MEMORY SYSTEMS STUDY

by

G. R. Peterson
Principal Investigator
and
R. C. DeVries and J. L. Melsa

ENGINEERING RESEARCH LABORATORIES

COLLEGE OF ENGINEERING
UNIVERSITY OF ARIZONA
TUCSON, ARIZONA

SIGILLVM VNIVERSITATIS ARIZONENSIS
SVRSVM
ESTABLISHED 1885

DISCLAIMER

SC-DC-3568

Annual Report No. 2

16 Nov. 1962 to 15 Nov. 1963

MEMORY SYSTEMS STUDY

FOR

Sandia Corporation
P. O. Box 5800
Albuquerque, New Mexico

BY

G. R. Peterson
Principal Investigator
and
R. C. DeVries and J. L. Melsa

Engineering Research Laboratories
Office of the Dean
College of Engineering
The University of Arizona
Tucson, Arizona

PART I

LEARNING AND ADAPTIVE CONTROL SYSTEMS

ABSTRACT

Results of theoretical studies of learning control systems are presented. The need for definitions is discussed and definitions of successful, adaptive, and learning control systems are presented. The basic structural elements of learning control systems are discussed. The environmental characteristics of control situations in which learning may be applicable are discussed. Learning control systems are classified in accordance with the environmental situation in which they might operate. The structure and components suitable to various environmental situations are discussed.

SECTION 1.1 INTRODUCTION

In previous reports we have discussed at some length the importance of producing a coherent engineering description of the processes loosely grouped under the generic term "machine learning". In the following sections we shall discuss some of the aspects of this problem of producing an engineering description. First, we will discuss the problem of definitions of the terms involved. Second, we will discuss the structural elements of machines and devices which have the characteristics of adaptation and learning. Third, we shall consider the characteristics of environmental situations which might indicate the need for systems having such capabilities. Fourth, we shall consider the relationships between the problem of machine learning and the problem of self-repairing systems. Fifth, we shall discuss the problems which remain to be solved and the direction which we believe future research should take.

SECTION 1.2 DEFINITIONS

In attempting to study the characteristics of learning systems it has been increasingly obvious that some definitions are needed. It should be noted that the desire for a definition is not motivated by the desire to define <u>per se</u>, but rather by a need to put discussion on some firm basis. For example, consider the question as to whether a learning system can be open loop. One method of showing that a learning system can be open loop is to create an example, but anybody else can immediately dismiss the example by stating that it is not learning. If we can establish some sort of firm definitions, then at least we have a chance of removing such subjective factors from discussions.

As discussed in earlier reports there are two types of definitions that we can make, the <u>functional</u> definition and the <u>elemental</u> definition. The functional definition is the definition in terms of what the system does, of its external or "black box" characteristics. The elemental definition, on the other hand, is the definition in terms of what the system is built out of, of how the building blocks are tied together and what is in the building blocks. The functional definition is, in general, to be preferred where it is possible to find one, since this is the type of definition which can be most easily couched in mathematical terms and can therefore be considered to be most precise. The elemental definition invariably involves a subjective choice as to the model of the system which we wish to use. Therefore, if the definitions are completely elemental, then the classification of a system will vary depending upon the model chosen. On the other hand the functional behavior is fixed, independent of the model, so that a functional definition will eliminate some ambiguity of classification. Therefore, our attempt in this study has been to reach definitions as nearly functional as possible. We have not found it possible to produce completely functional "black box" definitions or descriptions of either adaptive or learning processes. In the first annual report an example was given to show why no functional definition of adaptivity could ever be completely sufficient.

Although our primary concern has been with learning systems we have found it necessary to study quite carefully the concept of adaptivity. This is because of the fact that adaptivity is of necessity one of the characteristics of learning systems. So, until we know what adaptivity is, we cannot have any very clear understanding of what learning is. There has certainly been a great deal more activity and effort devoted to adaptive systems than to learning systems, with the result that a great deal more has been written, and some attempts have been made at definitions. However, none of them seem to really fill the needs of our study. The definitions so far formulated by workers in the field have either been so general as to convey no useful information, or have been so tied down to an arbitrary model as to make it quite impossible for any two people to reach any agreement as to what is being talked about. As an example of the too general definitions, we have the definition offered by Truxal[1] in which he said, "An adaptive system is any system designed with an adaptive point of view." One can scarcely argue with that definition, but it is completely circular and therefore doesn't really say very much of interest.

In the same category of a too general definition is one recently offered by Zadeh[2]. In this definition Zadeh draws on the biological concept of an adaptive system as one which responds in a satisfactory fashion to its environment. He simply says that we will consider as adaptive any system which performs satisfactorily with respect to some specific set of inputs and some specific set of requirements. This really comes down to saying that adaptive is synonymous with successful. Again, this does not seem to be a particularly useful point of view although you can't argue that it doesn't make some sense. As Donalson[3] has pointed out, if we accept this point of view, then every control system is adaptive, and the word is redundant when used to describe control systems.

On the other extreme of the definition which is perhaps too specific is that offered by Gibson[4], who says that an adaptive system is a system which exhibits the functions of modification, identification, and decision. It is perfectly true that most of the systems that have been so far constructed and which people have labeled adaptive have had these ideas involved in them. However, these ideas themselves are so complex that it is difficult to find two engineers that will agree on what is meant by identification, decision, or modification, and thus the purpose of a definition in trying to provide a grounds for common discussion is defeated.

Thus, it appears to us that none of these definitions seem to accomplish any useful purpose. We feel that what is needed is a definition which falls somewhere in between, one which is as functional as possible, and yet not so general as to provide no useful information, as is the case of Zadeh's completely functional definition. The key to our point of view is found first of all in Truxal's definition, in which we find the observation that there is a <u>point of view</u> involved in an adaptive system. In other words, the intent of the designer in creating the system is somehow involved here and it does not seem possible to separate this idea out completely. Also, it has been noted that all systems that have adaptive characteristics display nonlinearity in some manner. In fact, yet another definition that has been offered is that an adaptive system is a nonlinear system with a purpose. With this idea in mind we offer the following definition.

First of all, we must specify what we mean by successful performance of a control system. We assume we are concerned with a general system S, which we assume has a purpose or goal, i.e., there is something we want this system to do, and it is possible to measure the degree to which our purpose is being accomplished. There is, of course, always a subjective decision as to what should be considered a part of the system and what should be considered the "rest of the world". However we shall assume that this decision has been made. This system, when subjected to an input $\overline{X}_1$ responds in a manner which is indicated by an output $\overline{X}_2$ (Fig. 1). $\overline{X}_1$ is in general a vector-valued time function.

$$\overline{X}_1 = \begin{cases} X_{11}(t) \\ X_{12}(t) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ X_{1n}(t) \end{cases}$$

$\overline{X}_1$ includes not only inputs in the usual sense of information or energy deliberately applied with the purpose of stimulating the desired response, but also includes all environmental influences which <u>to our knowledge</u> have any influence on the success of the system in achieving its goals and can be measured. The output $\bar{X}_2$ is also a vector-valued time function in the general case.

$$\bar{X}_2 = \begin{cases} X_{21}(t) \\ X_{22}(t) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ X_{2m}(t) \end{cases}$$

By the output we do not necessarily mean output in the usual sense of, e.g., a shaft position or a certain amount of cement per hour. Rather, $\bar{X}_2$ is taken to be the set of <u>all</u> that information and <u>only</u> that information about the state and response of the system which is pertinent to the problem of measuring the performance of the system relative to its goals. As will be discussed later, some initial conditions of the system may be involved in the goal structure. If this is the case, these initial conditions will be considered a part of $\bar{X}_2$.

It is important to notice that there is a difference in our degree of knowledge about the inputs and outputs. In the case of the inputs we can never, except for trivial cases, be sure that we have considered all of those and only those factors which influence the response of the system. On the other hand, successful operation is a matter of definition and we can specify the factors which are to be considered in our goal. Further, we <u>should</u> specify our goal only in terms of those factors which we can measure. In other words, we are not here concerned with situations where a goal either cannot be defined, or cannot be measured, or both.

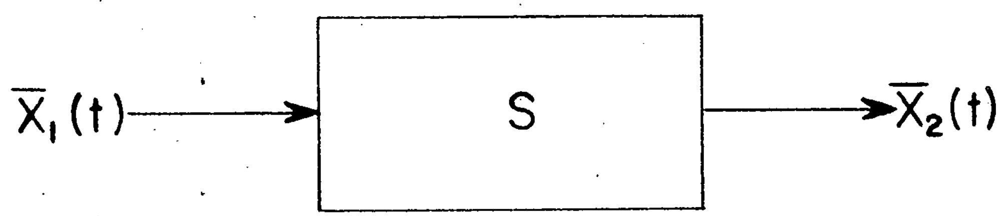


FIG. 1 BASIC MODEL OF CONTROL SYSTEM

Accordingly then, the performance is measured by a performance function

$$\overline{P}(\overline{X}_2,t)\begin{cases} P_1(\overline{X}_2,t) \\ P_2(\overline{X}_2,t) \\ \cdot \\ \cdot \\ \cdot \\ \cdot \\ P_k(\overline{X}_2,t) \end{cases}$$

a vector-valued function of $\overline{X}_2$ and t, with real components. Then we define $\overline{W}[\overline{X}_1(t), \overline{X}_2(o)]$ as the set of all values of $\overline{P}(\overline{X}_2(t),t)$ for which the performance of S is acceptable. Note that $\overline{W}$ may be a function of $\overline{X}_1(t)$ and $\overline{X}_2(o)$, i.e., what is acceptable may depend on the input and the initial conditions of S. The inclusion of initial conditions and the inclusion of t as an explicit variable in $\overline{P}$ are quite important.

The concept of adaptivity implies a change in the system after a change in the inputs and in a control system there will always be a requirement for some level of response to be achieved within some length of time. To put it another way, in control systems we are always concerned with a cost function, and the cost function always involves time.

