ARE THE MAN AND THE MACHINE RELATIONS?

Burton R. Wolin
System Development Corporation
Santa Monica, California

Summary

Humans are considered as general purpose computers. They ask two questions of the environment, "What is most likely to be the situation next?", and "What do I do now?" A research program is described which seeks to determine how, and how well, humans can answer the former question or predict the environment.

Introduction

As environments requiring control have become more complex, and the speeds of events in those environments have increased, there has been a trend to supplementing or replacing men, with computers.

We can first ask the most general question about complex systems: What are they primarily intended to do?

The answer is that they are designed and produced to give some measure of prediction and control over some aspect or aspects of the real-world environment. To accomplish this general task they must have sensors, so that they can obtain data from the environment. They must also have some power to discriminate between events of interest in the environment and others. They must be able to analyze, synthesize, organize, and classify the data events of interest. They must, of course, have effectors and some control over them.

What have we described? We have described a government, a business enterprise, a military system, an air-traffic-control system, and a biological organism. We have described in general what has recently come to be called an "open system," that is, any system which maintains or increases its organization by feeding off the environment.

Chapman, Kennedy, Nevell, and Bicl¹ pointed out that a complex of men and hardware performing a complex task could be likened to an organism; that such system grows organization and learns from and adapts to its environment.

Chapman et al studied the manual air-defense system, which was a system where men were the central components. They performed all of the significant operations in the system. They discriminated, classified, and calculated. They figured out what the situation was and what to do about it.

This manual air-defense system was also, interestingly, one of the first systems which became inadequate to cope with the increasingly complex, fast changing environment. The system was modified, in two stages; first, to employ a special-purpose computer to aid the control function, and second, to employ a general-purpose computer to aid in the performance of the system functions generally.

The supplementing or replacing of men by computers has been forced upon us so fast that occasionally some unease is felt about the whole process. In many ways we were neither ready, nor prepared, for this development.

There are many statements made, based on traditional beliefs, about what men can and cannot do, or about how they should and should not be used. These statements lack a basis of evidence and are quite imprecise.

It is well known that given enough time, and a problem which is not too complex or too imprecise, men can do a respectable job of solving such a problem. Exactly what the relations are between enough time, amount of complexity, and degree of precision are not known. All we have really said is that men have limitations. We also know amazingly little about how men do solve problems, make decisions, and perform in general high-level intellectual functions. We do have a set of terms for the functions they have traditionally performed, with computers.

The answer is that they are designed and produced to give some measure of prediction and control over some aspect or aspects of the real-world environment. To accomplish this general task they must have sensors, so that they can obtain data from the environment. They must also have some power to discriminate between events of interest in the environment and others. They must be able to analyze, synthesize, organize, and classify the data events of interest. They must, of course, have effectors and some control over them.

What have we described? We have described a government, a business enterprise, a military system, an air-traffic-control system, and a biological organism. We have described in general what has recently come to be called an "open system," that is, any system which maintains or increases its organization by feeding off the environment.

Chapman, Kennedy, Nevell, and Bicl¹ pointed out that a complex of men and hardware performing a complex task could be likened to an organism; that such system grows organization and learns from and adapts to its environment.

Chapman et al studied the manual air-defense system, which was a system where men were the central components. They performed all of the significant operations in the system. They discriminated, classified, and calculated. They figured out what the situation was and what to do about it.

This manual air-defense system was also, interestingly, one of the first systems which became inadequate to cope with the increasingly complex, fast changing environment. The system was modified, in two stages; first, to employ a special-purpose computer to aid the control function, and second, to employ a general-purpose computer to aid in the performance of the system functions generally.

The supplementing or replacing of men by computers has been forced upon us so fast that occasionally some unease is felt about the whole process. In many ways we were neither ready, nor prepared, for this development.

There are many statements made, based on traditional beliefs, about what men can and cannot do, or about how they should and should not be used. These statements lack a basis of evidence and are quite imprecise.

