Artificial Intelligence: The Heuristic Programming Approach

JAMES R. SLAGLE, Ph.D.
Johns Hopkins University

© 1971

McGraw-Hill Book Company
NEW YORK ST. LOUIS SAN FRANCISCO
DÜSSELDORF LONDON MEXICO
PANAMA SYDNEY TORONTO

8 The General Problem-solver Program

A program called General Problem Solver (GPS) solved a wide variety of problems (Newell, Shaw, and Simon, 1960; Ernst and Newell, 1967; Ernst and Newell, 1969). Of the many versions that were programmed, only the "final" version (1967) is described in this chapter. Ernst and Newell (1967) say that the next version, if there is one, should be an entirely new program. J. R. Quinlan (1969) has written and experimented with a successor of an earlier version of GPS. GPS is a truly unified and multipurpose program and not merely a conglomeration of programs, each specialized to perform one task.

So that the reader can think concretely, some examples of the external behavior of GPS are now given. Suppose that GPS is given in a suitable language the following "monkey task." (This task was first specified by John McCarthy, 1963.) A monkey and a box are in a room, and some bananas are hanging from the ceiling. The monkey can walk, move the box, and climb on the box. He can get the bananas if he is on the box and the box is under the bananas. What should he do so that he gets the bananas? After

working for less than 4 min, GPS gives the answer that the monkey walks to the box, moves it under the bananas, climbs it, and gets the bananas. To enable GPS to perform some simple indefinite integrations, it is given several very simple rules for performing some arithmetic, algebra, differentiation, and integration. After working for about 3 min on the specific task, $\int te^{t^2}\,dt$, GPS prints out the answer $\frac{1}{2}e^{t^2}$. After working for about 6 min on another specific task, $\int(\sin^2 ct \cos ct + t^{-1})\,dt$, GPS prints out the answer $(\sin^3 ct)/3c + \ln t$.

In general, the way that GPS works is that it searches a goal tree by trying to transform a goal into easier subgoals. Here and elsewhere we say "GPS," when actually it would be more precise to say "the problem-solving executive program," which is the top-level program in GPS. The problem-solving executive program of GPS selects subgoals, selects methods, and applies the selected methods to the selected subgoals. A task is given to GPS in terms of objects, operators, and differences. GPS uses a general technique called *means-ends analysis*, in which it selects an operator as a function of the difference between the present object and the desired one. Differences are taken between what is given and what is desired. For example, a difference between two objects is a feature which occurs in one object but does not occur, or has a different form, in the other object. GPS uses the difference to select an operator relevant to reducing that difference.

Some of the findings and conclusions concerning GPS are the following. GPS worked on 11 widely different tasks. It was able to perform tasks involving inductive and deductive reasoning. It performed the tasks very slowly. Means-ends analysis was found to be very effective on some tasks but not on others. The fact that the general techniques used by GPS resemble those used by humans is very important to those researchers for whom the psychological purpose of Artificial Intelligence is very important. GPS needed to use only four types of goals. It should have ways of viewing a task as a whole, and the way in which GPS selects the next goal to be tried will need to be improved, especially when the goal trees become large. GPS may be the forerunner of a program that will, in some significant sense, perform the task of doing its own learning.

8.1 THE REPRESENTATION OF A TASK

Each task given to GPS consists of a task environment and a specific task. Each task environment consists of all the information common to tasks of a certain type, for example, the information common to all integration tasks. Each task environment consists of operators, differences, a difference-ordering, a table of connections, details for matching objects, and miscellaneous information. Each specific task consists of a top goal and its objects. Table 8-1, which specifies the monkey task, illustrates a specific task and most parts

