I will, in what follows, try to maintain the position that there is nothing wrong with viewing man as an information processor (or indeed as anything else) nor with attempting to understand him from that perspective, providing, however, that we never act as though any single perspective can comprehend the whole man. See-ing man as an information-processing system does not in itself de-humanize him, and may very well contribute to his humanity in that it may lead him to a deeper understanding of one specific aspect of his human nature. It could, for example, be enormously important for man's understanding his spirituality to know the limits of the explanatory power of an information-processing theory of man. In order for us to know those limits, the theory would, of course, have to be worked out in considerable detail.

Before we discuss what an information-processing theory of man might look like, I must say more about theories and especially about their relation to models. A theory is first of all a text, hence a concatenation of the symbols of some alphabet. But it is a symbolic construction in a deeper sense as well; the very terms that a theory employs are symbols which, to paraphrase Abraham Kaplan, grope for their denotation in the real world or else cease to be symbolic.[3] The words "grope for" are Kaplan's, and are a happy choice—for to say that symbols "find" their denotation in the real world would deny, or at least obscure, the fact that the symbolic terms of a theory can never be finally grounded in reality.

Definitions that define words in terms of other words leave those other words to be defined. In science generally, symbols are often defined in terms of operations. In physics, for example, mass is, informally speaking, that property of an object which determines its motion during collision with other objects. (If two objects moving at identical velocities come to rest when brought into head-on collision, it is said that they have the same mass.) This definition of mass permits us to design experiments involving certain operations whose outcomes "measure" the mass of objects. Momentum is defined as the product of the mass of an object and its velocity (mv), acceleration as the rate of change of velocity with time ($a = dv/dt$), and finally force as the product of mass and acceleration ($f = ma$). In a way it is wrong to say that force is "defined" by the equation $f = ma$. A more suitable definition given in some physics texts is that force is any influence capable of producing a change in the motion of a body.[4] The difference between the two senses of "definition" alluded to here illustrates that so-called operational definitions of a theory's terms provide a basis for the design of experiments and the discovery of general laws, but that these laws may then serve as implicit definitions of the terms occurring in them. These and still other problematic aspects of definition imply that all theoretic terms, hence all theories, must always be characterized by a certain openness. No term of a theory can ever be fully and finally understood. Indeed, to once more paraphrase Kaplan, it may not be possible to fix the content of a single concept or term in a sufficiently rich theory (about, say, human cognition) without assessing the truth of the whole theory.[5] This fact is of the greatest importance for any assessment of computer models of complex phenomena.

A theory is, of course, not merely any grammatically correct text that uses a set of terms somehow symbolically related to reality. It is a systematic aggregate of statements of laws. Its content, its very value as theory, lies at least as much in the structure of the interconnections that relate its laws to one another, as in the laws themselves. (Students sometimes prepare themselves for examinations in physics by memorizing lists of equations. They may well pass their examinations with the aid of such feats of memory, but it can hardly

be said that they know physics, that, in other words, they command a theory.) A theory, at least a good one, is thus not merely a kind of data bank in which one can "look up" what would happen under such and such conditions. It is rather more like a map (an analogy Kaplan also makes) of a partially explored territory. Its function is often heuristic, that is, to guide the explorer in further discovery. The way theories make a difference in the world is thus not that they answer questions, but that they guide and stimulate intelligent search. And (again) there is no single "correct" map of a territory. An aerial photograph of an area serves a different heuristic function, say, for a land-use planner, than does a demographic map of the same area. One use of a theory, then, is that it prepares the conceptual categories within which the theoretician and the practitioner will ask his questions and design his experiments.*

Ordinarily, of course, when we speak of putting a theory to work, we mean drawing some consequences from it. And by that, in turn, we mean postulating some set of circumstances that involves some terms of the theory, and then asking what the theory says those particular circumstances imply for others of the theory's terms. We may describe the state of the economy of a specific country to an economist, for example, by giving him a set of the sorts of economic indices his particular economic theory accommodates. He may ask us some questions which, he would say, emerge directly from his theory. Such questions, by the way, might give us more insight into whether he is, say, a Marxist or a Keynesian economist than any answers he might ultimately give us, for they would reveal the structure of his theory, the network of connections between the eco-