It is well known that given enough time, and a problem which is not too complex or too imprecise, men can do a respectable job of solving such a problem. Exactly what the relations are between enough time, amount of complexity, and degree of precision are not known. All we have really said is that men have limitations. We also know amazingly little about how men do solve problems, make decisions, and perform in general high-level intellectual functions. We do have a set of terms for the functions they have traditionally performed, with computers.
much and what is learned?

4. What are the measurable characteristics of men which influence how much and what will be learned from (particular) experiences?

One cannot go to a book and get the answers to these questions. Later, a research program will be described which is intended to help answer some of these questions.

Some system designers have decided that the best way to deal with man's limitations or what man cannot do is to eliminate him as far as possible from a system. They attempt to design completely automatic functions to be performed wholly by the computer. To the extent that any such attempt fails, which sometimes happens, man is used as backup: The function becomes "semiautomatic." It is frequently found, however, that the man can't help. To the extent that the automatic function fails, the semiautomatic function also fails. This failure of man to remedy the problem reinforces the attitude which such designers started with - that men are no darned good. This is the wrong conclusion. Where other system designers have started out to design semiautomatic functions, and have treated men with the same care they would any component, such semiautomatic functions have proven highly successful.

If one remembers that we are here discussing man-computer systems which operate in real-time with high data loads upon complex problems, one other comment is in order. The designers of a function frequently find that they have a choice as to the set of operations to use to fulfill a function. At least they have an apparent choice. However, the particular characteristics of the computer with which they must work almost inevitably reduces the number of choices actually available. A general-purpose computer is general purpose in much the same way a man is general purpose: Given enough time, it can perform a function more or less well.

Having seen men working alone fail and succeed, and men and computers working together both fail and succeed, it seemed reasonable to assume that not enough was known about either man or computers or their interaction with one another. One way of finding out more is to do research.

The research program I shall describe has two objectives. The first objective is to study the behavior of men in complex environments to find out what they can and cannot do well, and what factors limit or increase their effectiveness. The second objective is to study men as analogues of general-purpose computers, to determine how they perform complex functions, so that we can learn how to program (and maybe even design) computers to perform such functions when the real-time requirements will make use of the man impossible.

Such complex environments as those against which we wish to apply man are available in many real cases. Actual systems operating in actual environments could provide a laboratory for such research. However, as anyone knows who has ever tried, research to solve the problems of a large-complex man-computer system using that system as a laboratory vehicle presents difficulties. First, research is very slow, because changing anything in a system is a major enterprise. Second, research is imprecise, because changing anything in the system may have far ranging effects which were unanticipated, unsought, and very confusing. Third, the research is somewhat wasteful, because even if a particular problem is solved, the investigator is highly uncertain if the solution will ever work again. The generality of what he has found is unknown.

It seems a reasonable proposition that if the problems one encountered in working on systems could be extracted or abstracted and generalized, the appeal to research could be simplified. One always investigates some aspect of a system at a time anyway, rather than all aspects at once.

What important problems in systems can be extracted, abstracted, and generalized? Human operators in systems face two major situations over and over.

First, some have to make decisions about what to do: They have to answer the question "what do I do now?" To answer that question they have to answer a prior question, "what is the situation?" or "what is most likely to be the situation next?"

Second, for some operators there are rules like "if A then do B". These operators have the problem of determining when A has occurred, or anticipating when A will occur. Of course, translating to a question, these operators have a job very similar to that of the former ones: "What is the situation?; etc."

It is possible to decompose general questions into more detailed questions, and this we can do for "what is the situation?" To answer this, we may need the answers to many more specific questions, like

What is it?
Where is it?
How many are there?
Where is it going?
What is it doing?
When is it going to happen again?

The operators may have to process data which contain not only the answer to the question they seek but irrelevant or incompatible data as well. The data which contain the pertinent information may be at a detailed level, so that the operator must summarize or aggregate
ARE THE MAN AND THE MACHINE RELATIONS?

