Processing Constraints and Problem Difficulty: A Model

Kenneth Kotovsky
kotovsky@psy.cmu.edu

Nicholas Kushmerick
kushmerick@psy.cmu.edu

Psychology Department
Carnegie Mellon University
Pittsburgh, PA 15213

Abstract

In this paper we examine the role played by working memory demands in determining problem difficulty during the solution of Tower of Hanoi Problem isomorphs. We do so by describing a production system model that accounts for subjects' performance on these problems via a dynamic analysis of the memory load imposed by the problem and of changes in that load during the problem solving episode. We also present the results of detailed testing of the model against human subject data. The model uses a highly constrained working memory to account for a number of features of the problem solving behavior, including the dichotomous (exploratory and final path) nature of the problem solving, the relative difficulty of the problems, the particular moves made in each state of the problem space, and the temporal patterning of the final path moves.

Introduction: Human Problem Solving Performance

One of the large issues in problem solving research is the question of what makes a problem hard or easy to solve. This issue is of interest both to educators interested in designing instruction so as to more effectively understand and promote the growth of problem solving skills, and to cognitive scientists interested in the information processing involved in problem solving. It is from the latter perspective that we attempt here to examine how the processing demands imposed by different problem features interact with limited working memory to determine problem difficulty. We do so by describing a model of problem solving that accounts for subject performance on a set of Tower of Hanoi Problem isomorphs. The model incorporates a dynamic analysis of problem processing demands in the form of memory load, the effects of changes in that load over the course of problem solving, and the limitations inherent in short term memory. In attempting to account for problem difficulty by means of the processing demands or "load" imposed by various features of the problem, the model includes design parameters based on fairly precise data obtained from a number of experiments with human subjects. The model is tested via detailed comparisons of its behavior against that of human subjects.

While the work presents a detailed analysis of the model's application to a particular domain, a set of isomorphs of the Tower of Hanoi Problem, the implications of the work are not limited to that problem domain, but rather address a set of issues applicable to a wide variety of problem solving situations.

In 1974, Hayes and Simon published the first of a series of articles on Tower of Hanoi problem isomorphs (Hayes & Simon 1974, 1977, Simon & Hayes 1976). The set of isomorphs they investigated consisted of problems that had the same problem state space as the three disk Tower of Hanoi Problem. Two problem types were labelled either "Monster Move" problems because they involved monsters passing globes back and forth, or "Monster Change" problems because they had monsters changing the sizes of globes they were holding. A major finding was that isomorphic problems could differ significantly in difficulty. In particular, they obtained difficulty (solution time) ratios averaging about 2:1 between Monster Change and Monster Move problems. Kotovsky, Hayes, and Simon (1985), extended the investigation to a broader array of problems, obtaining difficulty ratios of up to 16:1 for their hardest/easiest pair of isomorphs; the easiest being the original Tower of Hanoi disk/peg problem.

The different problems are defined by the subjects' internal problem representation which included the move operators that define move legality. This representation is engendered by either an external depiction of important features of the problem such as the physical pegs and disks of the Tower of Hanoi problem, or a "cover story" that describes the monsters, their globes and the moves or changes the monsters can make with their globes. Table 1 presents the rules for two problems that were used to produce much of the data discussed here, a Monster Move Problem, and a Monster Change Problem. The cover story that defined the problem had three different sized monsters trying to move or change globes to convert an initial arrangement of monster—globe pairings into a final arrangement.

<u>*Monster Move Problem*</u>
1. Only one globe may be transferred at a time.
2. If a monster is holding two globes, only the larger of the two may be transferred.
3. A globe may not be transferred to a monster who is holding a larger globe.
<u>*Monster Change Problem*</u>
1. Only one globe may be changed at a time.
2. If two globes have the same size, only the globe held by the larger monster may be changed.
3. A globe may not be changed to the same size as the globe of a larger monster.