Table 8-1 The specification for GPS of the monkey task

I. Task Environment
  A. Miscellaneous information: The set of places (on the floor) = {place 1, place 2, under the bananas}
  B. Operators
    1. CLIMB =
      a. Pretest: The monkey's place is (the same as) the box's place.
      b. Move: The monkey's place becomes on the box.
    2. WALK =
      a. Variable: x is in the set of places.
      b. Move: The monkey's place becomes x.
    3. MOVE BOX =
      a. Variable: x is in the set of places.
      b. Pretests
        1. The monkey's place is in the set of places.
        2. The monkey's place is the box's place.
      c. Moves
        1. The monkey's place becomes x.
        2. The box's place becomes x.
    4. GET BANANAS =
      a. Pretests
        1. The box's place is under the bananas.
        2. The monkey's place is on the box.
      b. Move: The contents of the monkey's hand become the bananas.
  C. Differences
    *D*1 is the monkey's place.
    *D*2 is the box's place.
    *D*3 is the contents of the monkey's hand.
  D. Difference Ordering: *D*3 is more difficult to reduce than is *D*2 which is more difficult to reduce than is *D*1.
  E. TABLE OF CONNECTIONS = All operators are desirable for reducing all differences.

II. Specific Task
  A. TOP GOAL = Transform the initial OBJ into the desired OBJ.
  B. Objects.
    1. INITIAL OBJ =
      a. The monkey's place is place 1.
      b. The box's place is place 2.
      c. The contents of the monkey's hand are empty.
    2. DESIRED OBJ = The contents of the monkey's hand are the bananas.

of a task environment. The table of connections and details for matching objects are not illustrated in this table, but are explained fully in Section 8.4. In the computer, tasks are specified for GPS in a language that is similar to but more formal than the language used in Table 8-1.

8.2 THE PROBLEM-SOLVING EXECUTIVE PROGRAM

The task specification is given to the problem-solving executive program, whose main steps are given below. The antecedent goals and the difficulty of a goal, which are mentioned in the steps, are defined in Section 8.4. Generally speaking, the heart (the inner loop, steps D through L) of the procedure proceeds locally; that is, in a fairly natural way it generates subgoals and moves between supergoals and subgoals (and to antecedent goals) according to successes and failures. The exceptions to this "local" characteristic are step I and step D(1). Steps A and B represent the "desperation" case, that is, the case in which all the methods for the top goal have been exhausted [see step D(1)]. Just which goal is selected in step B will not be explained here because it is of no intrinsic interest and the case seldom occurs. The following executive procedure is given a task and starts in step C.

A. If all methods for all goals have been exhausted, print this fact and stop.
B. Select a goal that has unexhausted methods, and go to step E.
C. Select the top goal.
D. If all the methods for the selected goal are exhausted,
  1. If the selected goal is the top goal, go to step A.
  2. If the selected goal is not the top goal, select its supergoal and go to step D.
E. Select an unexhausted method for the selected goal.
F. Try the selected method on the selected goal.
G. If the selected method fails, that is, it neither achieves the goal nor generates a subgoal,
  1. If the selected goal is the top goal, go to step D.
  2. If the selected goal is not the top goal, select its supergoal and go to step D.
H. If the selected method succeeds in achieving the goal,
  1. If the goal is the top goal, print the answer and stop.
  2. If the achieved goal is not the top goal, select its supergoal and go to step D.
I. The method succeeds in generating a subgoal. If the subgoal is equivalent to some previously generated goal, go to step C.
J. If the subgoal of the selected goal is undesirable because it is more difficult than the supergoal of the selected goal, go to step D.

K. If the subgoal is undesirable because it is more difficult than the antecedent of the selected goal, select the antecedent goal and go to step D.

L. If the subgoal passes this evaluation (steps J and K), select the subgoal and go to step D.

8.3 AN EXAMPLE OF PERFORMING A TASK

Table 8-2 and Figure 8-1 show how GPS uses the executive procedure to perform the monkey task specified in Table 8-1. As with any task, the executive procedure starts at step C, which selects the top goal. Since not all methods have been exhausted, the procedure goes to step E. Since the top goal is a transform-type goal, step E selects one of the methods relevant to transform-type goals. It selects the *transform* method. Note that as a mnemonic we use the same name for a goal type as for one of its methods. The transform method matches the initial OBJ with the desired OBJ. If they already match (there are no differences), the method achieves the goal. If there are differences, the method creates the subgoal of reducing the most difficult difference. Step F tries the transform method on the top goal. Since, in step G, the selected method succeeds, the procedure goes to step H. In step H, the method does not succeed in achieving the goal; the initial OBJ does not match the desired OBJ, since the contents of the monkey's hand are empty in the initial OBJ and are the bananas in the desired OBJ. In step I the method succeeds in generating the subgoal, goal 2, to reduce the difference *D*3 (the contents of the monkey's hand), since this is the most difficult (and in fact the only) difference between the initial OBJ and the desired OBJ. Goal 2 passes the evaluation in steps J and K, since the selected goal (top goal) has neither a supergoal nor an antecedent goal. Step L selects goal 2 and goes to step D.

