Computer Simulation of Human Thinking

Allen Newell; Herbert A. Simon

Science, New Series, Vol. 134, No. 3495 (Dec. 22, 1961), 2011-2017.

22 December 1961, Volume 134, Number 3495 **SCIENCE**

CURRENT PROBLEMS IN RESEARCH

Computer Simulation of Human Thinking

A theory of problem solving expressed as a computer program permits simulation of thinking processes.

Allen Newell and Herbert A. Simon

The path of scientific investigation in any field of knowledge records a response to two opposing pulls. On the one side, a powerful attraction is exerted by "good problems"—questions whose answers would represent fundamental advances in theory or would provide the basis for important applications. On the other side, strong pulls are exerted by "good techniques"—tools of observation and analysis that have proved to be incisive and reliable. The fortunate periods in a science are those in which these two pulls do not paralyze inquiry by their opposition but cooperate to draw research into fruitful channels.

When this happy condition is not substantially satisfied, science is threatened by schism. Some investigators will insist on working on important problems with methods that are insufficiently powerful and that lack rigor; others will insist on tackling problems that are easily handled with the available tools, however unimportant those problems may be.

Stress arising from the mismatch of ends and means is seldom completely absent from any science; examples could be provided from contemporary biology, meteorology, or mathematics. But it has been blatantly apparent in the science of psychology. This is true even if we leave out of account the tremendously important practical problems that are posed for the field by its potential applications in the clinic, in education, and in many areas of social policy. In basic research the disparity has been strikingly visible. We can fairly classify most psychological research, and even most research psychologists, by their orientation on this issue. "Gestaltism" is one of the labels applied to question-oriented psychology; "behaviorism" is the label most commonly applied to method-oriented psychology. It is no accident that research on human thinking, problem solving, personality, verbal behavior, and social phenomena has tended to attract psychologists closest to the "Gestalt" end of the continuum, while research on animal behavior, physiological psychology, rote memory, and simple motor skills has been primarily the domain of behaviorists.

It is commonly agreed that the dividing lines between the two points of view have become less clear since World War II. Several reasons might be given for this trend, but a full explanation would include the impact of new ideas drawn from cybernetics and the rapidly developing communications sciences. Complex electronic devices using feedback mechanisms to secure adaptive behavior have clarified concepts such as "goal seeking" and "learning" and have showed how these concepts could be made operational. This clarification has encouraged problem-oriented psychologists to give more precise operational meaning to terms that had been vague, and has encouraged technique-oriented psychologists to tackle problems that earlier appeared too complex for their tools.

The developments now taking place in psychology involve much more, however, than just a borrowing of new terms and new metaphors from other sciences. They involve the use of the digital computer as a tool both for constructing theories and for testing them. Enough has already been learned about this tool and its potentialities to indicate that many of the "good problems" of psychology are now within reach of the "good techniques."

We should like to discuss here one of several important applications of the computer to psychological research—its use as a device for simulating the processes of human thinking. We shall not attempt a review of computer-based research in this one sphere of application but shall present instead a specific example drawn from our own work.

The Behavioral Phenomena

Let us begin with a sample of the phenomena we wish to explain. We seat a subject in the laboratory (a college sophomore, member of a ubiquitous species in psychological research). We present him with a problem, which we tell him is a problem in "recoding" symbolic expressions. We present a certain expression:

$$R\cdot(\sim P \supset Q) \qquad (1)$$

and ask him to obtain from it a second expression:

$$(Q \vee P)\cdot R \qquad (2)$$

by applying to the first expression a succession of rules of transformation drawn from a list which we also put before him.

Dr. Newell is Institute Professor of systems and communications sciences, and Dr. Simon is professor of administration, Graduate School of Industrial Administration, Carnegie Institute of Technology, Pittsburgh, Pa. Both authors are consultants to the Rand Corporation, Santa Monica, Calif.

Readers familiar with symbolic logic will recognize the expressions and the rules, but the subjects were unacquainted with formal logic. The subjects read the first expression, for example as, "(r) dot (tilde-p horseshoe q)." They made no use of the meanings of the expressions in their usual interpretation but simply manipulated them as organized collections of symbols. If the reader wishes to follow the analysis in detail, he should adopt the same point of view.