Table 1: Problem Rules

The fact that the problems are isomorphic in the above work removes problem search space structural features (branchiness of the problem space, length of the minimum solution path, etc.) as possible sources of the large difficulty differences that were discovered. In an attempt to discover what were the sources of the difficulty differences, two findings emerged from the Kotovsky, Hayes and Simon (1985) work that are the starting points for the research reported here. Those findings are: (a) the crucial role played by the move

This research was supported in part by the Personnel and Training Research Programs, Psychological Sciences Division, Office of Naval Research, under contract N00014-85-K-0696.

operator and its interaction with human information processing limitations in determining problem difficulty, and (b) the discovery of a dichotomous pattern of moves that occurred as people moved through the problem space to reach a solution to the problem.

The first finding was that more difficult problems employed move operators that imposed more of a processing load via the number of entities (monsters and globes) and the number of separate loci that had to be imaged in testing the legality of a move. A ranking of individual moves or problems by the number of separate entities and loci that had to be simultaneously held in working memory was predictive of both the difficulty of making individual moves and also of overall problem difficulty. Thus, in one experiment, when subjects were asked to judge the legality of single moves that were presented tachistoscopically, their response latencies were correlated with the number of entities that had to be imaged in order to make the judgement. For example, in the Move Problem, subjects' judgements were relatively fast when they compared the sizes of two globes held by the same monster (Rule 2) and relatively slow when they compared the sizes of two globes held by different monsters (Rule 3). The explanation was that comparing two globes when they were at two separate loci entailed carrying one in memory and thus imposed an added unit of memory load. An even harder comparison occurs in the Change Problem where subjects had to imagine changing the size of a globe, and then test the imaged size against the size of another globe that was held by either the same monster (Rule 2) or another monster (Rule 3). In that case, the imposed load was higher because of the need to imagine the size change before doing the comparison. There were positive correlations between the processing load imposed by the move operators, operator application time, and problem difficulty. In the current work, this load is termed the "image" load.

The second finding was that subjects' move making was dichotomous, consisting of an initial, "exploratory phase" (whose length differed across problems) in which moves were slow, particularly in the harder isomorphs, and resulted in no net progress toward the goal, followed by a subsequent "final path" phase (virtually identical across problems) where moves were uniformly fast, relatively error free, and quickly led to the solution. The difference was due to subjects in the final path phase being able to plan and execute subgoals (as evidenced by final path patterns of move latencies), whereas earlier in the problem, they were not able to.[1] The empirical evidence suggested that subjects attained this ability to subgoal by reducing the memory demands involved in learning and applying the problem move legality tests during the course of solving the problem (Kotovsky et al 1985).

This "rule learning" effect on the memory load imposed by learning and remembering the problem rules or move legality tests occurred as people made moves. They started out making errors (illegal moves) because of starting to move before they knew the rules, but became progressively better at using the problem rules as they received practice. The difficulty of learning the move rules presented in isolation was determined, and on that basis and primarily via a linguistic analysis of the statement of the problem rules, the relative processing demands of the various problems' rules were determined. This "rule load" decreased as the problem solving proceeded and the rules were learned.

The result of this rule learning was that subjects' moves in the state space regularly exhibited the surprisingly dichotomous "exploratory" and "final path" pattern described above. The exploratory moves were made slowly, they occupied the major phase of the problem solving time, and they were more difficult (took much longer) in the harder isomorphs. Furthermore, subjects were as far from the goal after making these moves as they were at the beginning of the problem. In contrast, the final path moves (after subjects had compiled the problem rules) were relatively error free, were made very rapidly, were executed at a similar speed across all problem isomorphs, and led almost immediately to a problem solution. This dichotomous pattern of slow or difficult move making that made no net progress, and whose length reflected the relative difficulty of the problems, followed by a rapid dash to a solution in the last minute or so of the solution process, regardless of isomorph, is characteristic of a number of other problems as well (Kotovsky & Simon 1989).