Table 8-2 GPS performs the monkey task

INITIAL OBJ =
- a. The monkey's place is place 1.
- b. The box's place is place 2.
- c. The contents of the monkey's hand are empty.

DESIRED OBJ = The contents of the monkey's hand are the bananas.

OBJ 2 =
- a. The monkey's place is place 2.
- b. The box's place is place 2.
- c. The contents of the monkey's hand are empty.

OBJ 3 =
- a. The monkey's place is under the bananas.
- b. The box's place is under the bananas.
- c. The contents of the monkey's hand are empty.

OBJ 4 =
- a. The monkey's place is on the box.
- b. The box's place is under the bananas.
- c. The contents of the monkey's hand are empty.

OBJ 5 =
- a. The monkey's place is on the box.
- b. The box's place is under the bananas.
- c. The contents of the monkey's hand are the bananas.

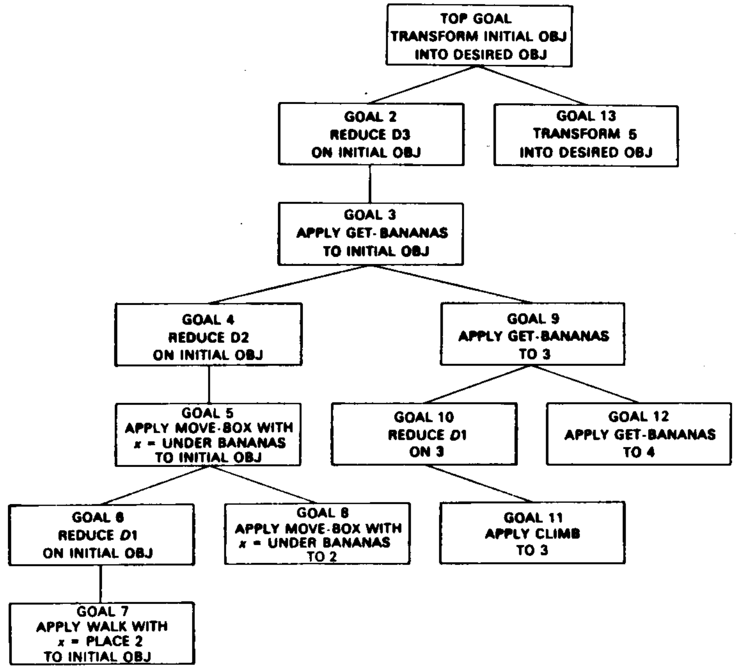

Figure 8-1

With goal 2 selected, step D is uneventful. Step E selects the *reduce* method, one of the methods relevant to reduce-type goals. If successful, the reduce method creates a subgoal of applying to the object an operator relevant to reducing the specified difference. In steps F through I, the method is applied successfully and generates goal 3, to apply the operator GET BANANAS to the initial OBJ. Since goal 3 passes the evaluation in steps J and K, step L selects goal 3 and goes to step D.

With goal 3 selected, step E selects the *apply* method, one of the methods relevant to apply-type goals. The apply method applies the operator directly if it can. If the apply method cannot apply the operator directly, it creates a subgoal to reduce the most difficult difference blocking the application of the operator. Step F tries the method on goal 3. In step H the selected method does not succeed in achieving goal 3; the operator GET BANANAS cannot be applied directly because it requires that the box be under the bananas and that the monkey be on the box. In step I, the method succeeds in generating the subgoal, goal 4, to reduce the difference *D*2 (the box's place), which is more difficult than *D*1 (the monkey's place). Step L selects goal 4 and goes to step D.