We asked the subject to announce aloud each rule that he wished to apply and the expression that would result from its application. The experimenter then wrote the new expression on a blackboard. We also asked the subject to talk aloud about what he was doing—"what he was thinking about." We recorded the entire session on tape.

Here is the protocol of a subject working on the problem stated above (subject No. 9, problem $\alpha 1$).

Subject: "I'm looking at the idea of reversing these two things now."

Experimenter: "Thinking about reversing what?"

Subject: "The R's . . . then I'd have a similar group at the beginning, but that seems to be . . . I could easily leave something like that 'til the end, except then I'll. . . ."

Experimenter: "Applying what rule?"

Subject: "Applying, . . . for instance, 2. That would require a sign change."

Experimenter: "Try to keep talking, if you can."

Subject: "Well . . . then I look down at rule 3 and that doesn't look any too practical. Now 4 looks interesting. It's got three parts similar to that . . . and there are dots, so the connective . . . seems to work easily enough, but there's no switching of order. I need that P and Q changed, so . . . I've got a horseshoe there. That doesn't seem practical any place through here. I'm looking for a way, now, to get rid of that horseshoe. Ah . . . here it is, rule 6. So I'd apply rule 6 to the second part of what we have up there."

Experimenter: "Want to do that?"

Subject: "Yeah."

Experimenter: "OK, to line 1 you apply rule 6. Line 2 is **R•(PvQ)**."

Subject: "And now I'd use rule 1."

Experimenter: "Rule 1 on what part? You can use it with the entire expression or with the right part."

Subject: "I'd use it both places."

Experimenter: "Well, we'll do them one at a time . . . which do you want first?"

Subject: "Well, do it with P and Q."

Experimenter: "**R•(QvP)**. Now the entire expression?"

Subject: "Yeah."

Experimenter: "On line 3, rule 1 . . . you'd get **(QvP).R**."

Subject: "And . . . that's it."

Experimenter: "That's it all right; OK . . . that wasn't too hard."

The research problem, then, is to construct a theory of the processes causing the subject's behavior as he works on the problem, and to test the theory's explanation by comparing the behavior it predicts with the actual behavior of the subject. How can a computer help us to solve this problem?

Nonnumerical Computer Program as a Theory

An electronic digital computer is a device for adding, subtracting, multiplying, and dividing very rapidly. But it is now known to be much more than this. Speed in executing arithmetical operations is achieved by providing the computer with a program (usually stored in the computer memory) to govern the sequence of its operations, but designed to make that sequence conditional on the results of previous operations.

The instructions that make up the computer program, like the data on which it operates, are symbolic expressions. But while the data are normally interpreted as numbers, the instructions are interpreted as sequences of words—as sentences in the imperative mode. When the computer interprets the instruction "add *A* to *B*," it produces the same result that a person would produce if he were asked in English to "add the number labeled *A* to the number labeled *B*."

We see that a computer is not merely a number-manipulating device; it is a symbol-manipulating device, and the symbols it manipulates may represent numbers, letters, words, or even nonnumerical, nonverbal patterns. The computer has quite general capacities for reading symbols or patterns presented by appropriate input devices, storing symbols in memory, copying symbols from one memory location to another, erasing symbols, comparing symbols for identity, detecting specific differences between their patterns, and behaving in a manner conditional on the results of its processes.

Let us return now to our human subject in the laboratory. His behavior, which we wish to explain, consists of a sequence of symbol emissions. This statement does not depend on the "thinking aloud" technique used in these experiments. It would be equally true if the subject had responded to the task in writing, or by pushing buttons. In all cases, his behavior can be interpreted as a sequence of symbol productions—in the last case cited, a sequence of *L*'s and *R*'s, where *L* stands for "left button" and *R* stands for "right button."

We can postulate that the processes going on inside the subject's skin—involving sensory organs, neural tissue, and muscular movements controlled by the neural signals—are also symbol-manipulating processes; that is, patterns in various encodings can be detected, recorded, transmitted, stored, copied, and so on, by the mechanisms of this system. We shall not defend the postulate in detail—its true defense lies in its power to explain the behavior. Nor shall we speculate in detail about the precise neurophysiological mechanisms and processes that correspond to terms such as *symbol transmission, stored symbol, copying*, and the like.