* It must not be thought that this heuristic function of theory is manifest only in science. To name but one of the possible examples outside the sciences, Steven Marcus, the American literary critic, used theories of literary criticism freshly honed on the stone of psychoanalytic theory to do an essentially anthropological study of that "foreign, distinct, and exotic" subculture that was the sexual subculture of Victorian England. See his *The Other Victorians* (New York: Basic Books, 1966). More recently he wrote in the preface of his *Engels, Manchester, and the Working Class* (New York: Random House, 1974), "The present work may be regarded as part of a continuing experiment . . . to ascertain how far literary criticism can help us to understand history and society; to see how far the intellectual discipline that begins with the work of close textual analysis can help us understand certain social, historical, or theoretical documents." In neither book was a theory of literary criticism "applied," as, for example, a chemical theory may be applied to the chemical analysis of a compound; instead, Marcus' theories were used heuristically, as travelers use maps to explore a strange territory.

nomic laws in which he believes. Finally, we expect to be told what his theory says, e.g., that the country will do well, or that there will be a depression. More technically speaking, we may say that to put a theory to work means to assign specific values, by no means always numerical, to some of its parameters (that is, to the entities its terms signify), and then to methodically determine what values the theory assigns to other of its parameters. Often, of course, we arrive at the specifications to which we wish to apply a theory by interrogating or measuring some aspect of the real world. The input, so to speak, to a political theory may, for example, have been derived from public-opinion polls. At other times our specifications may be entirely hypothetical, as, for example, when we ask of physics what effect a long journey near the speed of light would have on the timekeeping property of a clock. In any case, we identify certain terms of the theory with what we understand them to denote, associate specifications with them, and, in effect, ask the theory to figure out the consequences.

Of course, a theory cannot "figure out" anything. It is, after all, merely a text. But we can very often build a model on the basis of a theory. And there are models which can, in an entirely nontrivial sense, figure things out. Here I am not referring to static scale models, like those made by architects to show clients what their finished buildings will look like. Nor do I mean even the scale models of wings that aerodynamicists subject to tests in wind tunnels; these are again static. However, the system consisting of both such a wing and the wind tunnel in which it is flown is a model of the kind I have in mind. Its crucial property is that it is itself capable of behaving in a way similar to the behaving system it represents, that is, a real airfoil moving in a real airmass. The behavior of the wing in the wind tunnel is presumably determined by the same aerodynamic laws as govern the behavior of the wings of real airplanes in flight. The aerodynamicist therefore hopes to learn something about a full-scale wing by studying its reduced-scale model.

The connection between a model and a theory is that a model *satisfies* a theory; that is, a model obeys those laws of behavior that a corresponding theory explicitly states or which may be derived from it. We may say, given a theory of a system B, that A is a

model of *B* if that theory of *B* is a theory of *A* as well. We accept the condition also mentioned by Kaplan that there must be no causal connection between the model and the thing modelled; for if a model is to be used as an explanatory tool, then we must always be sure that any lessons we learn about a modeled entity by studying its model would still be valid if the model were removed.

People do, of course, derive consequences from theories without building explicit models like, say, scaled-down wings in wind tunnels. But that is not to say that they derive such consequences without building models at all. When a psychiatrist applies psychoanalytic theory to data supplied to him by his patient, he is, so to speak, exercising a mental model, perhaps a very intuitive one, of his patient, a model cast in psychoanalytic terms. To state it one way, the analyst finds the study of his mental model (*A*) of his patient (*B*) useful for understanding his patient (*B*). To state it another way, the analyst believes that psychoanalytic theory applies to his patient and therefore constructs a model of him in psychoanalytic terms, a model to which, of course, psychoanalytic theory also applies. He then transforms (translates is perhaps a better word) inferences derived from working with the model into inferences about the patient. (It has to be added, lest there be a misunderstanding, that however much the practicing psychoanalyst is committed to psychoanalytic theory and however much his attitudes are shaped by it, psychoanalytic therapy consists in only small part of direct or formal application of theory. Nevertheless, it is plausible that all of us make all our inferences about reality from mental models whose structures, and to a large extent whose contents as well, are strongly determined by our explicitly and implicitly held theories of the world.)

Computers make possible an entirely new relationship between theories and models. I have already said that theories are texts. Texts are written in a language. Computer languages are languages too, and theories may be written in them. Indeed, for the present purpose we need not restrict our attention to machine languages or even to the kinds of "higher-level" languages we have discussed. We may include all languages, specifically also natural languages, that computers may be able to interpret. The point is