We obtain optimal performance. Optimal performance may be specified and perhaps measured. It would be generated by a source or sources. These data events may contain information relevant to the question to be answered. The relevant information in the data events may or may not be adequate to reduce the operator's uncertainty completely and may or may not contain redundant elements.

These same data events may contain noise, or irrelevant or incompatible data.

The data events may be presented all at once to the operator or may be presented over some interval of time as a sequence.

In any case, the operator must perform certain operations to obtain or produce the answer to the question. He receives data events which have been generated by some sources(s) and perhaps operated upon already by some other device(s). The data events may contain information relevant to the question to be answered. The relevant information in the data events may or may not be adequate to reduce the operator's uncertainty completely and may or may not contain redundant elements.

The Model

We assume that our human general-purpose computer has the task of obtaining the answer to a question. He receives data events which have been generated by some sources(s) and perhaps operated upon already by some other device(s). The data events may contain information relevant to the question to be answered. The relevant information in the data events may or may not be adequate to reduce the operator's uncertainty completely and may or may not contain redundant elements.

These same data events may contain noise, or irrelevant or incompatible data.

The data events may be presented all at once to the operator or may be presented over some interval of time as a sequence.

In any case, the operator must perform certain operations to obtain or produce the answer to the question. Loosely described, he must separate relevant from irrelevant data and perform coding operations on the relevant data to organize it and extract the answer to his question. This means that the operator, like an artificial automaton, can be thought of as asking a sequence of questions, performing necessary operations to obtain the answer to each as they are raised, and going on to the next until he has the answer to the final question. Also, as with the computer, neither the questions themselves nor the operations to obtain the answers to them need be in a fixed, predetermined sequence.

If the foregoing analogy holds, then the transfer function through our human general-purpose computer can be flow diagrammed, and the logical operations which must be performed can be specified and perhaps measured. It would then be possible to do two things: We could estimate how long it would take a human operator to extract the answer to a question from any data event, and we could determine how to program the human operator and arrange the data event so that we obtain optimal performance. Optimal performance, of course, may turn out to be inadequate, so we can expect to find limitations to the use of humans for many data-processing tasks.

Recapitulating, complex data events are generated by a source or sources. These data events may contain relevant information with varying amounts of redundancy. They may also contain noise or irrelevant or incompatible data. The human data processor, like a computer, performs operations upon the data event. The desired output or product is an answer to some question. Data processing is the extraction of information from data events. Figure 1 illustrated the notion of an information-to-data ratio. It can be seen that the more redundancy and noise there are in the data, the harder it is to find the information. Figure 2 shows the general model described previously. This model will be the basis for the discussion to follow.

The Research Program

Rather than accept traditional beliefs about what men can or cannot, should or should not, do, our approach is to subject these beliefs to empirical study.

What of the belief, which we have already cited, that important decisions are the responsibility of man?

Ward Edwards, at the September 1961 meeting of the Human Factors Society, gave a paper (published in the April issue of Human Factors) contending that men are not good decision makers but are good situation analyzers. He was referring specifically to the case where data are fuzzy, incomplete, or probabilistic in nature.

Edith Neimark and Edwards have both conducted experiments where their subjects had to make a decision. They could purchase relevant information as they wished. There are large and consistent individual differences between subjects. The subjects did not maximize to expected values, but tended somewhat in that direction. Interestingly, the subjects tended to buy too much information. That is, they paid for and obtained more information than they needed but did not use it in the best possible way in making a decision. It should be noted that all the information was relevant. Where it was not, one would expect as much or more purchase, but even less effective decision.

What have we seen? We have seen that men can be considered in some sense like general-purpose computers; they solve arithmetic and logical problems if given enough time or if the problem isn't too complex. However, when the problems are complex and time is limited, men have trouble problem solving: They do not behave like the ideal rational man. We have also seen that certain traditional beliefs about man's capabilities and limitations have tended to influence design of man-computer systems, but that these beliefs have lately been subject to inspection.