Instead we shall adopt the tactic, highly successful in other sciences, of allowing explanation at several distinct levels, without for a moment denying that the mechanisms producing the behavior are ultimately reducible to physiological mechanisms and that these, in turn, are reducible to chemical and physical mechanisms. Just as we explain what goes on in the test tube by chemical equations and subsequently explain the chemical equations by means of the mechanisms of quantum physics, so we will attempt to explain what goes on in the course of thinking and problem solving by organization of symbol-manipulation processes, putting to one side the task of explaining these processes in neurophysiological terms.

This approach to building a theory of complex behavior is depicted in Fig. 1. We are concerned with the top half of the figure—with reducing the overt behavior to information processes. If this reduction can be carried out, then a second body of theory will be needed to explain information processes on the basis of neurological mechanisms. Tunneling through our

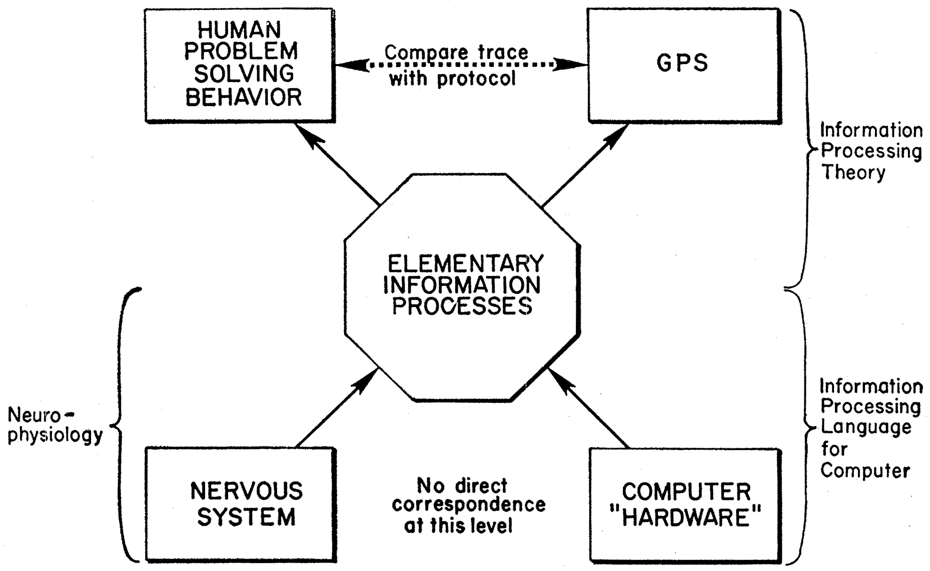

Fig. 1. Levels in an information processing theory of human thinking.

mountain of ignorance from both sides will prove simpler, we hope, than trying to penetrate the entire distance from one side only.

Using Fig. 1, we begin to see how a computer can help with the half of the tunneling operation that concerns us here. We postulate that the subject's behavior is governed by a program organized from a set of elementary information processes. We encode a set of subprograms (subroutines) for a digital computer, each of which executes a process corresponding to one of these postulated information processes. Then we undertake to write a program, compounded from these subroutines, that will cause the computer to behave in the same way that the subject behaves—to emit substantially the same stream of symbols—when both are given the same problem. If we succeed in devising a program that simulates the subject's behavior rather closely over a significant range of problem-solving situations, then we can regard the program as a theory of the behavior. How highly we will prize the theory depends, as with all theories, on its generality and its parsimony—on how wide a range of phenomena it explains and on how economical of expression it is.

It can be seen that this approach does not assume that the "hardware" of computers and brains are similar, beyond assuming that both are general-purpose symbol-manipulating devices and that the computer can be programmed to execute elementary information processes that are functionally quite like those executed by the brain. When we begin to theorize about the reduction of information processes to hardware, the brain and the computer (at least the computer used in this particular way) part company (see Fig. 1). The former calls for a physiologist, the latter for an electrical engineer or physicist.

From a formal standpoint, a computer program used as a theory has the same epistemological status as a set of differential equations or difference equations used as a theory:

1) Given a set of initial and boundary conditions, the differential equations predict the successive states of the system at subsequent points in time.

2) Given a set of initial and subsequent environmental inputs, the computer program predicts the successive states of the system (the subject's symbol emissions and the state of his memory) at subsequent points in time.

With this use of the computer we construct "equations" for nonnumerical symbol-manipulation phenomena without ever translating the phenomena into numerical form.

