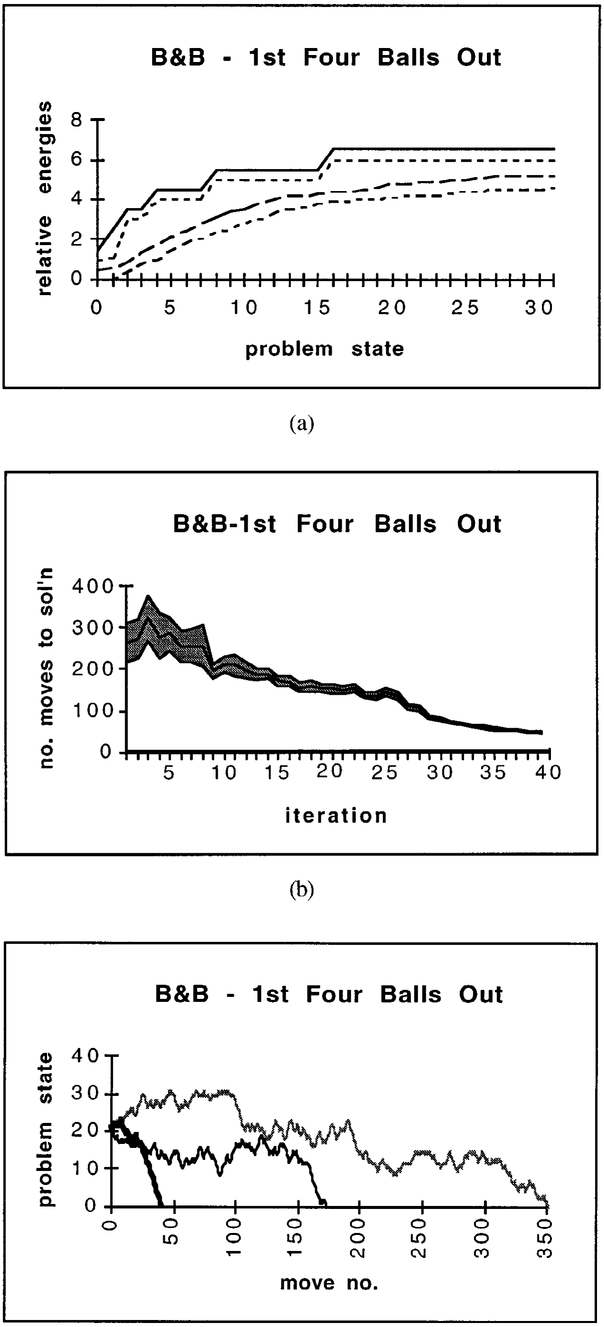


(a)

(b)

(c)

FIGURE 22. Balls and Boxes problem First Ball Out isomorph. (a) Energy function for the initial conditions (solid line) and iterations 1 (top short dash), 15 (long dash), and 40 (bottom short dash). Note that the energy values for the different iterations are displaced by 0.5 each for purposes of display. (b) Number of moves vs. iteration. (c) Visited states (problem space state number) vs. move number during problem solution for iterations 1 (gray line), 15 (thin black line), and 40 (thick black line).



(a)

(b)

(c)

FIGURE 23. Balls and Boxes problem First Four Balls isomorph. (a) Energy function for the initial conditions (solid line) and iterations 1 (top short dash), 15 (long dash), and 40 (bottom short dash). Note that the energy values for the different iterations are displaced by 0.5 each for purposes of display. (b) Number of moves vs. iteration. (c) Visited states (problem space state number) vs. move number during problem solution for iterations 1 (gray line), 15 (thin black line), and 40 (thick black line).

TABLE 3. Versions of the Balls and Boxes Problem.

Problem	Energy Levels Configuration
Misleading	All nodes set at an energy level (0 to 5) corresponding to the number of balls out of their boxes, with increasing number of balls out being equated to smaller energy levels (energy = 5 − number of balls out). This is often misleading in that taking out the wrong ball can increase the distance to the goal rather than reduce it. These initial assumptions are those that might be expected from a problem solver who initially follows a downhill search heuristic that naively assumes that removing any ball is desirable.
No Information	All nodes at the same energy level (5) except goal state (0).
First Ball Out	All states corresponding to configurations where the most restricted ball is out (problem space state 15 and below) were given lower energy levels (4), with all other states except the goal state set at 5. The goal state was set to 0.
First Four Balls	All states corresponding to configurations where n balls (*in correct order*) were out had a lower energy level equal to $5 - n$. The goal state was set to 0. This problem provided a great deal of potentially useful information to the solver.

condition takes an average of 846 moves to solve the problem on each iteration and there is, of course, no learning. These results are depicted in Figure 24.