This discovery of distinctive exploratory and final path phases provided a plausible link between move operator difficulty and problem difficulty. The issue was that although the processing load imposed by the move operators predicted the ordering of isomorph difficulty, the differences in move time were not great enough to account for the very large differences in problem solution time. The linkage was that during the exploratory phase, the difficulty of remembering and applying the rules (legality tests) prevented subjects from planning move sequences; even those of length two, and prevented their making progress toward the goal (moving down the final path).[2] An information processing analysis of the load imposed in making goal-subgoal pairs of moves showed that the harder, Monster Change problems imposed much higher memory loads than the Monster Move problems. A single move in the Monster Change problem always requires one more entity to be imaged than an equivalent move in the Monster Move problem. The direction of this difference is what would be expected from

[1]There is evidence that subjects, just prior to the final path, were not able to successfully execute a subgoal-goal pair of moves when they were in a situation that called for such a move pair. The evidence was obtained from an analysis of moves made when subjects were in a subgoal situation just prior to the final path. This was defined as a locus in the problem state space that required a subgoal move before a progress-making move could be made. Even when the analysis was restricted to the end of the exploratory phase, the move latencies indicate that the subjects did not execute the moves as a move pair at a frequency greater than chance. If they encountered the same situation during the final path phase, it was correctly executed as a move-pair with a frequency well above chance (Kotovsky & Fallside, 1989).

[2]2An alternative explanation for the exploratory-final path dichotomy has been provided by Anderson (1990) as part of his rational analysis of cognition. His very interesting analysis is predicated on the relative costs of hill-climbing versus means-ends analysis, rather than the near-impossibility of means-ends analysis within the memory limitations we believe are operating. A detailed comparison of the two models is beyond the scope of this paper.

the pattern of move operator difficulties found for individual moves in the various isomorphs. When subjects tried to plan pairs of moves, the load differences were magnified.

To test the hypothesis that subjects were planning move-pairs during the final path phase, the move latencies of the final path were analyzed for evidence of subgoal-goal pairs of moves.[3] The analysis indicated that the subjects solved these five-move minimum path problems in two rapid sequences of moves, corresponding to two goal-subgoal pairs of moves, followed by a final fast move of the last globe to its final goal position. The pattern of move latencies was long-short followed by long-short-short. This is what we would expect if the subject attained the ability to plan and execute a subgoal-goal move pair, as contrasted with the pattern if they made individual moves, or planned and executed all five final path moves as a compiled whole. The long-short pattern of move latencies is presumably due to a planning-plus-move step, followed by a move step. Not only did the subjects for whom final path move latencies were recorded exhibit the long-short and long-short-short temporal patterns, but as noted earlier, subjects in the exploratory phase did not.

The Model

The model we report on attempts to account for the above findings with a small number of mechanisms and assumptions. The model consists of a production system[4] containing 21 productions that incorporates the following features: a limited working memory, a state memory that records recent moves and acts to prevent looping, a representation of the relative difficulty of the problem rules, a chunking or learning mechanism that operates on the problem rules and a derived measure of move latencies. These major features of the model are implemented as follows:

Working Memory is the common "work-space" of the model. It is very limited, able to hold, in the "standard" version of the model, about five entities or chunks. Its contents are the model's current subgoals (the overall goal of getting each monster its own sized globe is assumed to be in a longer term store), the problem rules, and the entities (globes, monsters) that have to be held in mind or imaged in order to be compared as part of the rule-legality checks during move planning and execution. The previously described image load acts here. The load imposed by various items in its contents at any point in time are calculated on the basis of empirical data acquired from the experimental work discussed above. An additional feature is that if this memory is overloaded, every item it contains is a candidate for being displaced, with equal likelihood. There is a working memory *breakpoint*, an inflection point near which each entity currently in working memory becomes separately eligible for displacement, with a probability obtained as follows:

$$p(\text{displacement}) = \frac{1}{1 + e^{breakpoint - load}}$$

where

load = "image load" (the number of imaged entities involved in the move or planned move)
+ "spatial load" (the number of loci that had to imaged in the move or planned move)
+ "subgoal load" (the number of subgoals)
+ "rule load" (remembering the current move rule)

The breakpoint is thus the point where the p(displacement)=$^1/_2$. The function rapidly goes to zero when the load is less than the breakpoint and to one when the load moves above the breakpoint. This relatively steep function is meant to approximate the behavior obtained in tests of memory span where the break is fairly sharp as the limit of a person's span is approached. For the data presented here, the breakpoint was chosen to be five. (It is possible to vary this parameter to match the performance of different subjects.) This stringently limited ability to manage information in making or planning moves is a major determinant of the model's behavior, as it is the behavior of the subjects. This is particularly true during the exploratory phase of the problem solving.