General Problem Solver

Our attempt to explain the problem-solving protocol, excerpted above, and others like it takes the form of a computer program that we call the General Problem Solver (GPS).

The program has means for representing internally (that is, in its memory) symbolic structures corresponding to the logic expressions, the rules for transforming expressions, and new expressions generated by applying the rules. The problem cited above is represented internally in the form of an expression that means "transform 1 into 2." We call the symbolic structures corresponding to the logic expressions *objects*; the structures corresponding to the problem statement and similar statements, *goals*. The program attains goals by applying *operators* to objects, thus transforming them into new objects.

The program has processes for applying operators to objects. It also has processes for comparing pairs of objects; these processes produce (internally) symbols that designate the differences between the objects compared. It has processes for generating new goals from given objects, operators, and differences.

The processes of GPS are organized around three types of goals and a small number of methods for attaining goals of these types (see Fig. 2).

1) *Transformation goals.* These are of the form already illustrated: Transform object a into object b.

Method 1. Compare a with b to find a difference, d, between them; if there is no difference, the problem is solved. Construct the goal of reducing difference d between a and b. If successful, the result will be a transformation of a into a new object, c. Now construct the new goal of transforming c into b. Attaining this goal will solve the original problem.

Method 1′. There is another method, the planning method, for attaining transformation goals. We do not have space to describe it in detail here. Briefly, it involves replacing the objects with corresponding abstracted objects, say, a'' and b'', then transforming a'' into b'' by means of the other methods and using the resulting sequence of operations as a *plan* for transforming a into b.

2) *Operator application goals.* These are of the form: Apply operator q to object a.

Method 2. Determine whether a meets the conditions for application of q. If so, apply q; if not, determine a difference between a and an object to which q is applicable. Construct the goal of reducing this difference. If successful, a new object a' will be produced, which is a modification of a. Now try to apply q to a'.

3) *Difference reduction goals.* As we have just seen, these are of the form: Reduce difference *d* between objects *a* and *b*.

Method 3. Find an operator, *q*, that is *relevant* to the difference in question (the meaning of relevance will be explained in a moment). Construct the goal of applying *q* to *a*. If successful, the result will be a transformation of *a* into a new object, *c*, which will not differ as much from *b*.

Thus, the General Problem Solver is a computer program comprised of rather general processes for reasoning about ends (goals) and means (operators). It is general in the sense that the program itself makes no reference to the precise nature of the objects, differences, and operators with which it is dealing. Hence, its problem-solving capacities can be transferred from one kind of task to another if it is provided with information about the kinds of objects, differences, and operators that characterize and describe the particular task environment it is to handle. Thus, to solve logic problems, it must be provided with a format for representing logic expressions, tests for the differences that must be recognized between pairs of expressions, and a list of the allowable operators. The rules of the game it is to play must be described to it.

At present, the General Problem Solver is also provided with a "table of connections" that lists the operators that are potentially useful (relevant) for reducing each of the recognizable types of differences. We have indicated in another place how this program could use its own problem-solving processes to construct the table of differences, and how it might even evolve a suitable set of differences if these were not provided to it in a new task environment (*1*).