In a similar manner, the procedure next generates and selects goals 5, 6, and 7. Assume that the procedure has just selected goal 7 and goes to step D. Step E selects the apply method for goal 7. In step H, the selected method succeeds in achieving goal 7; the monkey walks to place 2; object 2 in Table 8-2 is generated. Step H(2) selects goal 6, the supergoal of goal 7, and goes to step D. Step E selects a method relevant to goal 6. This method achieves a reduce goal when its subgoal has been achieved. In steps F through H(2), the method achieves goal 6, selects its supergoal (goal 5), and goes to step D. Step E selects a method relevant to goal 5. This method for an apply goal generates another apply goal when a reduce subgoal has been achieved. In steps F through I, the method succeeds in generating goal 8, a subgoal of goal 5. Step L selects goal 8 and goes to step D.

Similarly, the procedure next generates and selects goals 9 through 13. Assume that the procedure has just selected goal 13 and goes to step D. Step E selects the transform method for goal 13. Since in step H, object 5 matches the desired OBJ, goal 13 is achieved. Step H(2) selects the top goal and goes to step D. Step E selects a relevant method for the top goal. This method achieves the transform goal if a transform subgoal has been achieved. Since in step H(1) the goal is the top goal, the procedure stops after printing the answer that the monkey walks to place 2, moves the box under the bananas, climbs the box, and gets the bananas.

8.4 GOALS, MATCHING, OPERATORS, AND THE TABLE OF CONNECTIONS

This section, together with the previous three sections, should give the reader a good idea of how GPS operates.

Goals The four kinds of goals handled by GPS are *transform object A into object B*, *reduce difference D on object A*, *apply operator Q to object A*, and *select the element of set S which best fulfills criterion C*. The first three types were illustrated in the example of the previous section. Selection criteria, which are associated with selection goals, are built into GPS in the form of programs. Although such criteria are task-dependent information, they do not appear in the task specification because of limitations in both the task-

specification language and the problem-solving techniques of GPS. As will be seen later, some other task-dependent information is handled the same way because of the same limitations. It is important to note that GPS is less general to the extent that such information must be built in. It is to the great credit of GPS that only a small portion of the task-dependent information was built into GPS in the form of programs.

The antecedent goal of a goal *G* is the goal whose result is used in the statement of *G* and whose supergoal is the same as the supergoal of *G*. In Figure 8-1, for example, goal 2 is the antecedent goal of goal 13; goal 4 is the antecedent of goal 9; goal 6 is the antecedent of 8; goal 10 is the antecedent of 12. No other goal in Figure 8-1 has an antecedent goal.

The difficulty of a goal is the difficulty of the difference associated with the goal. The difficulty of the difference is determined by the difference ordering given in the task environment. Since reduce goals are the only type of goals which have differences directly associated with them, a goal of any other type is considered as difficult as its most difficult subgoal. For example, in Figure 8-1, the top goal does not have a difference directly associated with it, but, after goal 2 was generated, GPS considers the top goal to be as difficult as goal 2 because the difference *D*3 is the most difficult difference detected in matching the initial OBJ with the desired OBJ. Goal 4 was evaluated to be desirable because the supergoal of goal 3 is goal 2 and because the difference *D*2 was not more difficult than *D*3.

Matching Roughly speaking, the match procedure determines if two objects are the same and, if not the same, finds differences between them. The specification of the task environment often includes some details for matching objects, although the specification of the monkey task to GPS had no such details. The problem-solving executive "knows" how to match the kinds of objects which occur in the performance of the monkey task. In the specification of the task environment, the details for matching objects may tell GPS that two objects match only if they are identical. Unfortunately, some other task-dependent details for matching objects must be built into GPS in the form of programs. One program, for example, allows the matching of a variable in one object with an expression in another by means of substituting the expression for the variable.

Operators An operator transforms one object into another. Most operators are specified in the task environment, but others, called immediate operators, are built into GPS as programs. An immediate operator is applied without generating a subgoal to apply it. Each operator given in the task environment is either a move operator or a form operator. A form operator consists of two forms. An object that matches the first form may be transformed into another object according to the second form. A move operator consists of moves and possibly variables, pretests, and posttests. It has been found that move operators are usually more general than are form operators.