State Memory is posited to be a somewhat longer term recognition memory for recently made moves in the problem space. It functions to prevent looping (repeatedly traversing the same portion of the problem state space) or perseverating on trying the same move over and over again. It is implemented as a separate store to which the current move (and the state from which it is being attempted) is added (with probability = p) each time a subgoal or goal move is planned or made, and from which the oldest element is displaced (with probability = q) on each production firing. The probabilities, p and q, were set equal to 0.5 and 0.1 in the model data reported here.

The rule load designates the immediate memory load (number of chunks) imposed by remembering the problem move-legality rules. This was calculated primarily on the basis of a clausal analysis of the statement of the rules, bolstered by some data on the learning time required by subjects to learn to paraphrase or repeat the rules. The rule "loads" imposed by the problem rules in the order of source rule and destination rule, were for the Move Problem, 3 chunks and 4 chunks, and for the Change Problem, 4 chunks and 5 chunks respectively. These were the initial loads imposed by the rules. As subjects (human or computer) started making moves, they often made errors in which case they were forced to take back the illegal move and were re-exposed to the statement of the violated rule. This learning experience, by which subjects eventually learned to remember and apply the rule could eventually reduce the rule load to one. It was implemented via the following mechanism.

Rule chunking is a learning mechanism that is invoked each time a move rule was violated. Upon each violation, the relevant rule was "chunked" (ie. its imposed load was reduced to 1 from its initial value of 3, 4 or 5) with some probability. For the results reported here, the probability was set equal to 0.6 and 0.3 for the Move and Change problem respectively. This chunking in the model corresponded to the automatic restatement of any violated rule to human subjects each time it was violated.

Try Something Else is a mechanism that gets invoked

[3]An example of a subgoal-goal move pair is that encountered in the Move Problem when trying to move the medium globe to the medium monster when the medium monster is already holding the large globe. The completion of the move requires the subgoal of "clearing" the medium monster by moving the large globe elsewhere, followed by the goal move of moving the medium globe to the medium monster.

[4]The production system is written in a modified version of Grapes (Sauers & Farrell, 1982).

whenever the model's planning function resulted in its looping (perseverating) when trying to plan a particular subgoal-goal move pair or it's trying a move that had recently been tried (ie. was in state memory). The mechanism operated by randomly choosing a move from among the set of all "top goal" moves (ie. moves that would move a globe to its final or "home" position), + all trivial moves (ie. any moves that do not require legality checks at either the source or destination), + one randomly chosen move that is not currently in state memory.

Move latency is determined by assuming that the amount of time it takes a production to fire is a function of the complexity of a production (the number of elements on its condition side) and the working memory load extant at the time of the productions firing. The calculation assumes that the larger the current contents of working memory (as a percentage of the breakpoint capacity) the less resources available for the demands of the particular production and as a result, the slower the firing. This calculation of a memory-conditioned production firing time allows us to derive a measure of move latency by simply adding the times of all the productions that fire between a pair of moves. A move latency measure is needed if we are to compare the temporal patterns of the model's moves with the characteristic temporal patterns obtained from human subjects' data.

Results

The operation of the model consists of trying to reach the specified goal state, which has each monster ending up holding his/her corresponding sized globe, by trying to make moves that will get the correct globes to their appropriate monsters. When moves are blocked by the problem's rules, the model tries, as subjects do, to plan and execute subgoal moves that will clear the way for desired goal moves. In the beginning of solving the problem, the model often makes illegal moves (by forgetting one or more of the move legality checks). Further along in the problem solving, the model, as it remembers to check the legality of contemplated moves, tries to "unblock" desired moves by executing subgoal moves, but is unable to accomplish this because of the memory load involved in planning and making the pair of moves. This results in repeated attempts to make moves that are repeatedly frustrated, more error prone move-making (when a legality check is one of the items displaced from the overloaded working memory), or a retreat to an easier but less useful move. Toward the end of the problem-solving episode, the model frequently exhibits the human subjects' "final path" type of behavior of efficaciously and quickly making moves that achieve the goal state. This is accomplished when it has learned the rules well enough so that they do not overload working memory, thus allowing some planning to occur. As this verbal summary indicates, the model's behavior is roughly similar to that reported for human subjects. We turn now to a more detailed comparison of the behavior of the model with that of people solving the same problems.