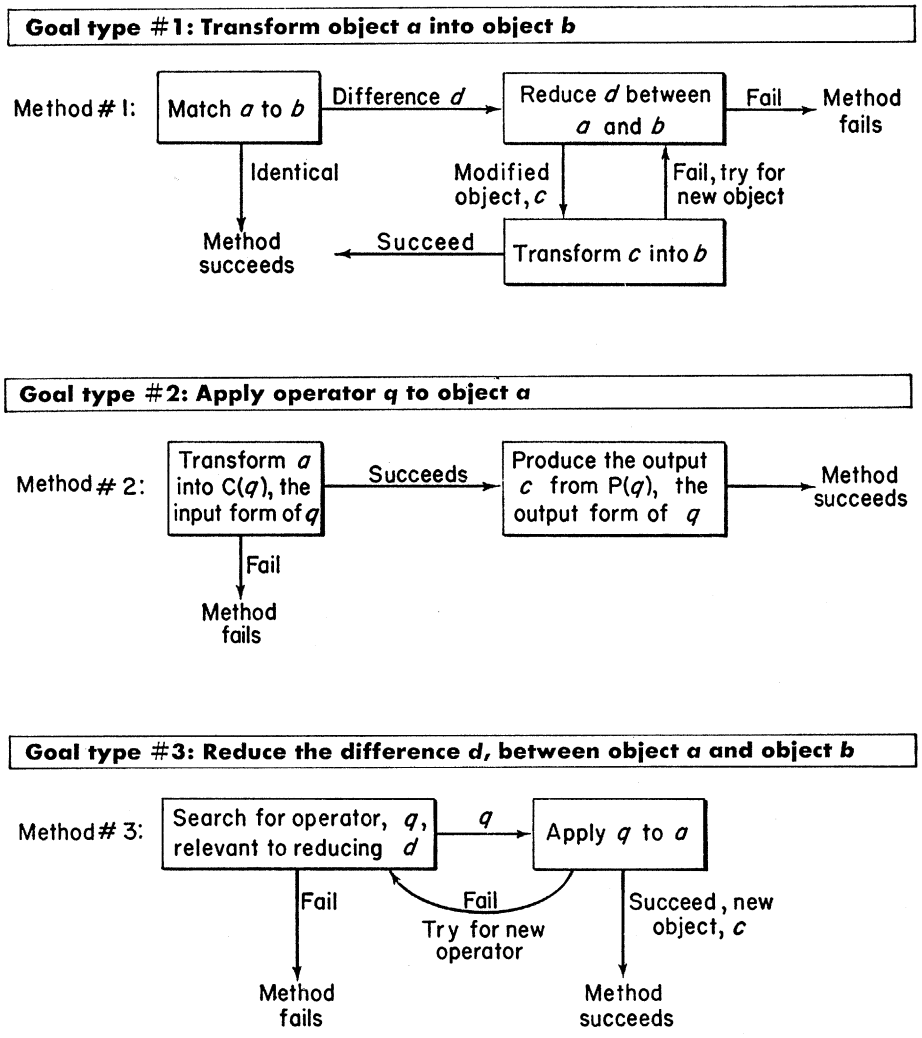

Fig. 2. Methods for means-ends analysis.

Testing the Theory

How adequate the program is as an information-processing theory of human problem solving can be asked at several levels of specificity. At the grossest level, we may ask whether the program does, in fact, solve problems of some of the sorts that humans solve. This it demonstrably does. Hence we may say that its program constitutes a system of mechanisms, constructed from elementary information processes, that is sufficient for solving some problems. It provides an unequivocal demonstration that a mechanism can solve problems by functional reasoning.

The general kinds of means-end analysis that the General Problem Solver uses are also the methods that turn up in the subjects' protocols. We have examined in fair detail some 20 protocols of subjects solving logic problems (*2*). Virtually all the behavior in these protocols falls within the general framework of means-end analysis. The three goal types we have described account for about three-fourths of the subjects' goals, and the additional goal types that appear in the protocols are closely related to those we have described. The three methods we have outlined represent the vast majority of the methods applied to these problems by the subjects. In addition, the planning method, mentioned above, appears in several different forms in the protocols.

Protocols of human problem-solving behavior in a range of tasks—playing chess, solving puzzles, writing computer programs—contain many sequences of behavior that are also quite similar to the means-end analysis of the General Problem Solver. We may cite, for example, the following excerpt from the thinking-aloud protocol of a chess player: "Again I notice that one of his pieces is not defended, the rook, and there must be ways of taking advantage of this. Suppose now, if I push the pawn up at bishop 4, if the bishop retreats I have a queen check and I can pick up the rook. If the bishop takes the pawn, then I can win a piece by simply again bringing either the queen down with check, or knight takes bishop."