Table of Connections In the task environment, the table of connections tells GPS which operators are relevant to reducing each difference. The reduce method, when it is tried on a goal to reduce the difference *D* on an object, chooses an operator which the table says is relevant to *D*. The method then generates the subgoal of applying the operator to the object. Table 8-3 shows the connections that might have been used with the monkey task but were not, as explained below. The table shows that the operators CLIMB, WALK, and MOVE BOX are relevant to reducing the difference *D1*, the monkey's place. The operator MOVE BOX is relevant to reducing *D2*, the box's place. GET BANANAS is relevant to the contents of the monkey's hand. In order to make it a little harder for GPS, the actual table of connections in the monkey task environment told GPS that every operator might be relevant to every difference. However, the reduce method can examine the internal structure of move operators in order to choose operators relevant to reducing each difference.

8.5 TASKS GIVEN TO GPS

GPS was given 11 tasks and spent an average of 17 sec per goal. It is instructive for the reader to compare his performance with that of GPS.

Monkey Task As was seen previously, GPS performed this task by generating 13 goals. It spent (approximately) 221 sec.

Integration Task First, GPS was given the task environment. Six operators told the GPS how to integrate six standard forms; four other operators told how to do simple differentiating; one other operator told GPS that the integral of a sum is the sum of the integrals. To make problem solving feasible, several immediate operators had to be built into GPS in the form of programs. These immediate operators include commutativity and associativity of addition and multiplication, simplification, arithmetic, a simple integration rule, and two

Table 8-3 Possible table of connections for the monkey task

	D1	*D2*	*D3*
CLIMB	*R*		
WALK	*R*		
MOVE BOX	*R*	*R*	
GET BANANAS			*R*

simple differentiation rules. Concerning all the operators in the task environment, Ernst and Newell (1967) say that the transformations of SAINT are more general than the operators of GPS because each transformation corresponds to many operators and that this is a nontrivial difference in the two formulations. The first specific task given to GPS was $\int te^{t^4}\,dt$. To perform this task, GPS generated 11 goals and took 187 sec. The second specific task was $\int (\sin^2 ct \cos ct + t^{-1})\,dt$, which took 22 goals and 374 sec.

Missionaries and Cannibals Task Three missionaries, three cannibals, and a boat are on one side of a river. The only way to cross the river is in the boat, which will hold either one or two people. If the cannibals ever outnumber missionaries on a side of the river, the cannibals will eat those missionaries. (For example, if a missionary and a cannibal arrive at a side where there is a cannibal but no missionaries, the missionary will be eaten.) In every other way, the cannibals will do (operate the boat, return) what they are told to do. How can everybody and the boat get safely to the other side of the river? To perform this task, GPS generated 57 goals and took 969 sec.

Tower of Hanoi Task In the initial position in this puzzle, there are four discs of varying size on the first of three pegs as in Figure 8-2. A move consists of moving the top disc from one peg and putting it on top of the discs (if any) on another peg. A restriction is that no disc may ever be placed on top of a smaller disc. How can all four discs be moved to the third peg? To perform this task, GPS generated 46 goals. Mainly because the differences and difference ordering in this task environment are in some sense optimal, GPS made no mistakes but went directly to the solution.

Task: Proving Theorems in First-order Predicate Calculus GPS was given a task environment for proving theorems using resolution (see Chapter 5). Certain information was built into GPS in the form of programs. The task

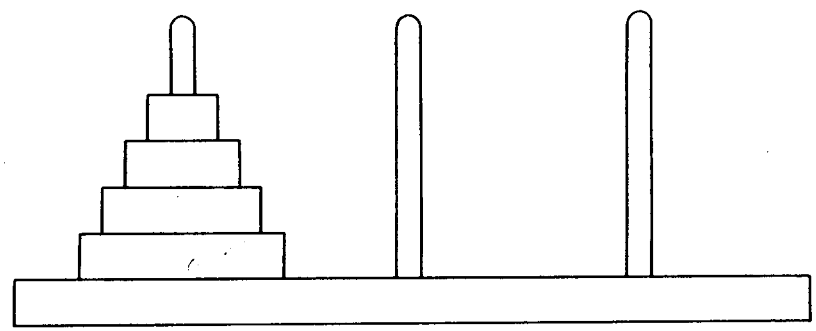

Figure 8-2 Initial position in the tower of Hanoi task.