It will be remembered that a system, of the sort with which we are concerned, obtains data from and about the environment of interest. What the system can learn about the environment, or what questions it can conceivably answer about the environment, depends on the nature of
the environment itself. Additionally, a limiting factor is the nature of the data about the environment available to the system. If the data events come to the system in strings or as a sequence, predicting or controlling the environment requires that the system find the order or pattern of the environmental events.

An environment can have two different kinds of order or pattern. One, for all practical purposes, can be thought of as determined or completely predictable. This simply means that the probability of any deviation from some rule is very low. Many events are of this kind. For instance, an overwhelming percentage of the time one can predict that a depression of a light switch will extinguish an electric light, or that when a letter 'Q' is seen in the English language that it will be followed by a letter 'U'. The general description of this kind of event is a very high conditional relation between some two events such that if A, then, effectively, always B.

The second kind of order or pattern in an environment is stochastic order. This kind of order is common in environments and can be troublesome. The old coin flipping problem is a good illustration of a simple case of this kind of order. If the coin is unbiased, a sequence of such events will tend to contain an equal number of head and tail events. If the coin is biased, a sequence of such events will tend to contain the numbers of heads and tails representing the amount of the bias. If the coin tossing mechanism is biased, or if coins with different amounts of bias are systematically rotated in the order of tosses, one can expect to get probabilistic contingencies between successive events. The differences between these cases are important.

If two events are each equally probable for all reasons, any prediction one can make will lead to chance success.

If two events are not equally probable, but the only source of order is in the event frequencies, then the best strategy is to predict always the most frequent event.

However, if there is some source of order in addition to the event frequencies, a much more complicated situation obtains: It is necessary to identify the contingent relations and adopt a more complex optimal strategy. Let us illustrate the difference in the latter two cases.

Figure 3 shows two generators, both resulting in sequences having .5 A events as the over-all event frequency. The generator shown in figure 3b has an additional property: Redundancy of no mean amounts between odd and even draws.

If an analyzer were presented with a sequence from one of these two generators, it would have two questions to answer: First, what characteristics does the generator have, and second, what strategy should be employed to optimize predictions?

If the analyzer were sensitive only to event frequencies, the first question for both generators would be answered that the generator is producing an equal number of A and B events.

If, on the other hand, the analyzer were able to find additional redundancy, it would report quite a different answer to the first question with respect to the sequence generated from b.

We haven't yet said anything about the answer to the second question, about strategies to optimize predictions. Remember that our analyzer is a human computer. It can be shown that the best strategy is to always pick A on odd draws and B on even draws. Does he optimize?

Quite simply, the answer is no. The human computer tends to find the pattern in sequences generated from sources like 3a, but to match the characteristics of the source. That is, he comes to predict A on odd draws about .7 times, and on even draws about .3 times. Now, it can be shown that by chance .7 of the A events he predicts on odd draws will be A's, and .7 of the B events he predicts on odd draws will be A's, while .3 of the A events he picks on odd draws will be B's, and .3 of the B events he picks on odd draws will be B's. Contrariwise, the same for even draws. Thus matching the generator 3b will result in, on the average, accurate prediction of .7 A events on odd draws and .7 B events on even draws. This is substantially better than an analyzer which determines over-all event frequencies only, where the prediction success would be .5 for A and B events on both odd and even draws. However, it can also be shown that the variance is not at a minimum when a matching strategy is employed for prediction. That is, for a given sample sequence from generator 3b, predictive success can be much worse by chance than .7. Variance is reduced to a minimum by always predicting A on the odd draw, and never predicting A on the even draw for sequences from generator 3b. If this be so, why doesn't our human computer employ the better strategy? The answer seems to be--and this statement does not come easily from a hard-headed behaviorist--that our human computer is a tele-logician. What does this mean? It means, first, that humans have goals. In this case the goal is to approach perfection insofar as possible. It means, second, that humans seek order or pattern in the world; they sometimes try to find the rules which govern the generation of events.