6.4.3. Model Performance Summary. The model was able to solve the Balls and Boxes problem, as it was the Tower of Hanoi problem, and as shown by the comparison with the random walk model, its learning mechanisms were effective; with its learning mechanisms intact, it learned to solve these isomorphs with many fewer moves over time (iterations) than when the learning mechanisms were disabled. We thus see that the model was capable of solving two types of problems even when provided with misleading initial assumptions about the problem space and that it learned effectively, albeit over a fairly long time course. We next turn to a consideration of the similarity of the model's behavior with that of human problem solvers at a crucial juncture in the solution process when, as a result of learning, the solvers suddenly exhibit much more efficacious behavior and finally converge rapidly on the goal.

6.4.4. Final Path. An important feature of the model's behavior is the manner in which it traversed the problem space in solving the problems. The progression through the space is depicted in panels c for each problem (Figures 20–23). The model's traversal of the space (from its start at position 21 to the goal at position 0) is similar to the results obtained with the

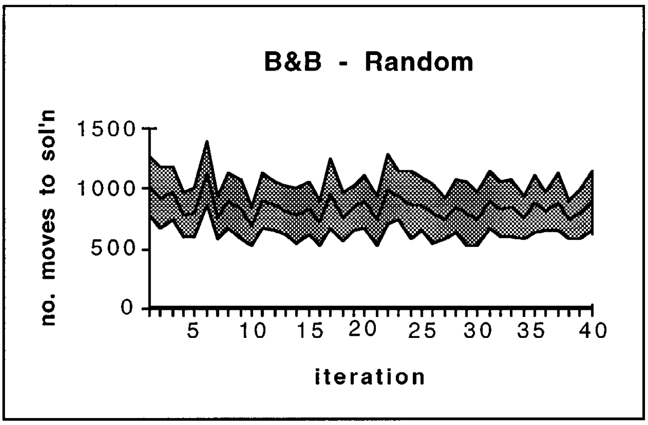


FIGURE 24. Balls and Boxes problem random walk solution: Number of moves to solution.

Tower of Hanoi problems in two important respects. Although the Misleading problem was not at first solved (on early iterations), all other problems were consistently solved, as was the Misleading problem on later iterations. (The figures present results for iterations 1, 15, and 40.) Additionally, the form of the solution again shows an acceleration late in the problem solving, as the model seems to undergo a transition from more or less random behavior to a rapid traversal of an often sizable portion of the problem space as it closes on the goal. This type of "final path behavior", as discussed in Sections 4 and 6.3.1, was also found in previous work with human subjects solving this problem (Kotovsky and Simon 1990). The human subjects' data are depicted in Figure 5, which shows a similar effect as the subjects seem to at first move almost randomly or at least in a manner that involves a very large number of moves with little or no progress toward the goal, and then make a rapid approach to the goal with a sustained period of almost error-free (nonreversing) move-making. The fact that this occurs even on the first or early iterations (for the model) and on the first iteration (for humans) means that it is not solely due to learning the shape of the problem space, but that it also depends on changes in the operating characteristics of the solver—a result broadly compatible with the effective interactions between energy function smoothing and progressively decreasing randomness seen in our model's performance. The fact that the model behaves in a roughly similar fashion may indicate that the manner in which this pattern arises might be similar for human and machine; that somehow the learning that occurs during the exploration of the problem space (in the early moves by the subjects and in both the early moves on a given iteration as well as on the accumulated moves from prior iterations by the model) allows the sudden seemingly expert rapid closing on the goal that is seen in both cases.