precisely that computers do *interpret* texts given to them, in other words, that texts determine computers' behavior. Theories written in the form of computer programs are ordinary theories as seen from one point of view. A physicist may, for example, communicate his theory of the pendulum either as a set of mathematical equations or as a computer program. In either case he will have to identify the terms of his theory—his "variables," in technical jargon—with whatever they are to correspond to in reality. (He may say l is the length of the pendulum's string, p its period of oscillation, g the acceleration due to gravity, and so on.) But the computer program has the advantage not only that it may be understood by anyone suitably trained in its language, just as a mathematical formulation can be readily understood by a physicist, but that it may also be run on a computer. Were it to be run with suitable assignments of values to its terms, the computer would *simulate* an actual pendulum. And inferences could be drawn from that simulation, and could be directly translated into inferences applicable to real pendulums. A theory written in the form of a computer program is thus both a theory and, when placed on a computer and run, a model to which the theory applies. Newell and Simon say about their information-processing theory of human problemsolving, "the theory performs the tasks it explains."[6] Strictly speaking, a theory cannot "perform" anything. But a model can, and therein lies the sense of their statement. We shall, however, have to return to the troublesome question of what the performance of a task can and cannot explain.

In order to aid our intuition about what it means for a computer model to "behave," let us briefly examine an exceedingly simple model: We know from physics, and indeed it follows from the equation $f = ma$ that we mentioned earlier, that the distance d an object will fall in a time t is given by

$$d = at^2/2,$$

where a is the acceleration due to gravity. In most elementary physics texts, a is simply asserted to be the earth's gravitational constant, namely, 32 ft/sec^2, where the unit of distance is feet and that of time is seconds. The equation itself is a simple mathematical model of a

falling object. If we assume, for the sake of simplicity, that the acceleration a is indeed constant, namely, 32 ft/sec^2, we can compute how far an object will have fallen after, say, 4 seconds: $4 \times 4 = 16$ and $16 \times 32 = 512$ and $512 \div 2 = 256$. The answer, as schoolchildren would say, is therefore 256 feet.

Mathematicians long ago fell into the habit of writing the so-called variables that appear in their equations as single letters. Perhaps they did this to guard against writer's cramp or to save chalk. Whatever their reasons, their notation is somewhat less than maximally mnemonic. Because computer programs are often intended to be read and understood by people, as well as to be executed by computers, and since computers are, within limits, indifferent to the lengths of the symbol strings they manipulate, computer programmers often use whole words to denote the variables that appear in their programs. Other considerations make it inconvenient to use juxtaposition of variables, as in xy, to indicate multiplication. Instead the symbol "*" is used in many programming languages. Similarly, "**" is used to indicate exponentiation. Thus, where the mathematician writes t^2, the programmer writes $t**2$. The equation

$$d = at^2/2$$

when transformed into a program statement* may thus appear as

distance = (acceleration * time **2)/2.

Let us now complicate our example just a little. Suppose an object is to be dropped from a stationary platform, say, a helicopter

* A significant technical point must be made here. Although the "statement" shown here is a transliteration of the equation to which it corresponds, it is not itself an equation. In technical parlance, it is an "assignment statement." It assigns a value to the variable "distance." "Distance," in turn, is technically an "identifier," the name of a storage location in which is stored the value which has been assigned to the corresponding variable. In mathematics, a variable is an entity whose value is not known, but which has a definite value nonetheless, a value that can be discovered by solving the equation. In programs, a variable may have different values at different stages of the execution of the program. In ordinary mathematics, e.g., in high-school algebra, the "equation" "$x = x + 1$" is nonsense. The same string of symbols appearing as an expression in a program has meaning, namely, that 1 is to be added to the contents of the location denoted by "x" and those contents replaced by the resulting sum.

hovering at some altitude above the ground. The object's height above the ground after it has fallen for some time would then be given by

$$\text{height} = \text{altitude} - (\text{acceleration} * \text{time} ** 2)/2.$$

Finally, suppose that the helicopter is flying forward at some constant velocity while maintaining its altitude. If there were no aerodynamic effects on the object dropped from the helicopter, it would remain exactly below the helicopter during its entire journey to the ground. The object's horizontal displacement from the point over which it was dropped would therefore be the same as the helicopter's horizontal displacement from that point, that is,

$$\text{displacement} = \text{velocity} * \text{time},$$

where by "velocity" we here, of course, mean the helicopter's velocity.

We now have, from one point of view, two equations, from another point of view, two program statements, from which we can compute the horizontal and vertical coordinates of an object dropped from a moving helicopter. We can combine them and imbed them in a small fragment of a computer program, as follows:

```
FOR time = 0 STEP .001 UNTIL height = 0 DO;
    height = altitude - (acceleration * time**2) / 2 ;
    displacement = velocity * time ;
    display (height, displacement) ;
END.
```

This is an example of a so-called *iteration statement.* It tells the computer to do a certain thing until some condition is achieved. In this case, it tells the computer to first set the variable "time" to zero, then to compute the height and displacement of what we would interpret to be the falling object, then to display the coordinates so computed—I shall say more about displaying in a moment—and, if the computed height is not zero, to add .001 to the variable "time"

and do the whole thing again, that is, to iterate the process. (This program contains an error which, for the sake of simplicity, I have let stand. As it is, it may run forever. To repair it, the expression "height = 0" should be replaced by "height < 0." The reason for this is left to the reader to discover.)