You will note that it is only after we have been able to describe completely the nature of the generator that it is possible to state the optimal strategy, and that in our example, we knew the characteristics of the generator from the outset. The human computer does not know the nature of the generator in an overwhelming number of cases with which he must concern himself. The important point here is that adopting an optimal strategy and discovering the characteristics of a generator...
ARE THE MAN AND THE MACHINE RELATIONS?

concurrently are incompatible and logically impossible operations. Once one decides to optimize, any additional unfound source of order won't matter. If one seeks more order, and expects to find it, it is ridiculous to consider any strategy optimal prior to finding that order.

Two uncertainties tend to keep human computers from settling on an optimal strategy unless the order seems completely predictable. The first uncertainty is whether the generator is stationary. That is, the rules by which the generator operates may change. The second uncertainty is whether there are additional sources of redundancy which the human computer has not yet been able to find. Generator 3b has only digram redundancy, or event pairs relationship, but a generator may have higher order, or N-gram relationships which would improve prediction, or even make the sequence completely predictable. An example of a very difficult case would be a random sequence of two kinds of events ten events long which then repeated itself. The problem in some sense is similar to learning to spell a long word, or even worse, to recognize variants as being the same long word when one's students are the generators.

A generator, then, can have many sources of order, or redundancy, such as a language has: The patterning may be very complex and the relations subtle. The order may be either stochastic in nature, or completely predictable, or some of each (such as U following Q but A, L, R ... following B). What appears to be stochastic order at one level of organization can become determine order at a higher level of organization (houses in a tract all being constructed in different orders and sequences all end up looking identical).

It should be clear from the above discussion that what at a very detailed level may be considered as a simple binary choice—that is, an event will or won't occur, or one or another of two events will occur—may turn out to be anything but a simple binary choice when strings of such events have been appropriately grouped into subsets. It is less obvious, but still true, that the entities which are produced by organizing lower order entities are recognizable as entities, or specific ordered sets, and not simply as collections of elements. These higher-order entities are manipulated, not the elements of which they are composed.

In addition to inventing generators which produce strings of events having various kinds of stochastic and determined order, which can be thought of as artificial languages, we have used English language prose to get at some of the same phenomena.

The technique we have used was one developed by Shannon. Our human computer is asked to guess his way, letter by letter, through a passage of English prose. He is told after each guess whether the letter he guessed is right or wrong. He has a twenty-seven letter alphabet, A to Z plus space. Shannon used this technique to measure average redundancy in English. His analysis was made at the letter level. That is, he measured the average number of guesses for the nth letter, given n minus one prior letters. Our interest, and therefore our use of the technique, is different. As the previous discussion indicates, we are interested in organization through levels. Specifically, we wish to know what characteristics of the organization of data events influence redundancy or the ability of our human computers to predict and how our human computers go about the job of reorganizing and predicting as they accumulate data.

Table I shows three sequential arrangements of eleven words taken from the five hundred most frequently used words in the English language (actually it is possible to devise about twenty different sentences using these eleven words, and fifteen different sentence variants were given to five subjects each). As we go from top to bottom in our table, from variant to variant of the word sequences, it can be seen that the sentence is decomposed; that is, less and less integrated, or organized.

Our human computers guess their way through the highly organized variant 1 with an average of 121 guesses. An average of 173 guesses were required to get through variant 2. An average of 233 guesses were needed to get through variant 3. (Actually variant 3 was presented as a list of words, not a sentence). There was no overlap in scores between these two groups. Table II shows the data for five subjects each on these three variants.

The sentences (or list for variant 3) have 53 character positions. Actually, 10 of the character positions were completely redundant in all cases for all subjects. For example, once RUNNI was guessed, no more than one guess was needed to complete RUNNING (space).

Now, it may look like this was a very clever experiment with a clean hypothesis. Actually, the hypothesis that word arrangement would influence uncertainty was a good guess, but how it would do so turned out to be complicated.