Thus our measure of performance $\overline{P}$ includes not only a measure of what the system does, but also of how long it takes to do it, measured from specified starting time. And, what level of performance we consider acceptable may depend on where we start. It might appear that t should be a variable in $\overline{W}$ since what is acceptable depends on when it happens. This factor can be taken into account by weighting t in the expression for S in the proper manner, and then choosing $\overline{W}$ accordingly. It might be convenient to consider $\overline{W}$ a function of t , but we feel that it is important to emphasize the factors involved in the measure of performance, and the criteria of acceptable performance. The level of performance must be measured in terms of what the system is doing at any time, but what is acceptable is determined in advance, possibly in relation to what the input is and where the system started.

To illustrate this concept of the separation of measure of performance and criterion of acceptability, consider the example of an angular position control system. Assume that the steady state ($t > 10$) error must be less than 0.01 radian, and also assume the required speed of response after any change of input is specified by the following set of values:

at $t = 1$ sec $|\epsilon| \leq 0.02$ rad

at $t = .5$ sec $|\epsilon| \leq 0.05$ rad

at $t = .2$ sec $|\epsilon| \leq 0.25$ rad

This can be shown graphically as in Fig. 2a, where the shaded area under the curve represents satisfactory performance.

In order to implement this performance criterion, we shall try

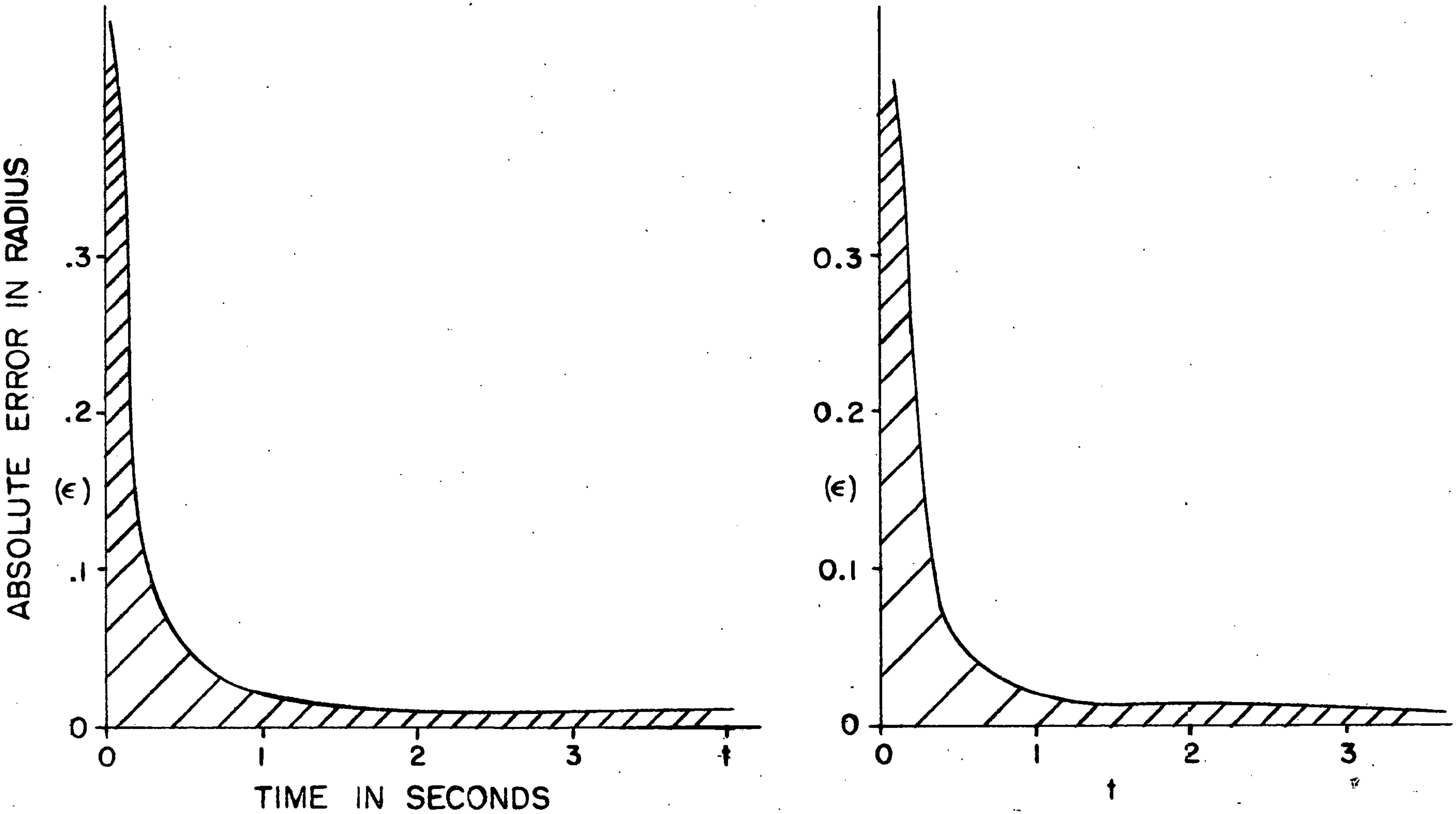
$$P' = |\epsilon| - \frac{1}{(10t)^2} \quad . \tag{1-1}$$

Fig. 2b is a plot of $|\epsilon|$ vs time for $P' = 0.01$. It will be seen that this is nearly identical to 2a. However this form of P includes the input since

$$P' = |\epsilon| - \frac{1}{(10t)^2} = |X_1 - X_2| - \frac{1}{(10t)^2}$$

So we will redefine P as follows

$$P = \left|X_2 - \frac{1}{(10t)^2}\right| \tag{1-2}$$



2a. ALLOWABLE ERROR VS. TIME

2b. $/\epsilon/$ vs t for

$$P = /\epsilon/ - \frac{1}{(10t)^2} = 0.01$$

FIG. 2 PERFORMANCE CURVES FOR ILLUSTRATIVE SYSTEM and define W as the set of all values of P such that

$$|X_1 - P| \leq .01 \ ,$$

i.e.

$$W = \left\{ P; \mid X_1 - P \mid \leq .01 \right\} \qquad (1\text{-}3)$$

Thus we have the measure of performance dependent only on the time behavior of the outputs, with the criterion of acceptable performance dependent only on the inputs. We believe that it should be possible to make this separation. As in this case, it may not always be convenient from a computational point of view, but we shall always indicate this separation in our general formulation of the problem in order to emphasize the concepts involved.

Finally, we will define $\{\overline{X}_1\}$ as some specified set of possible values of $\overline{X}_1$, generally those values for which the system is designed. With all these observations in mind, we make the following definition of successful performance.

Definition: A system S is successful on $\overline{X}_1$ with respect to $\overline{W}$ iff

$$\overline{P}[\,\overline{X}_2(t),t] \subset \overline{W}[\,\overline{X}_1(t),X_2(o)] \,\forall \overline{X}_1(t) \subset \{\overline{X}_1\} \ . \qquad (1\text{-}4)$$

Next we come to the problem of what is meant by adaptive. It appears, as discussed above, that this cannot be done on an entirely functional basis. It is necessary to make some assumption about the structure. Therefore, we shall use the following model. Let any control system be broken into two parts, the fixed plant F and the controlled plant C. (Fig. 3) By the fixed plant we mean that part of the system which, for technical or economic reasons, is not subject to structural alteration by the designer.

The controlled plant is, of course, the remainder of the system, that part of the plant which the designer can organize to suit his purposes. This decision is arbitrary and in some cases the location of the interface between fixed and controlled plant may be hazy, but in general it should not be too difficult to draw the line.

Let the inputs to the fixed plant be defined by the vector function $\overline{X}_3$. Without loss of generality $\overline{X}_3$ may be assumed to be the output of the controlled plant since if any component of $\overline{X}_3$ is also a component of $\overline{X}_1$, then that component will be passed through the controlled plant with unity gain. The inputs to the controlled plant are the components of $\overline{X}_1$ and information about the state and behavior of the fixed plant. This information consists of two parts, the components of $\overline{X}_2$, that information which is a factor in the computation of the performance measure, and any other information about the fixed plant which may be of use to the controlled plant. This latter part we shall call $\overline{X}_4$. Again without loss of generality we may assume that all components of $\overline{X}_2$ go to C . In the case of the open loop system the transfer function between $\overline{X}_2$ and $\overline{X}_3$ will be zero. Then we propose that a system shall be considered adaptive with respect to a specified input set $\left\{\overline{X}_1\right\}$ if and only if the relationships between the inputs and the outputs of the controller C cannot be specified by a linear integro-differential equation over all values of the inputs associated with the set $\left\{\overline{X}_1\right\}$. Formally, this may be stated as follows:

Definition: A system S is successfully adaptive on $\overline{X}_1$ with respect to $\overline{W}$ iff

$$P\left[\overline{X}_2(t),t\right]\subset\overline{W}\left[\overline{X}_1(t),\overline{X}_2(o)\right]\forall\overline{X}_1(t)\subset\left\{\overline{X}_1\right\}$$