environment was given in such a way that GPS uses a unit-preference strategy. GPS generated 59 goals in order to derive the contradiction from the following three clauses:

$$\{P(u,y)\}$$

$$\{\sim P(y,f(u,y)) \vee \sim P(f(u,y),f(u,y)) \vee Q(u,y)\}$$

$$\{\sim P(y,f(u,y)) \vee \sim P(f(u,y),f(u,y)) \vee \sim Q(u, f(u,y)) \vee \sim Q(f(u,y),f(u,y))\}$$

Father and Sons Task A father, his two sons, and a boat are on one side of the river. The only way to cross the river is in the boat, whose capacity is 200 lb. Everybody knows how to operate the boat. The father weighs 200 lb, and each son weighs 100 lb. How can they get themselves and the boat across the river? To perform this task, GPS generated 33 goals.

Three Coins Task Of three coins lying on a table, the first and third show tails, and the second shows heads. Each move consists of turning over two of the three coins. The task is to make *exactly* three moves after which all the coins show the same, either heads or tails. To perform this task, GPS generated 10 goals.

Parsing Task In the task environment given to GPS are 10 phrase-structure rules (for a simplified form of English), including the following five rules:

A. A NOUN PHRASE followed by a VERB PHRASE followed by a NOUN PHRASE followed by a period is a SENTENCE.
B. A word that can be used as an adjective is an ADJECTIVE PHRASE.
C. An ADJECTIVE PHRASE followed by a word that can be used as a noun is a NOUN PHRASE.
D. A word that can be used as a noun is a NOUN PHRASE.
E. A word that can be used as a verb is a VERB PHRASE.

The task environment tells GPS that the word "free" can be used as an adjective or verb, that "variables" can be used as a noun, that "cause" can be used as a noun or verb, and that "confusion" can be used as a noun. The specific task given to GPS is to parse the sentence (the initial OBJ):

Free variables cause confusion.

Since "free" can be used as an adjective, GPS uses rule B to transform the sentence into the object, "ADJECTIVE PHRASE variables cause confusion." Since "variables" is a noun, GPS uses rule C to transform this object into the object, "NOUN PHRASE cause confusion." Since "cause" can be used as a verb, GPS uses rule E to transform this object into the object, "NOUN PHRASE VERB PHRASE confusion." Since "confusion" is a noun, GPS

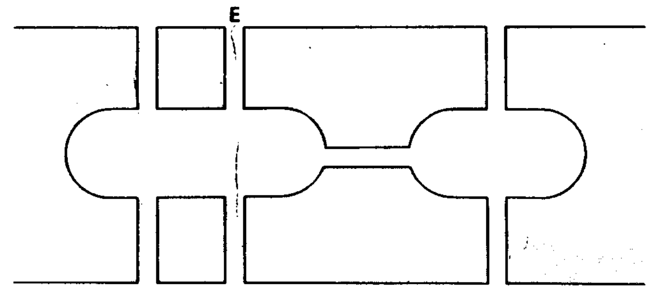

Figure 8-3 The seven bridges of Konigsberg.

uses rule D to transform this object into the object, "NOUN PHRASE VERB PHRASE NOUN PHRASE." It then uses rule A to transform this object into the final OBJ, "SENTENCE," and the parsing is complete. To perform this task, GPS generated 19 goals.

Seven Bridges of Konigsberg Task In a river in Konigsberg, Germany, were two islands connected with the mainland and with each other by seven bridges, as shown in Figure 8-3. The task is to start at point *E*, cross each bridge *exactly* once, and end at point *E*. In 1736, Euler, the famous mathematician, proved that this task is impossible. Nevertheless, this task was given to GPS. After generating 71 goals, GPS stopped because its memory was full. In its attempts it had found two ways of crossing six bridges. GPS should have ways of viewing a task as a whole instead of always just trying to apply operators within the task. After some failures on the task it should consider the possibility and perhaps even prove that the task is impossible.