A key insight into the behavior of the model can be found in exploring why this final path behavior can occur even on the first iteration, before it has traversed those states on the path, on those isomorphs which have landmarks that produce an energy differential in states distant from the goal (see, for example, Figure 22c). As the model moves early on in these isomorphs, it modifies the energy values through its adjacency learning mechanism and produces a subtle downhill ramp toward the goal. As problem solving proceeds and the annealing temperature cools, the model, when headed toward the goal, tends to not reverse due to the cooled temperature coupled with the slight energy ramp it is producing as it moves. It thus traverses the final path to the goal. In humans this corresponds to a tendency to not reverse direction when moving from a region of uncertainty, a tendency that often leads them to the goal.[20] These same isomorphs where the energy ramp is created also have a number of moves to solution compatible with that of humans solving the Balls and Boxes problem.

A similar finding about a dichotomous exploratory and final path phase of problem solving has also been reported for a set of TOH problem isomorphs, although, as discussed earlier, it is likely to involve additional cognitive-representational issues not explored here (Kotovsky et al. 1985). The similarity between the behavior of the model and of humans in this regard suggests that, just as the model's moves are governed by mechanisms having a somewhat random character early in the process (the annealing) with little knowledge of the space (the energy levels) but changing to a more directed and knowledge-guided approach later in the problem-solving episode, so too might humans move from a process having random-exploratory processes to one that is more knowledgeable or skilled.

The conclusion that the final path behavior is indicative of learning is supported by an understanding of the mechanisms that bring it about in the model: the progressive smoothing of the energy values of the problem space and the cooling of the annealing temperature. In humans, one piece of evidence for the final path being indicative of learning is that the "strength" or "goodness" of the final path[21] was strongly predictive of performance on the second solution attempt of the identical or of a similar problem, thus indicating useful learning (Kotovsky & Simon 1990). Similarly, on multiple solutions of the same problem, the final path length increased from solution to solution, again indicating progressive learning (Reber 1993).

The model at times (Figure 22c) but not always (Figure 20c) exhibits final path behavior on the first iteration. In this regard, its behavior is somewhat similar to that of human solvers who usually exhibit final path behavior on the first solution attempt but do not always do so. (See Kotovsky, Hayes, & Simon 1985 for TOH, and Kotovsky & Simon 1990 for B&B.) The fact that the model sometimes does not exhibit final path behavior until the second or later iterations could thus mean one of three things. It could be that there are differences between the mechanisms that yield this behavior in people and in the model, possibly resulting from our conservative approach of keeping the annealing temperature higher than is necessary or most efficacious on most problems. Another possibility is that there are differences between the initial energy assumptions in the different problem isomorphs that make it more or less likely to form an energy ramp when headed toward the goal. Additionally, we might think of the model's varied behavior as being reflective of the individual differences found with human solvers, possibly arising from different contributions of problem space learning and temperature cooling (randomness reduction) on different iterations, with no negative implications for the match or mismatch of underlying mechanisms.

7. CONCLUSIONS

The major findings obtained with the model solving both the Balls and Boxes and the Tower of Hanoi problems are that the model solves these problems with very little or even

[20] For isomorphs where no energy differential is initially available (such as the No Information problem), more than one iteration is likely to be required to create this energy ramp because the model must first visit the goal to create and propagate an energy differential. Even here though, it is possible that an energy differential may be introduced by a simple a priori assumption on the part of subjects. For example, the simple assumption that the start state is a "bad" one (i.e., has a higher assumed energy level than any other state) can produce such a differential and yield the kind of energy ramp that would lead to an antireversing bias such as is often seen in human subjects' behavior.

[21] Measured as the ratio of the number of moves to reach the goal from benchmark positions 10 moves and 15 moves from the goal, as well as by the number of reversals and illegal moves that occur between the last occupancy of position 21 and the attainment of the goal.

no information about the structure of the problem space, that it can recover from somewhat misleading information (of the type that an inappropriate downhill search strategy[22] would provide), and finally, that providing it with even relatively small amounts of information about the problem space has a significant influence on both the speed with which it solves the problem and the rapidity with which it learns. The major implication of these findings is that a model that uses simulated annealing and represents the knowledge it acquires by modifying the objective function can effectively solve problems with very limited knowledge and can achieve a correct downhill objective function representation of the solution path. Although providing information about the structure of the problem space accelerates the problem solving and learning, the provision of such information is not necessary in the problems we have investigated. What we have shown instead is that a more cognitively plausible approach of starting the model out with no information or a few "landmarks" that it knows in advance and allowing it to discover the structure of the problem space can be very effective. In addition, the interaction between annealing speed and the current state of knowledge suggests an intimate relationship between the optimal rate of annealing and the current "smoothness" of the energy function describing the model's knowledge of the problem space at any point in time.