The identification of exactly what variables contribute to "organization" is a difficult problem. However, the data show one distinct feature (which does nothing to answer the question). Whatever contributes to organization, a sequence which is well organized results in rapid early decrease in uncertainty, whereas a poorly organized arrangement shows either linear or uneven decrease in uncertainty.

Our fastest subject guessed his way through the 43 character positions in 94 guesses, or an average of just under 2.2 guesses per letter. Subject 5, guessing his way through the list of poorly arranged words in variant 3 of Table II, took 298 guesses, or an average of exactly 6
guesses per letter. It should be remembered that in both cases 10 character positions required only one guess each.

These data have permitted many other analyses not discussed here, but I believe the point is clear. Data processing by a human general-purpose computer is clearly a function of the organization characteristics of the data to be processed. This seems to be true because the human general-purpose computer is searching for organization in the data and uses data-processing techniques which yield the best results when the data are highly organized.

Can a computer help a man who is attempting to predict events where the organization or pattern is not known to the man? The question should perhaps be modified to how rather than whether the computer can help. To use the computer simply as a data transducer may not help and may impair the man's performance. If the computer can be used to organize the data, there is little doubt that computers can contribute mightily to the intellectual pursuits of man.

A Few Words about Automation of Research

The data we have obtained to date on both artificial and natural languages were obtained and processed by manual methods, or blood, sweat, and tears.

To obtain the data from 65 subjects for the sixteen word arrangements described above required each experimenter to spend 12 to sixty five minutes on each subject individually. Data reduction and analysis took weeks.

Since preparation of this paper began, our Systems Simulation Research Laboratory has become operational for human data processing research. We currently are collecting data from six subjects simultaneously. Shortly the capacity will be twenty four at once. This means that in three hours on the computer, we will increase our data-collection capacity over that of a man 120 times. Weeks of data reduction and analysis is reduced to minutes. In the intellectual pursuit called research there is no question that a computer can help significantly.

The key term for describing our laboratory is "general purpose." The computer, a Philco Transac, is a general-purpose computer. We program the computer with a general-purpose procedure-oriented language called JOVIAL which is a first cousin to Algol. The programs we write are highly parameterized and modularized so they can be used for a variety of purposes. The console equipment for communication between experimental subjects and the computer is modular, so that we can assemble a console from standard parts. Further, this equipment has been designed so that given modules are truly general purpose; the content, or meaning of displays and button actions are independent from the equipment. Like the telephone system, our system doesn't care what is talked about.

We have programs which generate, make up and distribute displays, read and interpret button insertions of our subjects, record specified data, and organize, reduce and analyze that data. The programs can be used for quite different research purposes. The attempt has been to design the laboratory on the order of a quick change artist, so that it can be made to resemble that which the investigation requires. Only one question remains: given the physical facility, do we have the intellectual facility to exploit our opportunity?

References


TABLE I Gradual Decomposition of a Sentence

1. The boy just got in time to the school running fast.
2. The boy just got to running fast the school in time.
3. Time the just to running got boy school fast in.

TABLE II Total Guesses For Sentences from Table I

<table>
<thead>
<tr>
<th>CONDITION</th>
<th>IX</th>
<th>XV</th>
<th>XVI</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>154</td>
<td>117</td>
<td>215</td>
</tr>
<tr>
<td>2</td>
<td>164</td>
<td>117</td>
<td>215</td>
</tr>
<tr>
<td>3</td>
<td>177</td>
<td>121</td>
<td>230</td>
</tr>
<tr>
<td>4</td>
<td>183</td>
<td>121</td>
<td>246</td>
</tr>
<tr>
<td>5</td>
<td>186</td>
<td>130</td>
<td>258</td>
</tr>
<tr>
<td>X</td>
<td>173</td>
<td>121</td>
<td>233</td>
</tr>
</tbody>
</table>

INPUT

Data
Redundancy
Noise
irrelevant data
incompatible data

Information-to-Data Ratio

Fig. 1. Information-to-Data Ratio

Fig. 2. Model for Data Processing

Fig. 3. Two generators with common event frequencies, different redundancy.