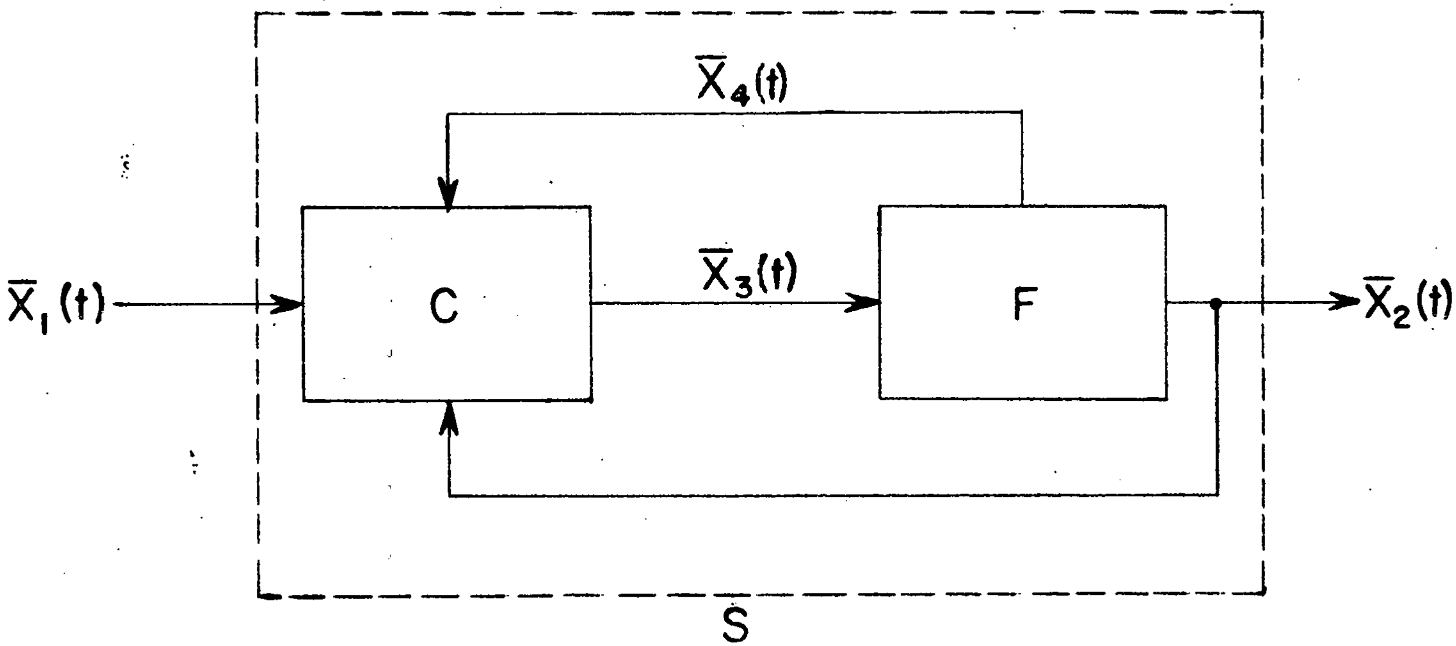


FIG. 3 BASIC MODEL OF ADAPTIVE CONTROL SYSTEM

and the relationship between the input and output of the controlled plant C <u>cannot</u> be described by an equation of the form

$$\sum_{k=1}^{4}\left[\sum_{n=1}^{N} A_{nk}(t)\,\frac{d^n X_k(t)}{dt^n} + \sum_{m=1}^{M} B_{mk}\int_0^t \cdots \int_0^{t_{m-1}} X_k(t_m)dt_1 \cdots dt_m \right. \tag{1-5}$$

$$\left. + D_k X_k(t)\right] = f(t) \text{ for all } \overline{X}_1(t)\subset\left\{\overline{X}_1\right\}$$

Now it will certainly appear at first that all we have done here is to state that any system which is nonlinear is adaptive. However, there is more to it than this. It will be noted that we have placed no restriction whatsoever on the form of the fixed plant <u>F</u>. We feel that this definition takes into account the concept that adaptivity generally implies nonlinearity deliberately introduced into the system by the designer. The controlled plant has been defined as that part of the system which is under the control of the designer. We assume that a designer will not deliberately introduce nonlinearity into his system unless it is the only way he can accomplish his purpose. Usually linear systems are easier to analyze, to synthesize, and to fabricate, and nonlinearities should be deliberately resorted to only in the case where the desired results cannot be obtained through the use of linear systems. What we are trying to include is a consideration of the idea that the intent of the designer is important in considering whether or not a system should be called adaptive.

As an example, suppose that we are designing a system which will be required to handle inputs with an extremely wide range of magnitude, a range of magnitude beyond that of any linear amplifier that could be designed. One way to handle this would be to put a separate servo loop on the input which will measure the input and, when the input goes beyond a certain magnitude, automatically turn down the gain of the amplifier. Such a system would fit in with the notion of parameter adjustment as measure of adaptive behavior. However, in this simple case at least, it is obvious that the same precise result can be accomplished simply by putting in a nonlinear stage which will saturate when subjected to inputs beyond a certain magnitude. Thus, the designer can obtain the same result by the <u>deliberate</u> introduction of a nonlinearity, rather than by putting in a device which in a more conventional sense "adjusts parameters" of the system.

In Fig. 4 we have three somewhat more elaborate systems which will illustrate the same idea. The system of Fig. 4a could be considered a system where the gain is reduced if the product of the input and output gets too large. Thus again we have effectively an adjustable gain, but the mechanism for so doing is not an actual "adjustment" of gain, as by turning a potentiometer. On the other hand, suppose we have a situation where we wish to maintain a linear relation between input and output of the system S, but the fixed plant exhibits a saturation characteristic, with the gain decreasing as its input increases. We might compensate for this with the system shown in Fig. 4b where we effectively increase the input to the fixed plant in such a manner as to compensate for the decrease in gain of the fixed plant. If the relationships between the fixed and control plants were as shown in 4b, we would obtain the desired results of a linear input-output relationship upon S. Figure 4c represents a combination of the above two systems and might represent a situation where our purpose is to maintain an approximately linear relationship over a certain range but at the same

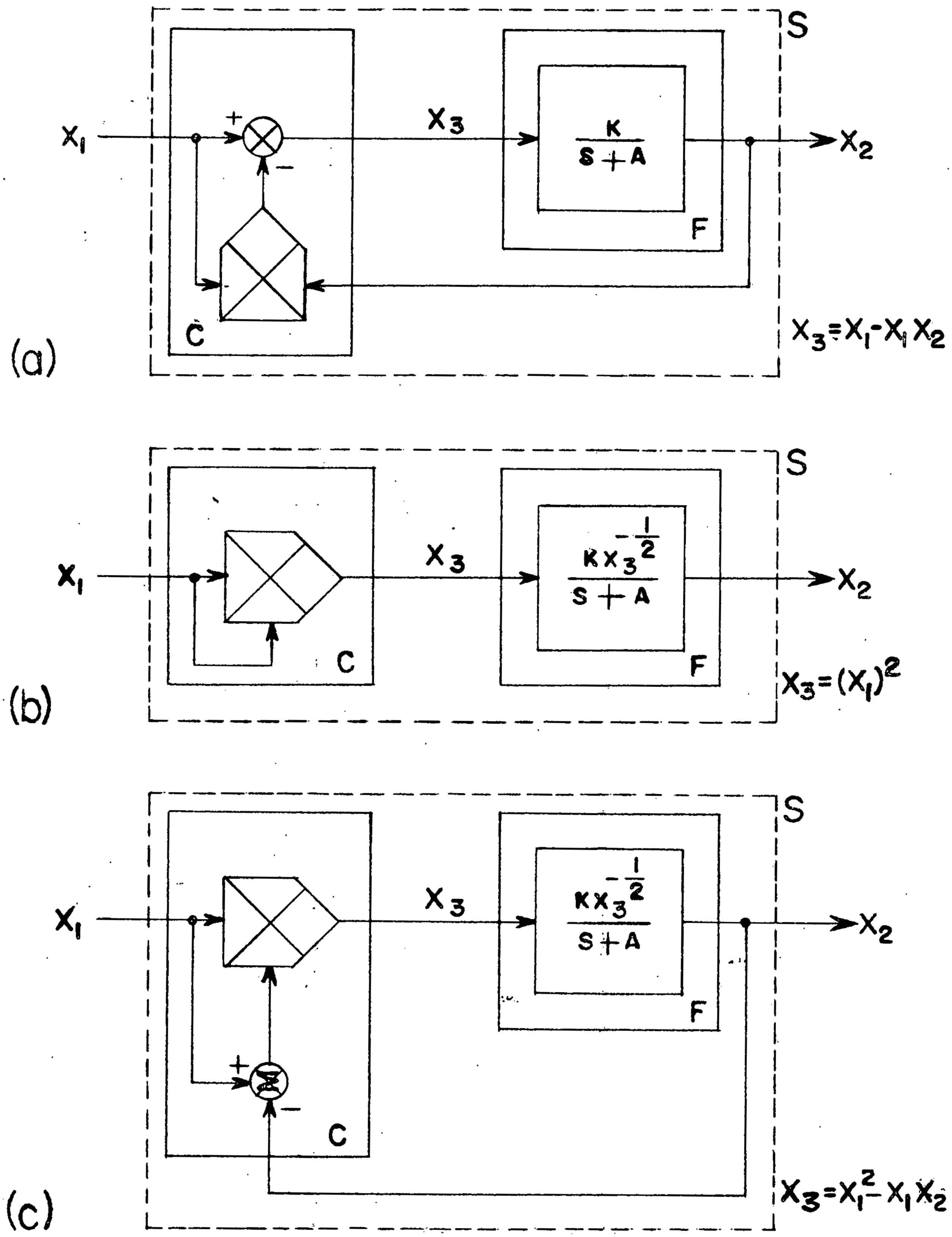


FIG. 4 EXAMPLES OF ADAPTIVE SYSTEMS

time to limit the excursions of the system. Again we have accomplished an essentially adaptive purpose but without the inclusion of explicit parameter variation as such.

It will be noted from Eq. 1-5 that time-varying coefficients are allowed in the non-adaptive system. The use of coefficients varying according to a prescribed function of time provides preprogrammed behavior. For example, we know that the behavior of a missile changes as it gains altitude and consumes fuel. These changes may make it necessary to change the gain, or other characteristics, of the guidance system. We could provide sensors to determine the altitude and weight of fuel, but this is not necessary since we know in advance how fast the missile will rise and consume fuel. So we provide for an automatic change of gain in accordance with these known rates. This method is perfectly satisfactory for some cases but we do not feel it should be considered adaptive. The system would not seem to be directly "responding" to the changes in the environment.