The behavior of the model illustrates certain key characteristics that are often found in the behavior of humans solving problems. The most general similarity is that as the model solves a problem it increasingly learns information about the problem space that can guide move selection. This learning accumulates over repeated solutions of the problem, resulting in more rapid solution. A particular feature of this learning is the finding that it results in two problem-solving phases: an initial exploratory phase characterized by a seemingly random component, followed by a final path phase consisting of a knowledge-guided set of moves to the goal. This similarity in the shape of the move records suggests that there may be analogous mechanisms operating in both humans and the model whose operations account for the transition between exploration and final path. As discussed earlier, the model sometimes takes a bit longer to reach the point of exhibiting this strong exploratory–final path dichotomy, on some isomorphs not doing so at all on the first solution of the problem. This also was true of some subjects' performances, but it may also indicate differences in the precise way the model achieves the transition on some problems. It is nonetheless the case that the basic behavior is similar and, further, is indicative of a transition from more or less random move-making to more directed behavior—an important transition for problem solving.

Perhaps the largest difference between the performance of the model and that of humans is in the number of moves required to solve the problem, with the model generally learning more slowly and requiring more moves than do the human subjects, at least on many of the isomorphs. Despite this difference, the characteristic behavior of human solvers is captured by the model in that not only do both exhibit final path behavior, but humans also exhibit imperfect problem-solving performance even on multiple solutions of the same problem isomorph. For example, Reber (1993) had subjects solve the same isomorph four times in

[22] The strategy is "inappropriate" in that the values assigned to the various states in the problem space are based on the appearance of closeness to the goal rather than the reality of how close the state is to the goal. Thus in the Balls and Boxes problem, for example, the values were based on how many balls were out of boxes such that every move that removed another ball from its box resulted in a lower energy level. In reality, many of those moves took the solver away from the goal and, correspondingly, should have been assigned higher energy levels. A similar effect holds for the Tower of Hanoi problem where moving disks to the goal peg was initially viewed as an unequivocally positive event, whether it was a correct move or not. Just as people reduce their reliance on such a simple downhill search heuristic as problem solving proceeds and they experience the lack of progress it yields in detour problems (Reber & Kotovsky 1992), so too the model progressively smoothes the energy function and thus escapes from the false direction of the same heuristic in the detour problem used here, the Misleading problem.

one study and although there was improvement from solution attempt to solution attempt, the humans were far from perfect on their fourth attempt, taking between 46 and 99 moves to solve it depending on the particular isomorph (down from 109 to 159 on the first solution). While these numbers do indicate that the model is generally taking much longer than humans to solve the problem, they are nonetheless not inconsistent with the claim that there are similarities in the basic processes utilized. As discussed earlier, more rapidly accelerating the annealing or dynamically adjusting it via a control mechanism that is sensitive to the variations in energy encountered during exploration of the problem space might eliminate or sharply reduce the differences. Additional support for the possible similarity of basic processes is provided by the finding that humans progressively decrease the probability of reversing a move within a given solution attempt and on subsequent solution attempts, as does the model via the cooling of the annealing. Another similarity is the lengthening of the final path seen in both humans and model across successive solution attempts (Reber 1993). The excess number of training or learning iterations we obtain in the model is reminiscent of the large number of training epochs that are commonly seen in PDP approaches to cognitive modeling (McClelland & Rumelhart 1986), and are at the very least an issue that is in need of further work. Finally, for those isomorphs in the Balls and Boxes problem where an energy ramp is created toward the goal, the number of moves taken by the model *is* compatible with those taken by humans solving the problem.