We cannot, of course, on the basis of this kind of evidence, conclude that GPS provides an adequate explanation for all these kinds of problem-solving

Computer Trace	Protocol of Subject
L0 (QvP).R L1 R.(~P⊃Q)	(L0 is expression to be obtained) (L1 is expression given at start) (Goal 1 is set by the experimenter)
GOAL 1 TRANSFORM L1 INTO L0 GOAL 2 CHANGE POSITION IN L1 GOAL 3 APPLY R1 TO L1 [A.B → B.A] PRODUCES L2 (~P⊃Q).R GOAL 4 TRANSFORM L2 INTO L0 GOAL 5 CHANGE POSITION IN LEFT L2 GOAL 6 APPLY R2 TO LEFT L2 [A⊃B → ~B⊃~A] PRODUCES L3 (~Q⊃P).R	I'm looking at the idea of reversing these two things now. Thinking about reversing what? The R's . . . then I'd have a similar group at the beginning but that seems to be . . . I could easily leave something like that 'til the end, except then I'll. . . . Applying what rule? Applying, . . . for instance, 2.
GOAL 7 TRANSFORM L3 INTO L0 GOAL 8 CHANGE SIGN LEFT L3 NONE FOUND	That would require a sign change.
GOAL 5 GOAL 9 APPLY R3 TO L2 [A.A → A] REJECT, NOT DESIRABLE GOAL 10 APPLY R4 TO L2 [(A.B).C → A.(B.C)] REJECT, NOT DESIRABLE GOAL 11 APPLY R5 TO L2 [A.B → ~(~Av~B)] REJECT, NOT DESIRABLE GOAL 12 APPLY R7 TO L2 [A.(BvC) → (A.B)vA.C)] REJECT, NOT DESIRABLE GOAL 13 APPLY R8 TO L2 [A.B → A] REJECT, NOT DESIRABLE	Try to keep talking, if you can. Well . . . then I look down at rule 3 and that doesn't look any too practical. Now 4 looks interesting. It's got three parts similar to that . . . and there are dots, so the connective . . . seems to work easily enough, but there's no switching of order.
GOAL 5 GOAL 14 APPLY R1 TO LEFT L2 [AvB → BvA] GOAL 15 CHANGE CONNECTIVE TO V IN LEFT L2 GOAL 16 APPLY R6 TO LEFT L2 [A⊃B → ~AvB] PRODUCES L4 (PvQ).R GOAL 17 APPLY R1 TO LEFT L4 [AvB → BvA] PRODUCES L5 (QvP).R	I need that P and a Q changed, so . . . I've got a horseshoe there. That doesn't seem practical any place through here. I'm looking for a way, now, to get rid of that horseshoe. Ah . . . here it is, rule 6.
GOAL 18 TRANSFORM L5 INTO L0 IDENTICAL	So I'd apply rule 6 to the second part of what we have up there. Want to do that? Yeah. OK, to line 1 you apply rule 6. Line 2 is R.(PvQ). And now I'd use rule 1. Rule 1 on what part? You can use it with the entire expression or with the right part. I'd use it both places. Well, we'll do them one at a time . . . which do you want first? Well, do it with P and Q. R.(QvP). Now the entire expression? Yeah. On line 3, rule 1 . . . you'd get (QvP).R. And . . . that's it. That's it all right, OK . . . that wasn't too hard.

Fig. 3. Comparison of computer trace (left) with protocol of a subject (right). The rules are shown in square brackets in the computer trace. In the protocol, the experimenter's words are underlined.

behavior. Many other mechanisms may be involved besides those that are incorporated in it. Only when a program simulates the entire sequence of behavior—for example, makes the same chess analysis as the human player—do we have any assurance that we have postulated a set of processes that is sufficient to produce the behavior in question.

These tests are still very general and do not take into account differences among the programs of different subjects. Obviously, not all subjects solve the problems in exactly the same way. The evidence presented thus far suggests that programs of most subjects share the general qualitative features of GPS, but there are variations in detail. We can subject the theory to further tests by seeing what modifications in GPS, if any, will enable us to predict, in detail, the symbolic behavior of a particular subject during some interval of his problem-solving activity.

In Fig. 3 we compare, in parallel columns, the protocol segment introduced earlier with the output of a particular version of GPS set to the task of solving the same problem. The right-hand half of Fig. 3 is the human subject's protocol; the left-hand half is the trace of the program. The language of the subject is much less stylized than the language of the computer. To fit the theory, we must, for example, interpret a sentence such as, "I'm looking at the idea of reversing these two things now," as equivalent to "Construct the difference-reduction goal of eliminating the difference in position of corresponding subparts in objects L1 and L2." To make such a translation is, in practice, not too difficult, and having made it, we can determine in great detail the similarities and differences between the programs of the subject and the computer, respectively.