It may be felt that this form of definition does not really accomplish a great deal more than a definition such as the one Gibson has offered.[5] It still requires an arbitrary decision on the part of the person making the definition as to what shall be considered fixed plant and what shall be considered controlled plant. It certainly might be asked just why this sort of a distinction should be preferred to a distinction such as Gibson suggested, involving such functions as modification, decision and identification. We feel that the important factor here is that we have made the arbitrary decision fall at a point where it is a good deal more likely that two different people discussing a given problem can reach some agreement. In addition, a mathematical description is possible. The choice of what shall be considered modification, decision, or identification is in many practical cases a very difficult decision to make, and a mathematical description is seldom possible. Indeed, these factors tend to be rather mixed up with one another, and it is very difficult, for example, to separate the modification and decision functions in any clear cut fashion, as Gibson himself has pointed out. On the other hand, it is difficult to conceive of a situation involving control where it would not be fairly straight forward to make a distinction between fixed plant and controlled plant. In the case of systems whose primary function is not control in the usual sense, this may become somewhat more difficult. But at this point in the study our consideration is devoted primarily to systems whose purpose is control. If the definitions and analysis which we produce find useful application to other types of systems, that is all to the good, but that is not our primary purpose at this time.

Next we must consider the problem of what we mean by a learning system as distinguished from an adaptive system. As we have noted in the past, the idea of learning seems to imply, among other things, a sense of improvement in performance with time. However, we find that it is difficult in many cases to distinguish between an improvement in time due to learning and an improvement in time simply due to the time required for an adaptive system to carry out its function. In other words, how do we distinguish between a system that is learning and a system that is merely <u>still</u> adapting to a given input situation?

As an answer to this question Gibson has proposed the following:

"A Gedanken experiment might be proposed to determine if a given system is a learning system. Under a given set of environmental conditions the system parameters are given an initial offset from optimum and the system is allowed to operate. If it adjusts its parameters so as to optimize its performance in accordance with a given index of performance, it is adaptive. As yet it is impossible to know if it also includes a learning feature. Now return the system to the initial parameter setting and allow it to proceed. If the same gradual process of adaptation takes place, the system is not a learning system. If on the other hand it recognizes familiar patterns and utilizes this information to move more surely (or more rapidly) to the optimum, it is a learning system."[4]

This test definition seems intuitively reasonable, and we agree with the general ideas it seems to express. But it is not nearly so precise as it might seem, and in some respects seems to raise more questions than it answers. If you consider it as a hypothetical test to be applied to a completely specified system, every detail of which is known, it is fairly reasonable. But such a restriction will limit us to fairly simple systems, and if there is anything that we may be sure of at this time, it is that useful learning systems will not be simple. On the other hand, if we consider this to be a specification of a practical test to be applied to real systems, it is virtually useless.

First, the concepts of recognizing familiar patterns and utilizing information are rather subjective, and it would be difficult in the general case to specify what sort of characteristics might indicate the existence of such activity. Second, just what parameters are to be returned to their original state? If we assume that some, if not all, learning systems will utilize memory to store information about past performance, then surely we should not restore all parameters to their original state, since this would mean returning the memory to its original state. This would eradicate the information on which the learning is based. So should we return every part of the system except the memory to its original state? Perhaps so, but this will be practical only in the case where the memory is a clearly identifiable and distinct part of the system, such as the memory unit of a digital computer. This may often be the case, but we cannot at this time rule out the possibility of a more subtle and sophisticated memory arrangement, e.g., one in which the stored information is distributed throughout the system. So we conclude that we cannot in general specify what parts of the system should be returned to their original condition.

Finally, even if we could agree in some specific case as to what constitutes the memory and should therefore be left alone, how could we be sure that we had returned all pertinent factors to their original condition? As discussed above, in any practical case we can seldom be sure that we have considered all pertinent factors. So, when we consider all these problems, it appears that the test definition does not really specify a test at all, but simply describes the sort of internal activity which might characterize a learning system.

At one point in this investigation we thought it might be possible to remove the ambiguities from this test definition by breaking it down into a number of precisely specified steps. However, as was discussed in an earlier report, no matter how complex the procedure we specified, we found we could always counter it with some obviously trivial system which could pass the test. So, reluctantly, we must conclude that there can be no purely functional definition that will distinguish between learning and adaptive systems. Since it appears that a learning system will always include the capacity for adaptive action, this is not surprising in view of the fact that we cannot find a functional definition of adaptive action either.

The obvious step, then, is to try to produce a definition that is as nearly functional as possible, with the arbitrary distinctions drawn in such a manner as to make reasonable agreement possible, as we did for the adaptive systems. At present, we have not solved this problem to our own satisfaction, but we offer the following as a tentative suggestion. First, we shall start with the model proposed before, with the system divided into a fixed and controlled plant. In addition, we shall specify a <u>maximum adaptation time</u> T, the maximum time allowable for the system to achieve satisfactory behavior under our criterion of acceptability. It is apparent that in any practical control system there is a maximum time we can allow for the system to respond in a satisfactory manner. Just what this length of time should be in any particular case must be left to the judgment of the designer, but it is apparent that the minimum possible value for T will be governed by the inherent response time of the fixed plant.

Then we note that, in addition to satisfying the requirements already stated, a non-learning adaptive system should have no components in the controlled plant which, at <u>any</u> time, have response times longer than T. By response time we mean the time required to reach steady-state after a significant change in the inputs. Just what is meant by steady-state and what is a significant change will depend on the system, but in any specific case these terms should not be difficult to define. If there are parts of the controlled plant with response time longer than <u>T</u>, these will be separated out from the controlled plant and treated as a third section, which we shall refer to as the <u>memory</u>, <u>M</u>. (Fig. 5)

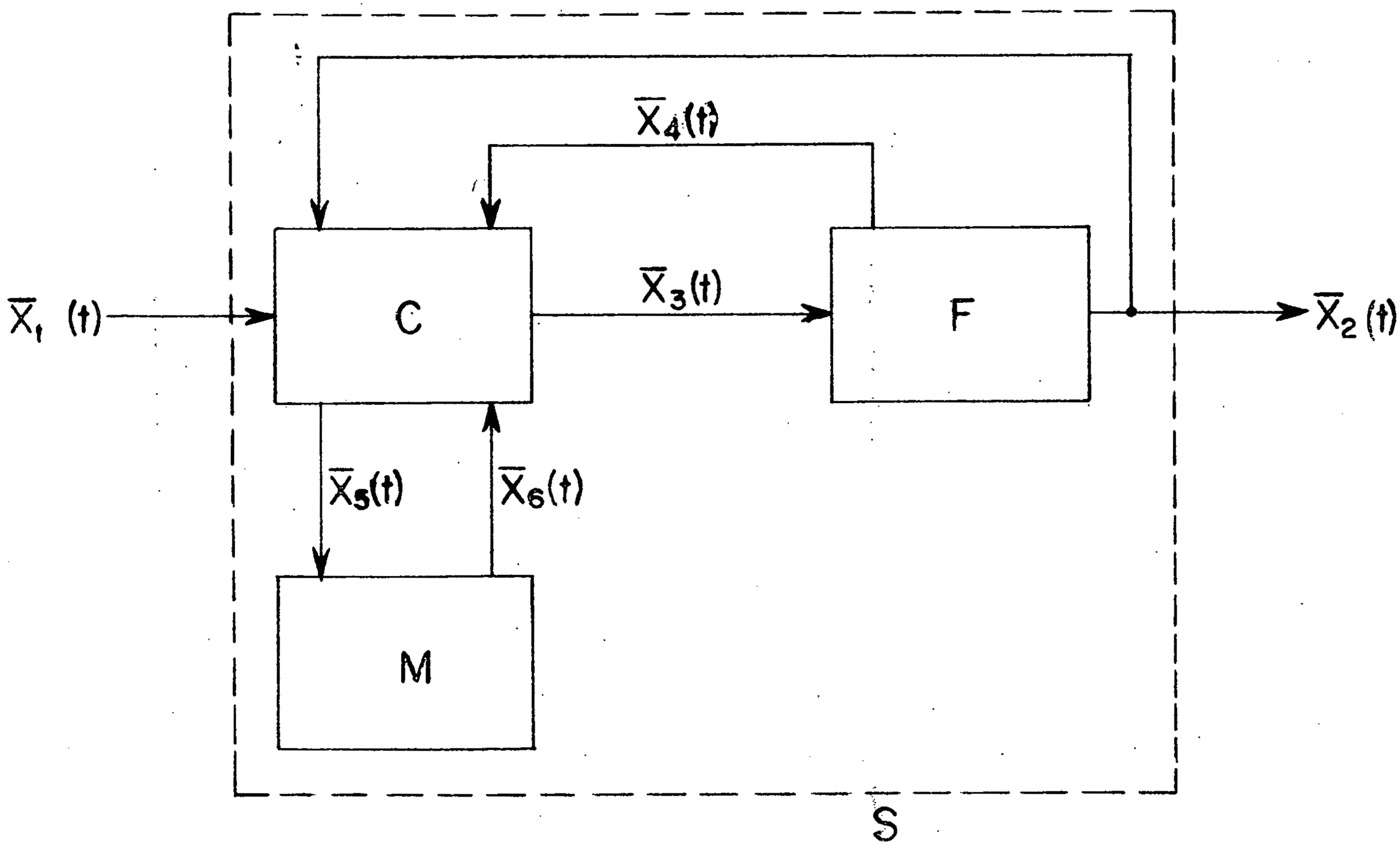


FIG. 5 BASIC MODEL OF LEARNING CONTROL SYSTEM

In a sense, what we are doing here is converting the system to a synchronous device, with time considered to come in discrete steps, with the length of these discrete steps determined by the time allowable for the system to achieve acceptable response. With the response time of the controlled plant restricted to be no more than the length of these discrete steps, the controlled plant may then be regarded as a sequential machine in the classical sense, with the state of the system being defined only at these discrete intervals, and inputs being allowed to change only at these discrete intervals. Thus we assume that at some time the system is in a particular state, as indicated by a particular output, and that this state is not changing. Now the input changes, marking the start of a discrete interval, and the state at the end of the interval will depend only on the state at the beginning of the interval and the nature of the change.