Water Jug Task Given a 5-gal jug and an 8-gal jug, how can you get *exactly* 2 gal in the 5-gal jug? Since there is a sink nearby, a jug can be filled from the tap or emptied down the drain. Water can be poured from one jug to another. No measuring devices other than the jugs themselves are available. To perform this task, GPS generated 24 goals.

Letter Series Completion Task The task is to add the next two letters to the letter series BCBDBE . . . . To perform this task, GPS generated 27 goals. It finds a simple description of the given series and then adds the required two letters. This task involved inductive reasoning.

8.6 FINDINGS AND CONCLUSIONS

Tasks On the positive side, GPS can perform a wide variety of tasks. It can work on tasks involving inductive reasoning (for example, the letter series completion task) and deductive reasoning (for example, the predicate calculus task). GPS is a truly unified and multipurpose program and not merely a conglomeration of programs, each specialized to perform one task. One piece of evidence for this assertion is that each method is used for an average of half the tasks. It may be the forerunner of a program that will, in some significant sense, perform the task of doing its own learning. Newell, Shaw, and Simon (1960) present a very detailed hand simulation of how GPS might learn, although they acknowledge that some difficulties remain. We have already stressed the great value in getting a program to be general enough to do its own learning. On the negative side, GPS performs tasks very slowly. It should have ways of viewing a task as a whole instead of always just trying to apply operators within the task.

Problem-solving Techniques Means-ends analysis works very well on some tasks but not on others, depending on the suitability of the differences and difference ordering given in the task specification. The general techniques used by GPS resemble those used by humans. GPS needed to use only four types of goals, an amazingly small number. It was found that the problem-solving techniques strongly interact with the way tasks are represented. For the problem-solving techniques of GPS, certain properties of a good representation of a task were found. For example, it is generally better to represent operators as move operators (with variables, pretests, moves, and posttests) rather than as form operators. The way in which GPS selects the next goal to be tried will need to be improved, especially when the goal trees become large. GPS usually makes a local selection, but sometimes makes the somewhat arbitrary selection of the top or some other goal. The author believes that a goal-selection scheme analogous to that used in MULTIPLE would help GPS.

EXERCISES

1. List the principal similarities and differences between MULTIPLE and GPS.

2. Suppose that the difference ordering in a task environment specification says that the difference $D3$ is more difficult than is $D2$, which is more difficult than is $D1$.

a. Suppose that the problem-solving executive procedure of GPS selects goal 3 (see Figure 8-4), generates the subgoal G, and soon will go to step D. When the procedure arrives at step D, which is the selected goal, if G is to

(1) Reduce $D1$ on INITIAL OBJ.
(2) Reduce $D3$ on INITIAL OBJ.

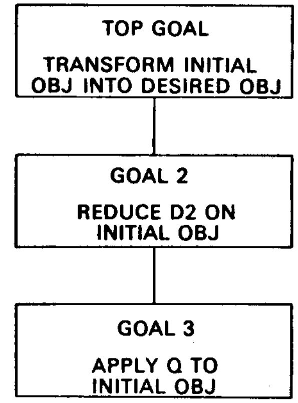

Figure 8-4

Explain your answers.

b. Suppose that the problem-solving executive procedure of GPS selects goal 4 (Figure 8-5), generates the subgoal G, and soon will go to step D. When the procedure arrives at step D, which is the selected goal, if G is to

(1) Reduce $D1$ on 2.
(2) Reduce $D3$ on 2.

Explain your answers.

3. In the seven bridges of Konigsberg task, show how one can, after starting at E, cross six bridges exactly once and not cross the other bridge.

4. Perform the following tasks and tabulate your times against the corresponding times taken by GPS.

a. Integrations.
b. Missionaries and cannibals.
c. Tower of Hanoi.
d. First-order predicate calculus.
e. Father and sons.
f. Three coins.
g. Water jug.
h. Letter series completion.