Let us consider some of the differences visible in the example at hand—differences that represent inadequacies of GPS in its present form as an accurate theory of the subject's behavior. Observe that the subject solves the entire problem in his head and then asks the experimenter to write the actual transformations on the blackboard. The GPS program, in the version shown here, makes no provision for such a distinction between the internal and external worlds; hence, the trace corresponds only to the subject's covert (but verbalized) problem solving. For example, GPS and the subject both discover in the same sequence the correct rules for transforming the problem expression, but the subject "publicly" applied these rules in the reverse order.

Another difference, characteristic of these data, and of such data in general, is that a number of things appear in the trace that have no correspondents in the human protocol—most prominently, the references here in the trace to rules 5, 7, and 8. We cannot tell whether these omissions indicate an error in the theory, or whether the subject noticed the rules in question but failed to mention them aloud.

In contrast to these differences, there is some striking correspondence in detail between the computer trace and the subject's protocol. First, in noticing differences between pairs of expressions, both GPS and the subject pay most attention to differences in the positions of symbols, next most attention to the presence or absence of "$\sim$" signs, and least attention to differences in connectives. This shows up, for example, in the refusal of both to apply rule 2, after mentioning it, to reorder the expression, because applying the rule involves changing a sign. Second, of the several possible paths to solution of the problem, both program and subject chose an application of rule 6 and two applications of rule 1.

These samples of success and failure will give the reader some indication of the kind of detailed comparison that can be made between theoretical predictions of computer models of this kind and actual human behavior. Much remains to be learned about how to make such comparisons and how to test their "goodness of fit." The fragmentary evidence we have obtained to date encourages us to think that GPS provides a rather good approximation to an information-processing theory of certain kinds of thinking and problem-solving behavior. The processes of thinking can no longer be regarded as completely mysterious.

Conclusion

A digital computer is a general-purpose symbol-manipulating device. If appropriate programs are written for it, it can be made to produce symbolic output that can be compared with the stream of verbalizations of a human being who is thinking aloud while solving problems. The General Problem Solver is a computer program that is capable of simulating, in first approximation, human behavior in a narrow but significant problem domain.

The General Problem Solver is not the only existing program of this type. There is a program, the predecessor of GPS, that also discovers proofs for theorems, but only in symbolic logic (*3*). There are programs for proving theorems in geometry (*4*), for designing electric motors, generators, and transformers (*5*), for writing music (*6*), and for playing chess (*7*). There are programs that "learn"—that is, that modify themselves in various respects on the basis of experience (*8*). We omit from the list those programs that make primary use of the computer's arithmetical capabilities and that are not particularly like human processes, even in their general organization. All of the programs listed, other than GPS, are limited to a single task environment, and none of them seeks to simulate the corresponding human processes in detail. Nevertheless, their underlying structures are all extremely similar, involving selective search for possible solutions based on rules of thumb, or heuristics. This communality provides further evidence of the basic correctness of the approach illustrated by the General Problem Solver in the construction of a theory of human thinking.

In our discussion we have limited ourselves to problem-solving programs. Several recent investigations undertake to simulate other kinds of human cognitive activity that have been studied in the psychological laboratory. Feldman (*9*) has written a simulation program for partial reinforcement experiments; Feigenbaum (*10*) and Feigenbaum and Simon have written a program that simulates subjects' behavior in rote memory experiments; Hunt and Hovland (*11*) and Laughery and Gregg have written programs that simulate concept-forming behavior. In addition, there are a substantial number of programs for pattern-recognition tasks. There are now a score or more of research psychologists who are constructing and testing information-processing theories of cognitive processes, formulating their theories as computer programs, and testing them by comparing the computer simulations with the protocols of human subjects.

Psychology has discovered an important new tool whose power appears to be commensurate with the complexity of the phenomena the science seeks to explain. As our skills in using this new tool develop, we may expect that the paralyzing conflict between the good problems in psychology and the good techniques will be greatly lessened (*12*).