Move data is presented in Table 2 which contains the numbers of legal and illegal moves in each of four problems. The problems are change and move problems, each with two different starting positions. The data show that the model solves the various problems in about the same number of moves, making about the same number of errors, as the human subjects, with the fit somewhat better for the Change problems than for the Move problems.

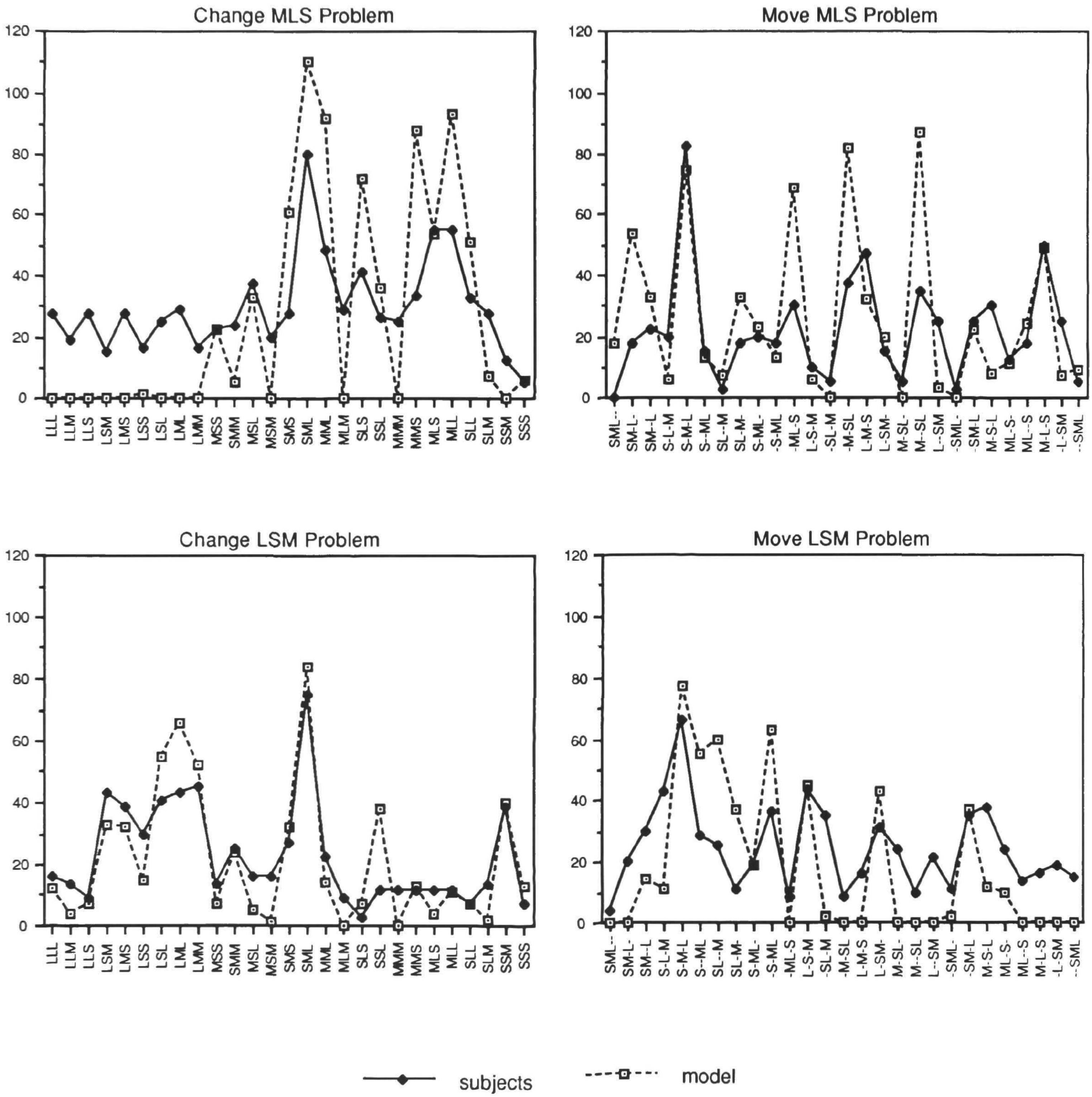
Problem: Start State *Move Type*	**Model**	**Human Subjects**
Change: LSM		
Legal	12	13
Illegal	10	10
Change: MLS		
Legal	17	18
Illegal	11	14
Move: LSM		
Legal	12	16
Illegal	6	9
Move: MLS		
Legal	17	15
Illegal	11	8