This work represents something of a hybrid approach in combining issues from human problem solving with a perspective arising from a more purely computational approach to issues of search, learning, and intelligent behavior. The model differs from some related computational approaches in that it relies on mechanisms that are at least broadly constrained by what is known about human cognitive architecture. These mechanisms include (1) the assumptions about how much knowledge of specific states in the problem space is built into the initial conditions of the model, (2) the reliance on a plausible learning mechanism that learns simple pairwise adjacency relationships, (3) the cooling of randomness whereby more undirected initial behavior gets replaced by more goal-directed expert-like behavior as problem solving proceeds, and (4) the recording of the "goodness" or value of various states in the problem space as being recognizable if they are familiar, without an overreliance on large amounts of memory for the entire move record of all moves made in approaching the goal. In so doing, it captures many important characteristics of human problem solving, particularly the exploratory/final path distinction, proposing a type of mechanism that gives rise to it.

The model does not, however, purport to be a complete model of human problem solving in that it does not have the type of full knowledge representation of the problem space that would allow it to plan moves with "understanding" based on knowledge of exactly where it is in relation to either the goal or even significant subgoals. Thus it does not have recourse to "higher level" problem-solving strategies such as means–ends analysis (Newell & Simon 1972), problem abstraction (Knoblock 1995), or the use of multiple problem spaces (Newell 1990). The model is an attempt to explore the power of a basic mechanism that we argue plays a significant role, without necessarily being a complete model of human problem solving, particularly of those processes that operate at a higher, more conceptual level. An additional difference is that human problem solvers, perhaps as a result of relying on such conceptual knowledge, "cool" their random move making faster than the model does. As a result of these differences, the model takes many more moves than humans do to learn to solve the problems, and also learns more slowly over iterations. The model is thus not a full cognitive model, but rather an attempt to produce intelligent behavior that is based on a set of simple but cognitively plausible mechanisms.

The model can, however, be made to perform much more efficiently than many of the results illustrated in the paper. The conservative approach we took maintained the same experiments throughout all example isomorphs. For many of the problems, faster initial annealing accelerations and more aggressive accelerations at an earlier transition state would generate move records much closer to those of humans. As well, it is likely that humans might not only have a type of context specificity or sensitivity that allows them to modify their exploratory behavior, but also possibly start with one heuristic and then switch to another. Such dynamically modified heuristics are a subject of future research but initial results indicate that they do improve the model's performance.

Computationally, the machine learning issues that are explored with the model include: the use of simulated annealing, the use of the objective function to represent all knowledge acquired about the problem and the effective use of this knowledge to guide move selection, the ability of the model to learn about the environment through the progressive learning of bidirectional pair-wise associations between adjacent states in the problem space without overt "teaching", the demonstration of its adequacy for performance in the domains tested, and our preliminary exploration of the relationship between the state of learning and the rate of annealing for optimal performance. These represent interesting issues in computational intelligence with possible mappings onto human cognition.

ACKNOWLEDGMENTS

The authors thank Justin Boyan, Kenneth Brown, Simon Szykman, and Sebastian Thrun for their helpful comments on this work.

REFERENCES

AFRIAT, S. N. 1982. Ring of linked rings. Duckworth, London.

AGOGINO, A. M., and A. S. ALMGREN. 1989. Techniques for integrating qualitative reasoning and symbolic computation in engineering optimization. Engineering Optimization, **12**:117–135.

BAIRD, L. 1995. Residual algorithms: Reinforcement learning with function approximation. *In* Machine Learning: Proceedings of the Twelfth International Conference. Morgan Kaufmann, pp. 30–37.

BARTO, A. G., S. J. BRADTKE, and S. P. SINGH. 1995. Learning to act using real-time dynamic programming. Artificial Intelligence Journal, **72**:81–138.

BARTO, A. G., R. S. SUTTON, and J. C. H. WATKINS. 1992. Learning and sequential decision making. *In* Learning and Computational Neuroscience: Foundations of Adaptive Networks. *Edited by* M. Gabriel and J. Moore. MIT Press, Cambridge, MA.

BROADBENT, D. E., and D. C. BERRY. 1988. Interactive tasks and the implicit-explicit distinction, British Journal of Psychology, **79**:251–272.

CAGAN, J., and A. M. AGOGINO. 1987. Innovative design of mechanical structures from first principles. AI EDAM: Artificial Intelligence in Engineering, Design, Analysis and Manufacturing, **1**(3):169–189.