Further, after the completion of the interval T, the system is considered to be in a "suspended" condition, with no change in state taking place until the input changes again, marking the start of the next interval. This concept of "suspended" operation simply means that we are not requiring that changes of input occur only at a fixed periodic rate, but rather that the changes can occur no faster than a certain rate, and that the state of the system shall be defined only at the time of a change and at a certain fixed time after each change. This last distinction is perhaps a minor one, but it is made to clarify the conceptual steps that may be necessary in considering a practical control system as a synchronous sequential machine.

Within this same framework of discrete time intervals, the slow-response section that we have set apart can generally be considered a finite-memory device in terms of automata theory, although it is not clear at this point whether such a description will have any advantages. We will assume in our model that the inputs X_5 come from the controller and the outputs X_6 go to the controller. As before this is not restrictive, since the transfer functions through C, to and from M, may be unity.

We believe that it should now be possible to find a definition, functional with respect to this model, which will reliably distinguish between a system which is learning and a system that is still adapting. The concept of considering a machine as a synchronous machine with its behavior measured only at discrete intervals of time gives us a means of defining what we mean by improvement with time of a learning system as differentiated from the improvement of adaptive system in the time which it requires to adapt. In order that a system may be considered learning it must first of all be successfully adaptive, and we have required that to be successfully adaptive the system must achieve steady-state behavior within some specified length of time. This specified length of time in turn becomes the minimum distinguishable interval of time with respect to which we judge learning behavior. In other words, <u>on the time scale with respect to which we judge learning, adaptation requires one single unit of time, whereas learning is taken to involve improvement over more than one of these minimum distinguishable units of time</u>.

This redefinition of time does not completely solve our problem since we feel that for a learning system to be significantly different from an adaptive system, it should have the ability, because of its memory, to retain its adaptation to more than one situation. The above concept of improved performance with respect to more than one discrete time interval would still allow the stepwise adaptive system. For example, suppose we have a situation where the adaptive process involves a trial-and-error or hill-climbing search for an optimum condition, and the nature of the process under control is such that we can allow the control system to hunt around for only some specified length of time. At the conclusion of this length of time, we require the control system to lock in on the best solution it has found up to that period of time, and then allow the process to go on to its completion. If, after some period of inactivity, the identical situation occurs again, we can see that the controller might start hunting again from the same place it was before and come to a better performance this second time around. But, unless the occurrence of these two applications of the specified input was separated by some significantly different input quantity, then we would not wish to consider this as learning.

What is involved here is a distinction between a significantly different input and simply the absence of any input at all. In other words, in some cases zero input may be something significant; in other cases it merely means that we have essentially turned the system off, or the system has remained quiescent until something else happens. In such cases, where there is a meaningful sense to the idea of the system simply being quiescent, we might then have the situation described, where there would be a recurrence of a particular input and, with respect to our discrete time intervals, we would apparently have some sort of learning behavior. This type of improvement we do not wish to include as learning. What we need is some means of specifying a <u>sequence</u> of <u>different inputs</u> to the device and measuring the performance with respect to this sequence of <u>different</u> inputs. We need something precisely analogous to the concept of an experimental sequence of inputs as used in automata theory. Unfortunately, we have a difficult problem in the practical case of defining just what is meant by a <u>different</u> input. It is in order to find a way around this problem that we have suggested the refinement of the model to the extent of separating those parts of the system with long response time into a third block.

For this purpose we now suggest the following test procedure with reference to the model shown in Fig. 5. First of all, we open the path $\overline{X}_5$ between the controller and the memory section so that nothing that happens to the controller will have any effect upon the memory section. Now we apply to the system an alternate sequence of at least two different inputs, say $\overline{X}_{1a}$ and $\overline{X}_{1b}$. We will judge whether or not these are different on the basis of the response of the system. If the response of the system to the two inputs is significantly different, in the terms of our performance criteria, then we will assume that the inputs are significantly different. We shall observe the alternate responses, and if these responses are satisfactory as defined before and if, further, they do not change over succeeding applications of the alternate inputs, then we will say that the system is successfully adaptive. (This is, of course, assuming that it fulfills the criterion of the adaptive definition made above.) If we find that the responses improve over successive applications, then we must assume that either (1) we have the case just discussed of a stepwise adaptive system and that the responses are not actually significantly different, or else (2) we have actually failed to remove from the controller all items which have response time longer than <u>T</u>.

Now then, if the system has passed the test so far, i.e., the alternate responses have been successful but have not improved with time, we then reconnect the memory section and repeat the procedure. If the alternate responses now show a progressive change with repeated applications of the alternating sequence, then we will say that the device has a learning capacity. Further, if the change is in the nature of an improvement in any significant sense, then we will say that it is a <u>successful</u> learning system. It should be emphasized that this is in the nature of a preliminary suggestion as to a definition. We have not yet been working on this particular approach for very long, and we are far from fully satisfied that we have taken into account all the factors.

The definition presented above represents an attempt to define a legitimate Gedanken experiment with respect to the specified model and is, in that sense, not yet complete. However, if we relax our restrictions, we can still use the conceptual model of the system with synchronous time to make a verbal definition, somewhat more intuitive in nature, that appears satisfactory at this time. As before, we shall define the discrete time interval <u>T</u> in terms of the longest time permitted for the system to reach satisfactory response to a particular input condition. With respect to learning, we measure time only in terms of these discrete intervals. We may then define a learning system as <u>an adaptive system in which, during any discrete time interval, the response of the system is determined, not only by the input presented and the state of the system at the beginning of this time interval, but also by information stored as to the response of the system during previous time intervals</u>.

It should be noted here that this last definition does not require the representation of the system in the form shown in Fig. 5. It requires only the concept of the synchronous time interval. The difference between the two definitions is that the test definition represents an attempt to indicate <u>how</u> one might determine that the response of the system <u>is</u> dependent upon the information stored about the system's behavior in previous time intervals. This is a reasonably clear intuitive concept, and, in that sense, the second definition offered may be considered complete. But in the case of complex systems this reasonably clear intuitive concept may sometimes become considerably less than clear. Therefore we do feel that an important task for the future is to continue work upon the test (Gedanken) definition in order to provide a more firm statement of the means for judging when the system is utilizing information stored from previous time intervals.

SECTION 1.3 STRUCTURAL ELEMENTS OF LEARNING CONTROL SYSTEMS

It will be recalled that at one point in the study we suggested that it might be possible to make a classsification of learning systems according to their structural organization. There is no doubt that some day this will be done, but we feel now that it would be premature to attempt it at this time. Since nobody has yet reported on a practical learning control system, it is obvious that we could scarcely make a realistic classification of systems by structural organization. This will be a reasonable thing to do at such time as a number of learning control systems have been built and certain types have been found to be applicable to certain situations. This of course has been precisely the situation in adaptive control. It was not until a considerable number of adaptive control systems had actually been built, or at least designed, that anybody could make a very serious effort in classifying the various types of adaptive control. This will probably be the case with learning systems.

At this point, it would seem that our best bet is to comment instead on some of the structural elements that will apparently be required in learning control systems. In the next section, under the heading of Environmental Characteristics, we will make some suggestions as to what sort of organizations might be suitable for certain kinds of problems, but until we have specific problems to work on and to test, we can do no more than make rather tentative and sketchy suggestions along this line.

The most important element in a learning system will generally be the memory. Indeed, the presence of memory will be the basic distinguishing feature of a learning system. By memory we mean a device having the characteristics specified in a definition presented in an early progress report. This definition is repeated at this time.

Definition: A memory device, a device for the storage of information, should have the following characteristics:

(1) It should be capable of taking on some number n distinct stable states.

(a) Each of the n states is defined by a unique combination of values of some m distinct, independent, physical quantities.

(b) The m quantities may be distinct in space or time.

(c) The word stable, in this context is taken to mean that the device, once placed in one of the n states, can be maintained in that state, as reliably distinguishable from any other state, for a finite period of time. There is no logical restriction as to a maximum or minimum time for which it should be possible to maintain a particular state. From a practical point of view, a device will have little value for storage unless a state can be maintained for a period of time which is long relative to the processing time of the the system of which the memory device is a part.

(d) The device may be volatile or non-volatile, i.e., the maintenance of a state may or may not require power.

(e) If the numbers of distinguishable values that may be taken on by the m quantities are denoted by

$$a_1, a_2, a_3, \ldots\ldots a_m$$

then

$$n = a_1 \times a_2 \times \ldots\ldots a_m$$

(2) It should be possible to determine in which of the n states the memory device is, at any desired time, by measuring the values of the m physical quantities.

(a) The state of the memory may or may not be changed by the measurement process (destructive or non-destructive read-out).

(b) It is generally desirable that it be possible to partially determine the state of the device by measuring some specified subset of the m physical quantities, e.g., reading one word of a memory.

(3) It should be possible to change the state of the device at will, from any one of the n states to any other state, by changing the values of any one or all of the m physical quantities.

(a) It should be noted that this requirement implies that the method of changing state must not involve any damage to the device. If the device were to be damaged, such that it no longer had the specified characteristics, then it would no longer be a memory device.

(b) For the device to be practical, it should be possible to change its state an indefinitely large number of times.

(4) The information stored in a memory device shall be specified by its state, in accordance with some previously specified code.