Table 2: Solution Path Length (Number of Moves)

Move latencies: Another way in which the behavior of the model can be compared with the behavior of human subjects was by means of the temporal patterning of moves and the rate of progress toward a solution during various phases of the problem solving. The issue is whether the model exhibited the exploratory and final path dichotomy found in most subjects' behavior; that is, whether the model exhibited a pattern of non-progress making moves followed by a rapid dash to a solution. A further question is whether the model also exhibited the evidence of subgoaling that was so crucial an element in the human subjects becoming able to move down the final path to the goal. The derived time measure is at best only an indicator of relative times, and its results should be interpreted with caution. Observation of the model's performance did reveal the characteristic dichotomy of exploratory and final path phases of moving. That is, there was a period, often quite extended, during which the model made non-optimal moves that contained a high frequency of illegal moves and resulted in no net progress toward the goal. This was followed by a rapid dash to a solution, during which the model made very few errors, and the moves were optimally directed toward the goal. Further, the temporal patterning of moves at the end of the problem solving episode for an absolute majority of solutions in which the model exhibited final path behavior did exhibit the L-S-L-S-S pattern of move latencies that indicated the subgoal-goal move pattern so evident on the subjects' final path. This evidence of final path behavior captures a major feature of the subjects' behavior; the learning of the problem rules and resultant acquisition of a bit of expertise at using them to plan moves. This learning occurs during the exploratory phase of the problem solving episode and allows the planning of subgoal-goal move pairs, thus permitting the subject, human or machine, to plan moves within the sharp constraints imposed by a very limited working memory.

Move pathways and state visits: One question that can be asked of the model's performance, in addition to how many moves it took to solve the problem, is whether it moves through the same portions of the problem state space as are traversed by the human subjects. The answer to this question is complicated by the fact that there is great diversity in the pathways chosen by different subjects and by different runs of the model. We constructed a display system that mapped out sets of subjects and sets of model runs and displayed the moves

Figure 1: Number of Visits to Each State of the Problem Space, Model and Subjects



(preserving their latencies) through the problem space from start to goal. In general, the agreement seemed to be high; the diversity in the subjects' moves was matched by the diversity found among different runs of the model. A more controlled comparison was performed by comparing the number of visits to each of the 27 states (monster-globe arrangements) in the problem space made by our human subjects with those made by a set of model runs for four different Monster Problems. The four problems were two Move Problems and two Change problems, where the pairs of each type were defined by different problem start states.[5] The data is contained in Figure 1, where it can be seen that the state visits of the model closely approximate those of the subjects.[6]