References and Notes

1. A. Newell, J. C. Shaw, H. A. Simon, in *Self-Organizing Systems*, M. C. Yovits and S. Cameron, Eds. (Pergamon, New York, 1960), pp. 153–189.
2. For discussion of the comparison of the trace of slightly different versions of GPS with protocols, see A. Newell and H. A. Simon, in *Current Trends in Psychological Theory* (Univ. of Pittsburgh Press, Pittsburgh, 1961), pp. 152–179, and ———, *Proc. Conference on Learning Automata, Karlsruhe, Apr. 1961* (Oldenbourg, Munich, 1961).
3. A. Newel and H. A. Simon, *IRE Trans. on Inform. Theory* **IT-2**, No. 3, 61 (1956).
4. H. Gelernter, J. R. Hansen, D. W. Loveland, *Proc. Western Joint Computer Conf. 1960* (1960), pp. 143–150.
5. G. L. Goodwin, *Power* **102** (Apr. 1958).
6. L. A. Hiller and L. M. Isaacson, *Experimental Music* (McGraw-Hill, New York, 1959).
7. A. Newell, J. C. Shaw, H. A. Simon, *IBM J. Research and Develop.* **2**, 320 (1958).
8. A. L Samuel, *ibid.* **3**, 210 (1959); A. Newell, J. C. Shaw, H. A. Simon, *Proc. Western Joint Computer Conf. 1957* (1957), pp. 218–230.
9. J. Feldman, *Proc. Western Joint Computer Conf. 1961* (1961), pp. 133–144.
10. E. A. Feigenbaum, *Proc. Western Joint Computer Conf. 1961* (1961), pp. 121–132.
11. C. I. Hovland and E. B. Hunt, *ibid.* (1961).
12. We acknowledge our debt to J. C. Shaw, our co-worker in the research described here.

Medical Research: Past Support, Future Directions

Aims of the National Institutes of Health are surveyed as its annual budget passes the half-billion mark.

Dale R. Lindsay and Ernest M. Allen

The health status of the nation is a complex matter, involving many factors. Cancer, tuberculosis, heart disease, pneumonia and influenza, arthritis, blindness, deafness, mental illnesses, diabetes—these are only a few of the hundreds of diseases and disabilities that have long afflicted mankind and that still persist as greater or lesser health problems in this and other countries.

New diseases have appeared in the world from time to time, and the industrial age has brought with it environmental health problems not dreamed of by earlier generations. Left to themselves these influences, together with the greater opportunities for the spread of contagion in a crowded urban society, would have brought our national health level to a new low, beneath that of the preponderantly rural society of a century ago. Yet, as we are all aware, such have been the advances in the broad attack upon these influences that there has been a steady improvement in the health status of the nation.

The picture has not been one of uniform improvement on all fronts, as may be seen in the death rates for our two major killers, heart disease and cancer (Table 1). We find encouragement, on the other hand, in figures such as those in Table 2, for three other disease categories. Still other diseases have declined to so low a level of importance in the total health picture that they must be looked for only among the fine details. Typhoid fever, malaria, and smallpox, once scourges, have been tamed. The hookworm problem is steadily diminishing in importance in areas where hookworm was once so prevalent. Pellagra is almost a thing of the past.

Health Parameters

We may feel the need of an over-all measurement that expresses or reflects the nation's present health status and permits us to evaluate past and future change. One that is informative is the age-adjusted death rate in our population for deaths from all causes. It stands now at only 44 percent of the death rate at the beginning of the century and has gone down appreciably even in the past several years (Table 3).

Another over-all measurement, a different health parameter of the population, is the average life span, known technically as the "life expectancy at birth." It stands at the highest figure in our history, is among the highest in the world, and has risen noticeably in even so short a period as the past 8 or 9 years (Table 4).

Further information, of a different sort, dealing with the prevalence of *all* illnesses, not just those that have a fatal outcome, might be had from figures on the average number of days per person per year lost from work or other normal activity because of illness—the average days of "incapacity." No information from which to compute this additional parameter is available for the past decade, but we may anticipate that such data for coming years will be available in the future (*1*).

The death rate, average life span, and average days of incapacity are not, of course, the only informative parameters of the health of a population that one might desire. The summary data that are available and that are given here, however, do reflect the generally favorable trend observed in the past half century and more. They also bring to sharp focus a challenge: It is necessary that the trend, where favorable, be continued or even accelerated, and that every effort be made to reverse the present trend in the incidence and outcome of diseases, such as heart disease and cancer, which have not yet responded favorably.

To accept such a challenge, it is necessary to understand the factors responsible for the improvement in health

---

Dr. Lindsay is chief of the Division of Research Grants and Dr. Allen is associate director for research grants, National Institutes of Health, Bethesda, Md.