After a period of two years this definition still appears to be satisfactory as a list of the characteristics which a practical memory device must have. We can, however, make a few additional observations as to the particular characteristics that a memory involved in a learning control system should have. First of all, we would expect that it should be a nonvolatile memory. If we are talking about a control system, we generally are talking about a system that is working in a practical workaday environment, i.e., not in a laboratory situation, and would thus be subject to power failures, accidental interruptions in service, and other unforseen circumstances. Therefore, it would seem quite desirable that the memory be nonvolatile. At the present state of our technology this would seem to restrict us primarily, if not entirely, to magnetic devices. We are not at this time aware of any practical nonvolatile memories which are nonmagnetic in nature. The cryogenic memory may in some respects be considered nonvolatile, but the rather special environmental requirements of the cryogenic devices would seem to indicate that they would not be useful in control system situations.

The question as to which form of magnetic memory would be most desirable is of course not one that can be answered for the general case. This will depend upon the characteristics of the particular control situation. It is not hard to envisage situations where any one of the four basic types of magnetic memory -- core, disk, drum, or tape -- might have their suitable applications. In a control situation of any complexity it is quite likely that a combination of these forms of storage would be valuable, just as in present-day digital computers. There is no doubt that the associative form of memory is attractive with respect to its logical principles. However, at this time this device is still primarily a laboratory curiosity. Further development will be required before the device can be properly evaluated, but it does seem quite possible that the device would have more significant applications in the area of control than in the more conventional arithmetic species of computer. This is indicated by the concept of a control computer as being required to associate a particular response with a particular environmental situation.

This concept of a learning system recognizing a previously encountered situation and associating some previous response with it raises the possibility of some form of pattern recognition device being incorporated in a learning control system. However, in order to gain a perspective on this question, we should consider the fact that the problem commonly referred to as pattern recognition actually breaks down into two distinct problems. The first is the problem of actually abstracting a situation from the physical environment in terms of certain useful measurements, i.e., the instrumentation problem. In general, we assume that the instrumentation is something that is selected ahead of time by the designer and is not in general subject to alteration by the machine. The second problem, that of classification, is a completely separate one. We recognize, of course, that the very process of measuring a situation in a sense classifies it, in that the measurement accuracy of our transducing devices segregates the situations encountered into classes. For example, if our voltmeter has an accuracy of 1 volt then situations involving voltages between 4 1/2 and 5 1/2-volts are segregated into the 5 volt classification. This is a crude example, but the idea is clear.

In pattern recognition systems, there is usually a further implication of a logical reduction of the number of input classes from that number of inputs which may be distinguished by our instruments to some lesser number. For example, if we have a situation that is measured by four variables, any one of which may take on, let us say, one of ten values, then we have a possibility of 10,000 differentiable input situations, assuming that there are no restrictions on any particular combinations of values. The classification problem is to classify these 10,000 separate situations into some smaller number of categories in whatever manner may be appropriate.

Thus, pattern recognition involves two distinct problems -- instrumentation and classification. The classification problem in this sense is simply a particular type of logical problem and does not necessarily imply any special apparatus, differentiated in any significant manner from any other type of logical capability that might be provided within the system. Indeed, there is the possibility that in some cases no classification will be necessary or desirable. It may be that we cannot group various types of inputs in any significant manner, so that it may be necessary for the device to list exhaustively in memory all the input situations that it has encountered in its experience. Then, when such a situation is encountered again, the problem is not one of classification so much as one of simple search. On the other hand, we know that classification is often involved in search where there is a large number of items in order to make the procedure more efficient. In summary, it would seem that in systems of any degree of complexity some sort of classification procedure is going to be involved somewhere in the system. This would seem to be a more accurate statement of the situation than to say that pattern recognition will necessarily be involved.

Next we might consider the problem of decision. It will be recalled in earlier discussion that Gibson has stated that decision is one of the characteristic abilities of the adaptive process. In this context he is concerned with (or has reference to) the ability to "decide" what to do about a situation once it has been "identified". We do not feel that decision in this sense is <u>necessary</u>, as indicated by our definition of an adaptive system. It may be present in more complex types of adaptive systems, but we do not think it is <u>necessary</u>, since we include among adaptive systems those which achieve adaptive behavior through the use of non-linearities deliberately inserted by the designer. In this case the idea of decision does not seem to have any particular meaning.

It would also appear that in the case of the learning system, the ability to make logical decisions is not a <u>necessary</u> characteristic. Certainly one method of implementing a learning system would simply be to provide the ability to associate with any given input a certain response. Then the "decision" procedure actually amounts to a search in memory for the response associated with a particular situation. Of course, this implies that the solution to any given particular situation is already in memory. In the situation where the machine is still learning, or where it is still encountering situations it has not seen before, it would appear that the decision process is more explicitly involved. This decision function that we are concerned with here is one type of logical operation. Thus, we again find the requirement for some degree of logical facility, but not necessarily of a basically different nature from that required if classification is involved in the operation of the system.

The third characteristic of the adaptive system as listed by Gibson was modification. Again, as we have indicated, we do not feel that this is necessarily a requirement in adaptive systems, since the deliberate use of nonlinearities may be used to provide for modification. You may, if you wish, look upon the change of transfer function as a result of nonlinearity as a form of modification, but this is not the explicit change of parameters implied by Gibson. However, it does appear that in the learning system modification in the sense intended by Gibson will, in general, be involved.

In our concept of a learning system, the response of a system during any time interval is dependent not only upon the current inputs and the state of the system at the beginning of the interval, but also on information concerning the behavior of the system during previous time intervals. This means that the output of the memory device is going to have some influence on the response of the system. In general, this output will be an information signal, typically perhaps a set of binary signals on some number of output lines from the memory. We might expect that for such information signals to have any influence on the behavior of the system, they would have to be translated into some form more suitable for causing changes in the active parts of the system, although such translation may not always be required. It is conceivable that these output signals from the memory might, in their original form, simply function as additional inputs in the system and cause the response to change in some manner. This would not correspond quite so directly to the idea of modification as would be the case, for example, if the memory signals were translated into a control signal causing a motor to change a potentiometer setting. But we believe that modification in the form of a distinct change in the arrangement of the system due to the action of signals from memory will usually be present in learning systems. It will often take a form similar to the control section of a digital computer, although other forms are, of course, conceivable. Experience will indicate which are most useful.

Next, we have the question of the measurement of goal in the learning system. We have pointed out that in any learning system, as indeed in any adaptive system, there is assumed to be a purpose, something that we wish to happen. A learning system cannot be defined without the concept of goal, since the very idea of improvement implies some sort of a reference standard. It is not necessary in all cases that the learning system explicitly measure its own performance. In other words, open loop learning is apparently possible and in a recent progress report we gave an example of this. However, we do agree with Gibson that the most significant adaptive and learning devices will be closed loop in function. That is, they will be checking on their own performance.

This problem of goal evaluation is not a problem distinct from any of the others we have already considered. It first of all involves measurement of the performance of the system. This simply means an enlargement of the measurement apparatus that we have already specified must be associated with the identification function. In addition to the various factors identifying the input situation and the environmental situation, we must also measure the factors which are connected with the response of the system. In addition, there may be a decision function involved. In order to evaluate whether we are progressing towards more satisfactory solutions in a learning situation, we will have to compare our progress at one point with our progress at some other point and decide which is better. This in turn involves some sort of logical operation. So therefore, with respect to any apparatus involved, the goal problem is just a particular aspect of the identification problem, and will require measurement ability and some degree of logical ability.

In summary then, we see that the typical learning system will involve three basic types of elements which differ from those found in the normal or ordinary feedback control system. Of these three, one, memory, would not be found in an adaptive system. Memory, as we have pointed out, is the characteristic element of the learning system that will most generally differentiate it from an adaptive system. The second element is some degree of logical ability. This may also be found in ordinary adaptive systems, but it is likely that learning systems would require a higher level of logical ability. Third, we must have some means of translating information signals into modification signals to the control plant. Again, this sort of function might be found in an adaptive system, but the nature of this function will be more complex in the learning system.

When we look at the nature of these three functional elements, memory, logical ability, and the ability to translate information signals into control signals, we recognize that we are basically talking about the sorts of things that are available in the normal digital computer. We certainly have memory in the normal digital computer. We certainly have logical ability in a normal digital computer, and we have the control function. The concept of translating an information signal into a control signal corresponds exactly to the function of the control section of a digital computer. This section takes instructions from memory in the form of information signals, usually binary numbers, and translates these into signals of the type that are necessary to cause the computer to do various things. Therefore, we may expect that the typical learning system will incorporate a digital computer.

Now this is not a terribly remarkable conclusion to be sure, but it is of interest to note that the need for the digital computer arises in a direct manner from the close correspondence between the basic functions which must be performed in the learning control system and those which a digital computer is designed to carry out. This does not mean that it would be impossible to build a learning control system without specifically including a digital computer or some part of a digital computer. It does make it seem quite likely that in general we may expect to find some sort of digital computer of the conventional type as a part of learning control systems.

SECTION 1.4 ENVIRONMENTAL CHARACTERISTICS OF LEARNING SITUATIONS

In this portion of the report we wish to discuss the problem of how we, as designers, can recognize the situation in which a learning control system may have some value. It is to be expected that, as the science of learning systems develops, more and more criteria for their use will be developed. It appears that at this time the most useful way of looking at a problem is to ask the question of how much is there about the problem that we don't know. This concept arises from the point of view that the more we don't know about a situation that we wish to control, the more complex the system will have to be in order to be able to perform successfully.

To illustrate this by example we might first start with a very simple type of control. Let us suppose that we wish to control the speed of a dc motor, and we propose to do so by controlling the current to the field of the motor with a rheostat. If we knew exactly the characteristics of the loads that were going to be on the shaft of the motor and if we knew precisely all the characteristics of the motor itself we could accomplish the desired result in an open loop fashion, i.e., we could calibrate the settings of the rheostat in terms of the corresponding speed. This, in point of fact, is often done.