We have assumed here that the computer on which this program is to run has a built-in display apparatus and the corresponding display instruction. We may imagine the computer's display to be a cathode-ray tube like that of an ordinary television set. The display instruction delivers two numbers to this device, in this example, the values of height and displacement. The display causes a point of light to appear on its screen at the place whose coordinates are determined by these two numbers, i.e., so many inches up and so many inches to the right of some fixed point of origin.

If we now make some additional assumptions about for example, the persistence of the lighted dot on the screen and the overall timing of the whole affair, we can imagine that the moving dot we see will appear to us like a film of the object falling from the helicopter (see Figure 5.1). It is thus possible, even compelling, to think of the computer "behaving," and for us to interpret its behavior as modeling that of the falling object.

It would be very easy for us to complicate our example step by step, first, for example, by extending it to cover the trajectory of a missile fired from a gun and, with that as a base, to extend it to the flight of orbiting satellites. We would then have described at least

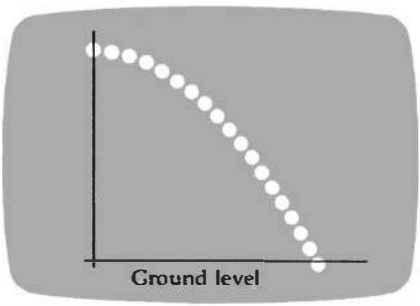

Figure 5.1.
Cathode simulation of the trajectory of an object dropped from a flying helicopter.

the most fundamental basis on which the orbital simulations we often see on television are developed. But that is not my purpose. Simple as our example is, we can learn pertinent lessons from it.

To actually use the model, an investigator would initialize it by assigning values to the parameters altitude and velocity, run it on an appropriate computer, and observe its behavior on the computer's display device. There would, however, be discrepancies between what the model, so to speak, says a falling object would do and the behavior of its real counterpart. The model, for example, makes the implicit assumption that there are no aerodynamic effects on the falling object. But we know that there would certainly be air resistance in the real situation. Indeed, if the object dropped were a parachute, its passenger's life would depend on air resistance slowing its fall. A model is always a simplification, a kind of idealization of what it is intended to model.

The aim of a model is, of course, precisely not to reproduce reality in all its complexity. It is rather to capture in a vivid, often formal, way what is essential to understanding some aspect of its structure or behavior. The word "essential" as used in the above sentence is enormously significant, not to say problematical. It implies, first of all, purpose. In our example, we seek to understand how the object falls, and not, say, how it reflects sunlight in its descent or how deep a hole it would dig on impact if dropped from such and such a height. Were we interested in the latter, we would have to concern ourselves with the object's weight, its terminal velocity, and so on. We select, for inclusion in our model, those features of reality that we consider to be essential to our purpose. In complex situations like, say, modeling the growth, decay, and possible regeneration of a city, the very act of choosing what is essential and what is not must be at least in part an act of judgment, often political and cultural judgment. And that act must then necessarily be based on the modeler's intuitive mental model. Testing a model may reveal that something essential was left out of it. But again, judgment must be exercised to decide what the something might be, and whether it is "essential" for the purpose the model is intended to serve. The ultimate criteria, being based on intentions and pur-

poses as they must be, are finally determined by the individual, that is, human, modeler.

The problem associated with the question of what is and what is not "essential" cuts the other way as well. A model is, after all, a different object from what it models. It therefore has properties not shared by its counterpart. The explorers we mentioned earlier may have built a functional model of the computer they found by using light-carrying fibers and light valves, whereas the real computer used wires and the kind of electronic gates we considered in Chapter III. They could then easily have come to believe that light is essential to the operation of computers. Their computer science might have included large elements of physical optics, and so on. It is indeed possible to build computers using light-carrying fibers, etc. Their logical diagrams, that is, their paper designs, would, up to a point, be indistinguishable from those of the corresponding electronic computers, because the former would have the same structure as the latter. What is essential about a computer is the organization of its components and not, again up to a point, precisely what those components are made of. Another example: there are people who believe it possible to build a computer model of the human brain on the neurological level. Such a model would, of course, be in principle describable in strictly mathematical terms. This might lead some people to believe that the language our nervous system uses must be the language of our mathematics. Such a belief would be an error of the kind we mean. John von Neumann, the great computer pioneer, touched briefly on this point himself:

> "When we talk mathematics, we may be discussing a *secondary* language, built on the *primary* language truly used by the central nervous system. Thus the outward forms of *our* mathematics are not absolutely relevant from the point of view of evaluating what the mathematical or logical language *truly* used by the central nervous system is."[7]

One function of a model is to test theories at their extreme limits. I have already mentioned that computers can generate films that model the behavior of a particle at extreme limits of relativistic

velocities. Our own simple model of falling objects could be used in its present form to simulate, hence to calculate, the fall of an object from a spaceship flying near the surface of the moon. All we would have to do is to initialize acceleration to the number appropriate for the gravity existing on the moon's surface (providing, of course, that the spaceship is not so high above the surface of the moon that the effect of the moon's gravitational field would have been significantly changed—another implicit assumption). For that simulation exercise we would not have to have any components in our model corresponding to air resistance or other aerodynamic effects: the moon has no atmosphere. (Recall that an astronaut simultaneously dropped a feather and a hammer onto the moon's surface and that they both reached the ground at the same time.)

It is a fact, however, that the moon's gravitational field varies from place to place. These variations are thought to be due to so-called masscons, that is, concentrations of mass within the moon that act somewhat like huge magnets irregularly buried deep within the moon. The masscon hypothesis was advanced to account for observed irregularities in the trajectories of spacecraft orbiting the moon. It is, in effect, an elaboration of the falling-body model we have discussed. The elaborated model is the result of substituting a complex mathematical function (in other words, a subroutine) for the single term "acceleration" of our simple model. I mention it to illustrate the process, in this case properly applied, of elaborating a model to account for new and unanticipated observations. But the masscon elaboration was not the only possible extension of either the theory or its computer model. It could have been hypothesized, for example, that the moon is surrounded by a turbulent ether mantle whose waves and eddies caused the spaceship's irregular behavior. There are dozens of very good reasons for rejecting this hypothesis, of course, but a good programmer, given a lot of data, could more or less easily elaborate the model with which we started by adding "ether turbulence subroutines" so that, in the end, the model behaved just as the spaceship was observed to behave. Such a model would, of course, no longer look simple. Indeed, its very complexity, plus the precision to which it carried its calculations, might lend it a certain credibility.

Earlier I said that the value of a theory lies not so much in the aggregation of the laws it states as in the structure that interconnects them. The trouble with the kind of model elaboration that would result from such an "ether turbulence" hypothesis is that it simply patches one more "explanation" onto an already existing structure. It is a patch in that it has no roots in anything already present in the structure. Computer models have, as we have seen, some advantages over theories stated in natural language. But the latter have the advantage that patching is hard to conceal. If a theory written in natural language is, in fact, a set of patches and patches on patches, its lack of structure will be evident in its very composition. Although a computer program similarly constructed may reveal its impoverished structure to a trained reader, this kind of fault cannot be so easily seen in the program's performance. A program's performance, therefore, does not alone constitute an adequate validation of it as theory.

I have already alluded to the heuristic function of theories. Since models in computer-program form are also theories (at least, some programs deserve to be so thought of), what I have said about theories in general also applies to them, perhaps even more strongly, in this sense: in order for us to draw consequences from discursive theories, even to determine their coherence and consistency, they must, as I have said, be modeled anyway, that is, be modeled in the mind. The very eloquence of their statements, especially in the eyes of their authors, may give them a persuasive power they hardly deserve. Besides, much time may elapse between the formulation of a theory and its testing in the minds of men. Computer programs tend to reveal their errors, especially their lack of consistency, quickly and sharply. And, in skilled hands, computer modeling provides a quick feedback that can have a truly therapeutic effect precisely because of its immediacy. Computer modeling is thus somewhat like Polaroid photography: it is hard to maintain the belief that one has taken a great photograph when the counterexample is in one's hands. As Patrick Suppes remarked,

> The attempt to characterize exactly models of an empirical theory almost inevitably yields a more precise and clearer understand-

> ing of the exact character of a theory. The emptiness and shallowness of many classical theories in the social sciences is well brought out by the attempt to formulate in any exact fashion what consitutes a model of the theory. The kind of theory which mainly consists of insightful remarks and heuristic slogans will not be amenable to this treatment. The effort to make it exact will at the same time reveal the weakness of the theory."[8]

The question is, of course, just what kinds of theories are "amenable to this treatment?"