5. Give a step-by-step explanation of how the problem-solving executive procedure generates and selects goals 5, 6, and 7 in Figure 8-1.

6. Give a step-by-step explanation of how the problem-solving executive procedure generates and selects goals 9 through 13 in Figure 8-1.

7. With the aid of a goal tree, give a step-by-step explanation of how GPS would use the problem-solving executive procedure to perform the task ob-

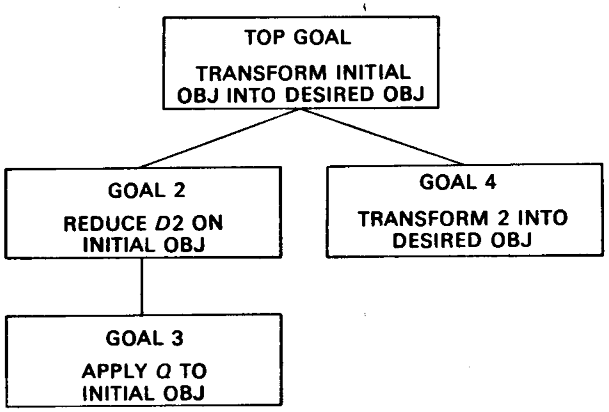

Figure 8-5

tained by having the following objects replace the objects in the specific monkey task in Table 8-1:

INITIAL OBJ =

- **a.** The monkey's place is place 1.
- **b.** The box's place is under the bananas.
- **c.** The contents of the monkey's hand are empty.

DESIRED OBJ = the monkey's place is under the bananas.

8. With the aid of a goal tree, give a step-by-step explanation of how GPS would use the problem-solving executive procedure to perform the task obtained by having the following objects replace the objects in the specific monkey task in Table 8-1.

INITIAL OBJ =

- **a.** The monkey's place is under the bananas.
- **b.** The box's place is under the bananas.
- **c.** The contents of the monkey's hand are empty.

DESIRED OBJ =

- **a.** The monkey's place is place 2.
- **b.** The box's place is place 1.

9. Give an example of a specific task which, when taken together with the monkey task environment in Table 8-1, would constitute a task that GPS would perform without generating any goal other than the top goal.

10. Are there any specific monkey tasks which would require GPS to generate more than 13 goals? Give a brief proof of your answer.

11. Let CLIMB DOWN be the MOVE operator that is the inverse of the operator CLIMB in the monkey task environment in Table 8-1; that is, CLIMB DOWN is the operator in which the monkey starts on the box and climbs down from the box.

a. Complete the following definition: CLIMB DOWN =

(1) Pretest.

(2) Move.

b. Give the row for CLIMB DOWN that should be added to the possible table of connections in Table 8-3 in order to make it easiest for GPS to handle CLIMB DOWN.

12. Suppose that the difference ordering in the monkey task environment in Table 8-1 were reversed; that is, the difference $D1$ is more difficult than is $D2$, which is more difficult than is $D3$. Which would be the first goal to be different from the corresponding goal in Figure 8-1? Explain your answer.

BIBLIOGRAPHY

Ernst, George W., and Allen Newell: "Generality and GPS," Center for the Study of Information Processing, Carnegie-Mellon University, Pittsburgh, Pa., January, 1967.

———, and ———: Some Issues of Representation in a General Problem Solver, *Proc. AFIPS Annu. Spring Joint Computer Conf.*, 1967, pp. 583–600.

———, and ———: "GPS: A Case Study in Generality and Problem Solving," *ACM Monograph*, Academic Press, Inc., New York, 1969.

McCarthy, John: "Situations, Actions, and Causal Laws," *Stanford Artificial Intelligence Project Memo* 2, Palo Alto, Calif., July, 1963.

Newell, Allen, J. C. Shaw, and Herbert Simon: A Variety of Intelligent Learning in a General Problem-solver, in M. C. Yovits and S. Cameron (eds.), "Self-Organizing Systems," Pergamon Press, New York, 1960, pp. 153–189.

Quinlan, J. R.: A Task-independent Experience Gathering Scheme for a Problem Solver, *Proc. Intern. Joint Conf. Artificial Intelligence*, 1969, pp. 193–198.