However, if we do not know in advance what kind of loads are going to be on the shaft, and if we do not know in advance how the characteristics of the motor may change with temperature or time, then we find open loop control is inadequate. If the load differs radically, then a potentiometer setting that in one situation means 500 rpm may mean 1000 rpm in another. So we have a certain degree of uncertainty. There is something about the conditions under which the system is going to operate that we don't know, so we go to a closed loop control system. We measure the speed of the motor and compare this with the speed that we want, and if there is an error we institute corrective action. This is the basic idea of feedback control. Now there is nothing new about this, but the point is to recognize that it is because there is something we don't know about the operating conditions of the system that we are required to use feedback control. If we have complete information about the system ahead of time, if we can predict exactly what the characteristics and the operating conditions of the system will be at every instant of time, then there is no need for feedback control.

When we come to consider the necessity for an adaptive system we find the same concept applies. Just as the need to go from an open-loop to a closed-loop control system is indicated by a certain lack of knowledge about some operating characteristics or environmental situations of the system, so the need to go to an adaptive system is generally indicated by an even greater uncertainty about these factors. For example, in our simple speed control system there is a limit to what range of operation we can get with simple feedback control. If we know in advance that the system must function satisfactorily over speed ranges of 500 to 2,000 rpm that is one situation. On the other hand, suppose we know less about what is likely to happen, in the sense that a far greater range of speeds are possible although not necessarily certain to be encountered. Suppose we want to operate satisfactorily from 1 rpm to 10,000 rpm. Then we have a control problem of an entirely different magnitude.

Or, if we know far less about the outputs that the system will be subjected to, we again have a different magnitude of control problem. If the load torques for the system may vary from 5 foot-pounds to 25 foot-pounds, that is one situation. If they are going to vary from one foot pound to 1,000 foot-pounds, we have quite another situation. Or, perhaps we know less about the environmental conditions. If we know that the temperature of the motor will never vary outside the range of 20^{o} to $40^{o}C$, then we have one design problem. If, on the other hand, we don't know whether the temperature is going to be minus 20^{o} or plus 120^{o}, then we have a higher degree of uncertainty and an entirely more complex and difficult control problem.

It is in these types of situations that we often find ourselves forced to go to the adaptive system. Thus we have the concept that the higher the degree of uncertainty about the situation, the more complex the system must be in order to respond satisfactorily under all the possible conditions that may occur. It would seem quite reasonable that this same idea can be extended still further to indicate that a learning system,being a still more complex device,will in turn be indicated by an even greater degree of uncertainty about the conditions under which the system will operate. Not only may we classify the system according to the degree of uncertainty about its operation but we may also, as Bellman[5] has suggested, classify it according to what it is that we are uncertain about. Thus if we don't know what the input range is going to be, this might indicate one type of system. If we don't know what the incidental environmental conditions are going to be, this might indicate a different type of system. If we don't know what our goal is going to be, this might indicate yet another type of system, etc. It is this type of approach which we wish to discuss here in attempting to classify systems, and to suggest something about the organization that would be appropriate, according to the degree and kind of uncertainty about operating conditions.

We may start by noting that the unknown factors about the system might be divided into three areas: (1) input unknown, (2) system unknown, (3) goal unknown. In most cases we would probably have some combination of uncertainty about all of these factors. For purposes of analysis it is convenient to start by considering only one of the three to be unknown at a time. This will require that we specify what is meant by the input being totally known, or the system being totally known, or the goal being totally known. For this purpose we must make a few refinements in our model.

With respect to the input, it will prove convenient to consider as separate those inputs which are involved in $\overline{W}$, and those which are not. Therefore we will refine the model slightly as shown in Figure 6, with the components of $\overline{X}_1$, separated into two categories, $\overline{X}_{1w}$ and $\overline{X}_{1e}$. $\overline{X}_{1w}$ includes those components of the input necessary to specify what is meant by satisfactory response, and $\overline{X}_{1E}$ includes the remaining components, which will often be environmental inputs which influence the behavior of the system, but do not occur in the goal structure per se.

We must make this same distinction among the initial values of $\overline{X}_2$. It will be recalled from the first section that the initial values of some components of $\overline{X}_2$ may be involved in our criterion of acceptable behavior. These will be indicated as $\overline{X}_{2W}(o)$.

The idea of a totally known goal will also require a bit of refinement in our concepts. In this analysis we will say that by a totally known goal we mean a situation wherein it is known precisely in advance what the

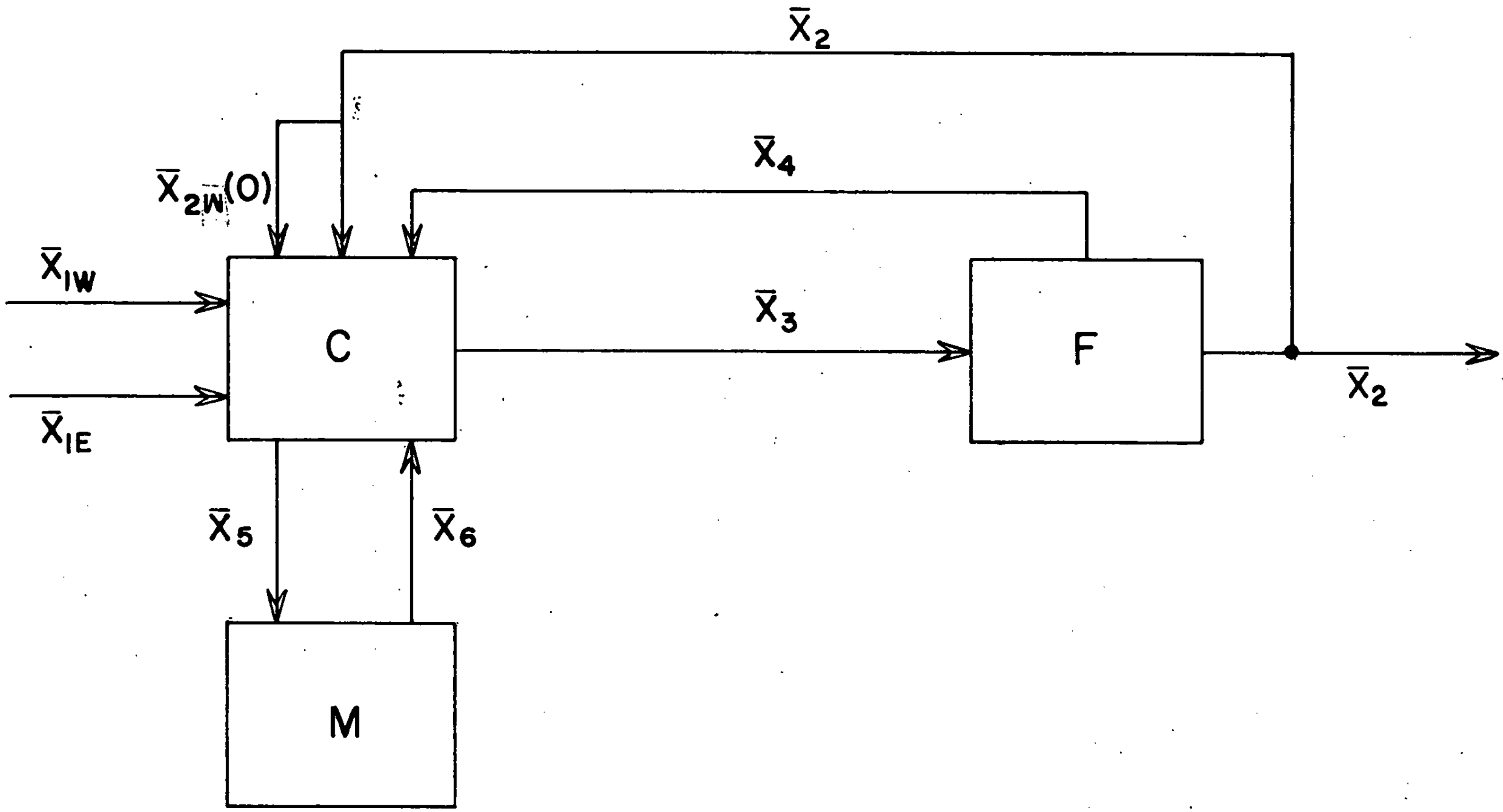


FIG. 6. MODIFIED LEARNING SYSTEM MODEL

system output should be in response to any given particular input. But if we know exactly what the output should be, this implies that the output should not change, and this rules out improvement, since the output must change to show improvements.

The answer to this apparent contradiction can be found in the fact that for a learning system the goal structure must be multi-dimensional. Then we have the possibility that during a single cycle of operation (interval T), the goal will be absolute with respect to some sub-set of the output variables. There must also be defined, as a higher-level goal, some sort of a cost function dependent upon the remaining variables within the goal structure. With respect to these variables improvement will be possible, so that we can have learning in a meaningful sense. For example, in a process control situation, the goal may be absolute in the sense that the output product must have a definitely specified chemical content that permits absolutely no variation. So with respect to this specification, the goal is absolute and known. On the other hand, there is room for improvement in that, while this content of the output is absolutely fixed, the cost of producing this given output may very definitely be subject to improvement. We shall refer to the criteria against which the adaptive performance is measured within the intervals of length T as the <u>adaptive goal</u> and the criteria against which behavior is measured over the sequence of intervals as the <u>learning goal</u>. Only the <u>adaptive goal</u> may ever be absolutely known.

Case IA - Input Distribution Unknown.

Here we are concerned with one case where the system is totally known and the adaptive goals are known, but something is unknown about the input. We shall take this specification to mean the desired output for any given input is known, that the response required by the controller in order to produce this output is known for any possible input and furthermore, that the entire class of inputs has been totally and exhaustively catalogued, but the distribution of these inputs is unknown. That is, we do not know whether any one input will occur any more often than any other. Notice that we do not permit the possibility that some inputs are going to occur that we have no way of anticipating. This case of unexpected inputs will be found to be logically similar to the case of the system being unknown.