Move comparisons: An even more detailed comparison of model and subject behavior can be made by comparing the particular moves made from each of the 27 problem space locations. This provides 105 comparisons (the number of each of the three legal moves together with the number of illegal moves from each state in the problem space).[7] The average correlation between moves chosen for the model and humans across the four isomorph types is 0.73, with the four problems' correlations ranging from 0.60 to 0.87. An even finer grain analysis can be made by considering each type of illegal move separately. That is, from each state in the problem space, there are three possible illegal moves that can be made by moving one globe. (If we allow the simultaneous moving of more than one globe, then a move can be made between any state in the search space and any other state, producing a total of 27×26 possible moves). The experiment constrained subjects so they could not move two globes at once (something that rarely happens even in unconstrained experiments), so the real limit on the number of possible moves is 27×6 (3 legal + 3 illegal moves) from each state, except for the three states where only two legal moves possible. Comparing the model with the subjects on the choice of moves for all 159 possible moves, the correlations range from 0.57 to 0.86, with a mean of 0.70. In evaluating this quite positive result, one possibility is that the structure of the state space itself constrains both model and subjects to certain states and moves because of the particular transitions that are allowed. In order to test this possibility, we constructed a random move generating model that randomly chose one of the six possible one-globe moves, and correlated its performance with that of the subjects. The result was that there is a correlation between the random model and the human performance data, but it is small compared to that found with the actual model. Thus the correlations between the random model and the human data ranged from 0.37 to 0.50 for the different problems, with a mean of 0.44. In addition, the random model accounted for none of the temporally patterned final path behavior that the model reproduced. Thus while the structure of the state space does play some role in constraining the behavior of those moving through it, it is not nearly constraining enough to serve as the explanation of the results we have obtained.

Our model, which incorporates the findings that have been made about the difficulties imposed by various features of the problem together with a sharply limited working memory provides a good account of a number of aspects of people's behavior on these problems. This is true whether the comparison is of the portion of the problem search space visited on a subject by subject basis, the frequency of particular state visits, the aggregate frequency of particular moves, the relative frequency of legal and illegal moves, the frequency with which all possible moves are made or the temporal patterns exhibited by the model in the final stages of its solution of the problem. These and other detailed comparisons of the model with human data allow a rigorous testing of the hypotheses that underlie the model's design, and provided convergent evidence for the conclusions reached in previous work on these problems. The work with the model is still in its early stages. We have yet to investigate transfer from one problem to another, nor have we used the power of the model to systematically explore factors that account for individual differences in performance or for performance on a wider array of problem isomorphs. Work on these issues is currently in progress.

[5] The subject data consists of sets of between 10 and 20 subjects per problem condition. The number of runs of the model was set at 25 to insure an adequate sampling, and the subject data were normalized to 25 in order to make the data sets comparable.

[6] State visits included visits to a state in the problem space resulting from either a legal or an illegal move. If only visits resulting from legal moves are considered, the fit is comparable.

[7] Three of the problem space loci allow only 2 legal moves, thus accounting for the 105 instead of 108 possibilities.

Acknowledgements

We wish to express our gratitude to Herbert A. Simon and J.R. Hayes for their contributions to the conceptual development of the work reported here, and to Aaron Benjamin, for his help in running the simulations and analyzing the data.

References

Anderson, J.R. (1990). *The Adaptive Character of Thought.* Hillsdale, NJ: Erlbaum.

Hayes, J.R., & Simon, H.A. (1974). Understanding Written Problem Instructions. In Gregg, L.W. (Ed.), *Knowledge and Cognition.* Hillsdale, NJ: Erlbaum.

Hayes, J.R., & Simon, H.A. (1977). Psychological Differences Among Problem Isomorphs. In Castellan, N.J., Pisoni, D.B., & Potts, G.R. (Eds.), *Cognitive Theory.* Hillsdale, NJ: Erlbaum.

Kotovsky, K., & Fallside, D.C. 1989. Representation and Transfer in Problem Solving. In Klahr, D., & Kotovsky, K. (Eds.), *Complex Information Processing: The Impact of Herbert A. Simon.* Hillsdale, NJ: Erlbaum.

Kotovsky, K., Hayes, J.R., & Simon, H.A. (1985). Why Are Some Problems Hard? Evidence From the Tower of Hanoi. *Cognitive Psychology* 17:284-294.

Sauers, R., & Farrel, R. November, 1982. Grapes User's Manual, Technical Report, Department of Psychology, Carnegie Mellon University.

Simon, H.A., & Hayes, J.R. 1976. The Understanding Process: Problem Isomorphs. *Cognitive Psychology* 8:165-190.