COHEN, A., R. I. IVRY, & S. W. KEELE. 1990. Attention and structure in sequence learning. Journal of Experimental Psychology: Learning, Memory and Cognition, **16**:17–30.

FEIGENBAUM, E. A. 1977. The art of artificial intelligence: Themes and case studies in knowledge engineering. *In* Proceedings of the IJCAI-5, Cambridge, **2**:1014–1029.

FULCHER, E. P. 1992. Reinforcement learning: On being wise during the event. *In* Artificial Neural Networks. *Edited by* I. Aleksander and J. Taylor. Elsevier, **2**:925–928.

HUANG, M. D., F. ROMEO, and A. SANGIOVANNI-VINCENTELLI. 1986. An efficient general cooling schedule for simulated annealing. ICCAD-86: IEEE International Conference on Computer-Aided Design—Digest of Technical Papers, Santa Clara, CA, pp. 381–384.

KIRKPATRICK, S., C. D. GELATT, JR., and M. P. VECCHI. 1983. Optimization by simulated annealing. Science, **220**(4598):671–679.

KNOBLOCK, C. A. 1991. Automatically generating abstractions for problem solving. Ph.D. dissertation. Carnegie Mellon University, Pittsburgh, PA.

KOTOVSKY, K., J. R. HAYES, and H. A. SIMON. 1985. Why are some problems hard: Evidence from Tower of Hanoi. Cognitive Psychology, **17**:248–294.

KOTOVSKY, K., and H. A. SIMON. 1990. What makes some problems really hard: Explorations in the problem space of difficulty, Cognitive Psychology, **22**:143–183.

MCCLELLAND, J. L., and D. E. RUMELHART. 1986. Parallel Distributed Processing, Vol. 2. MIT Press, Cambridge, MA.

NEWELL, A. 1990. Unified Theories of Cognition, Cambridge University Press, Cambridge, MA.

NEWELL, A., and H. A. SIMON. 1972. Human Problem Solving. Prentice Hall, Englewood Cliffs, NJ.

PAPALAMBROS, P., and D. J. WILDE. 1988. Principles of Optimal Design. Cambridge University Press, New York.

REBER, P. 1993. Working memory, learning and transfer of non-verbalizable knowledge: Solving the Balls and Boxes Puzzle, Ph.D. dissertation. Carnegie Mellon University, Pittsburgh, PA.

REBER, P., and K. KOTOVSKY. 1992. Learning and problem solving under a memory load. *In* Proceedings of the 14th Annual Conference of the Cognitive Science Society, Bloomington, IN, pp. 1068–1073.

RICH, E. 1983. Artificial Intelligence, McGraw-Hill, New York.

RUGER, H. A. 1908. The Psychology of Efficiency. Archives of Psychology, **2**(15).

RUMELHART, D. E., and J. L. MCCLELLAND. 1986. Parallel Distributed Processing, Vol. 1. MIT Press, Cambridge, MA.

SABES, P. 1993. Approximating Q-values with basis function representations. *In* Proceedings of the Fourth Connectionist Models Summer School.

SUTTON, R. S. 1988. Learning to predict by the methods of temporal differences. Machine Learning, **3**:9–43.

SUTTON, R. S. 1990. Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. *In* Proceedings of the 7th International Conference on Machine Learning, pp. 216–224.

THRUN, S. B. 1992. Efficient exploration in reinforcement learning. CMU-CS-92-102, School of Computer Science, Carnegie Mellon University, Pittsburgh, PA.

THRUN, S. B., and K. MÖLLER. 1992. Active exploration in dynamic environments. *In* Advances in Neural Information Processing Systems 4. *Edited by* J. E. Moody, S. J. Hanson, and R. P. Lippmann. Morgan Kaufmann, San Mateo.

VANDERPLAATS, G. N. 1984. Numerical Optimization Techniques for Engineering Design With Applications. McGraw-Hill, New York.

WILLIAMS, B. C., and J. CAGAN. 1996. Activity analysis: Simplifying optimal design problems through qualitative partitioning. Engineering Optimization, **27**:109–137.

WATKINS, C. 1989. Learning from delayed rewards, Ph.D. dissertation, Cambridge University.

WATKINS, C., and P. DAYAN. 1992. Technical note: Q-learning. Machine Learning, **8**(3/4):279–292.