With this rather restricted environmental specification, there would be a possibility of learning if the machine functioned by matching the input with a stored catalog of the possible inputs, with each entry in the catalog associated with a particular desired response of the controller. If, in advance, we did not know anything about the distribution of the inputs, then we would initially have to use random search. But suppose that the distribution of the input is actually non-uniform, i.e., there are some inputs which occur more often than others. We could then provide a learning capability by giving the machine the capacity of reviewing the pattern of inputs encountered with experience and altering its search procedure in some manner so as to look first for those inputs which are most likely to occur. Thus the efficiency of the search procedure could be improved, and the system would exhibit faster and presumably cheaper response.

We may note here that in order for a system of this sort to have any practical value, i.e., for this to be the best way to do the job, we have some very severe requirements upon the characteristics of our system and environmental situation. First of all, there must be a relatively large number of possible responses, so that a search for a proper one would be a lengthy procedure relative to the time requirements of the process being controlled. It is important to note there must be a large number of possible responses, as well as a large number of possible inputs. The reason for this is that if, as has been specified, we know what every possible response is, and we furthermore know which response should be associated with every possible input, then we have an ex post facto classification of our inputs into a number of categories equal to the number of possible responses. This in itself does not help, since the problem for the machine is to find which response class the input belongs to. However, since our system is totally known, it is not random, and therefore the association of a particular input with a particular response will not be random. This means that the inputs associated with any given response will have certain factors in common which will make logical classification possible, thus greatly reducing the search time.

Another requirement is that there be a sufficient complexity of relationship between the inputs and the desired responses of the controller that it would be faster to search through the table for the proper response than to try to compute the response from some sort of mathematical relationships. Obviously, if the mathematical relationships are reasonably simple, then computation would probably be faster than search. Note that we said that these relationships must be extremely complex, because we have specified that this is a known system. We know what the relationships are, and it is only under the circumstances that they are simply so complicated that we cannot compute upon them in a reasonable length of time that a search procedure becomes preferable.

Furthermore, in order for there to be any point in building in the learning facility, i.e., the part of the machine which can carry out a statistical analysis on the experience of the machine to determine the distribution pattern inputs, it must for some reason be impossible to determine this distribution ahead of time. If there were some way that we could make measurements or in any way predict in advance what the distribution of inputs would be, then we should simply build the proper search procedure into our machine in the first place. Thus, there would be no need to supply this relatively complex statistical capability. Another possibility is that the distribution of the inputs changes in an unpredictable manner but at a rate which is slow compared to the time required for the machine to evaluate the distribution. For example, if the input distribution changed every hour in a more or less random fashion, but it took the machine only five minutes to find out what the new distribution was after a change, then we could achieve useful learning behavior. On the other hand, if the change in distribution was at the rate of once every five minutes and it took the machine five minutes to find out what had happened, then the machine could never catch up and no useful behavior would be observed. Thus, there would again be no reason for supplying this statistical analysis capability.

With all these restrictions considered, we can now make a general statement as to the form of a system of this type. The first thing we note is that it would be an open loop system. Since we know in advance what response we want for any given input, and we know how to get it, there is no need to measure the output to be sure that we are getting it. We will need a logical facility for classifying any given input prior to search and controlling the search. We will need a memory in which we will store the possible input classifications in conjunction with the proper controller responses.

We will need a logical facility for making statistical analyses of the experience of the machine in order to determine the input distribution. We will need to provide a means whereby the machine can change its search procedures in accordance with the results of the statistical analysis. This change might be accomplished either by rearranging the information in the memory, or by changing the search procedure, or any combination thereof which might seem appropriate in the particular case. The block diagram of such a system is shown in Fig. 7.

Case I-B - Inputs Incompletely Catalogued

Next we should consider the situation where we do not know in advance what all the possible inputs to the situation might be. Actually, this reduces logically, and in terms of machine structure, to the case where we simply do not have enough memory capacity to exhaustively catalog all possible inputs. Whatever the case, whether we don't know what all possible inputs are, or we can't catalog them all, the situation is the same as far as the machine goes. When this type of input arrives, the machine cannot find it in memory and will therefore be faced with the necessity of a trial and error search for the proper solution.

For this second case, then, in addition to the items discussed above, we must provide the machine with the facility for trial-and-error search over the operating regions of the machine for the proper response to the unexpected inputs. Also, it would be desirable to allow for the possibility that some of these uncatalogued inputs might be frequent in occurrence and should be listed in memory. We might handle this situation by providing some spare, or standby, memory capacity, so that when new inputs occur we can store them and keep track of their frequency. If they seem to be

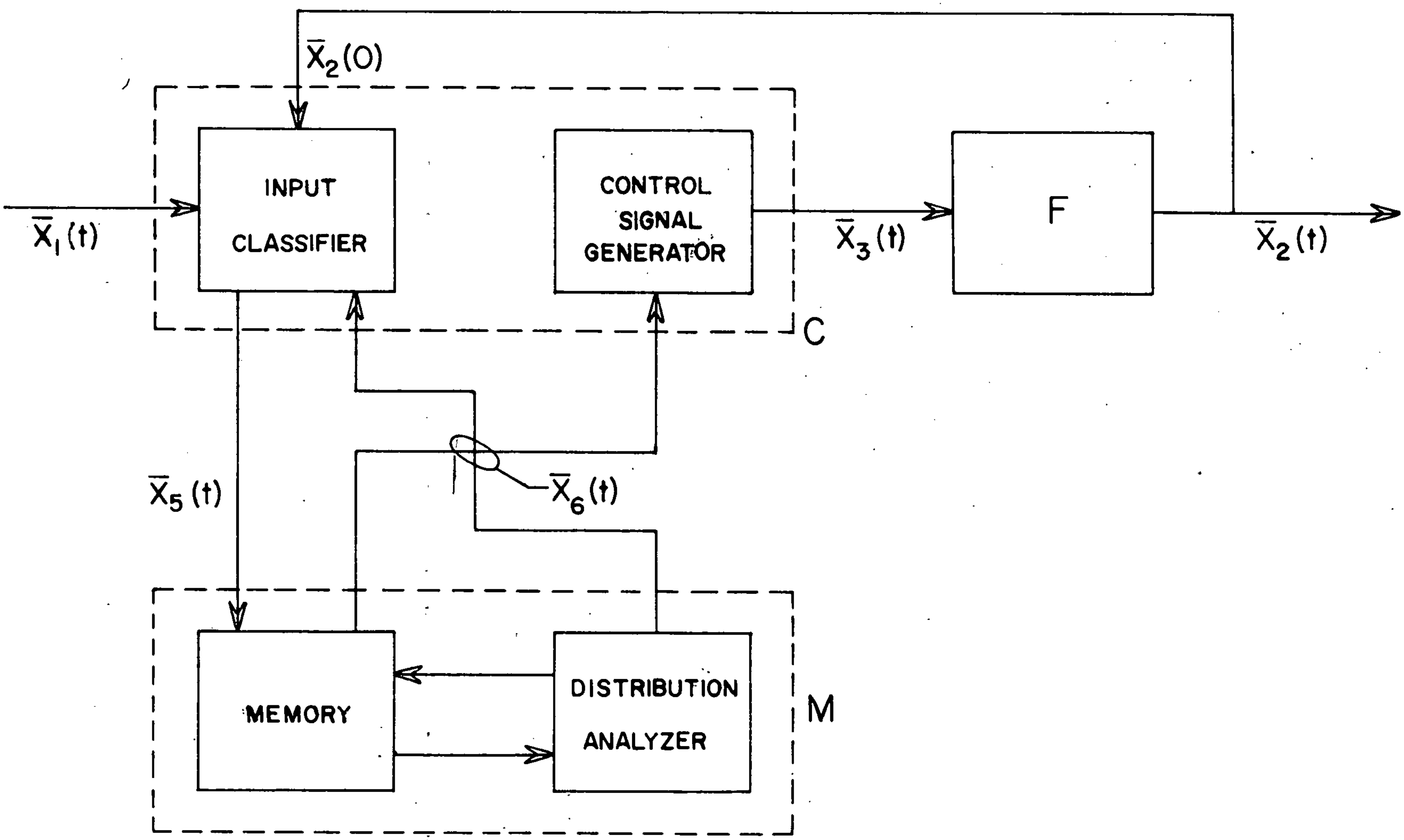


FIG. 7. TYPE I-A LEARNING CONTROL SYSTEM

significant, then they can be inserted into our regular list replacing some of the initial items which have turned out to be of lesser importance. The exact nature of the trial and error facility will be described in more detail later in the report. The block diagram of such a system is shown in Fig. 8.

Case II-A - System Partially Unknown

Now we come to the more interesting case where something is unknown about the system. The first situation we take up is the case where the equations of the system are not known, or if they are known they are so complicated as to make a straight forward mathematical analysis from them impossible in the length of time the system has to operate. These two situations are essentially equivalent in terms of the type of system that is appropriate. The essential fact is that we can't compute the answer; the reason why we can't compute it is not important. We further specify that this is the only thing that is not known. The inputs are known completely. We know what the total class of possible inputs is, including all initial conditions. We know the distribution of these inputs, and furthermore we know what system output should be associated with any input that may occur.

The thing we do not know is what signal or response the controller should have in order to cause the fixed plant to achieve the desired results. In this situation the only possible method is trial-and-error, since we have ruled out the possibility of computing the solution by our statement of the problem. The machine then must have the capability, given a certain input, to try various solutions to the problem, to measure the output of the system, and to compare this output with that which is desired. Note again that we know what output we want, we just don't know how to get it. When a solution is found, it is stored in memory, together with the input with which it is

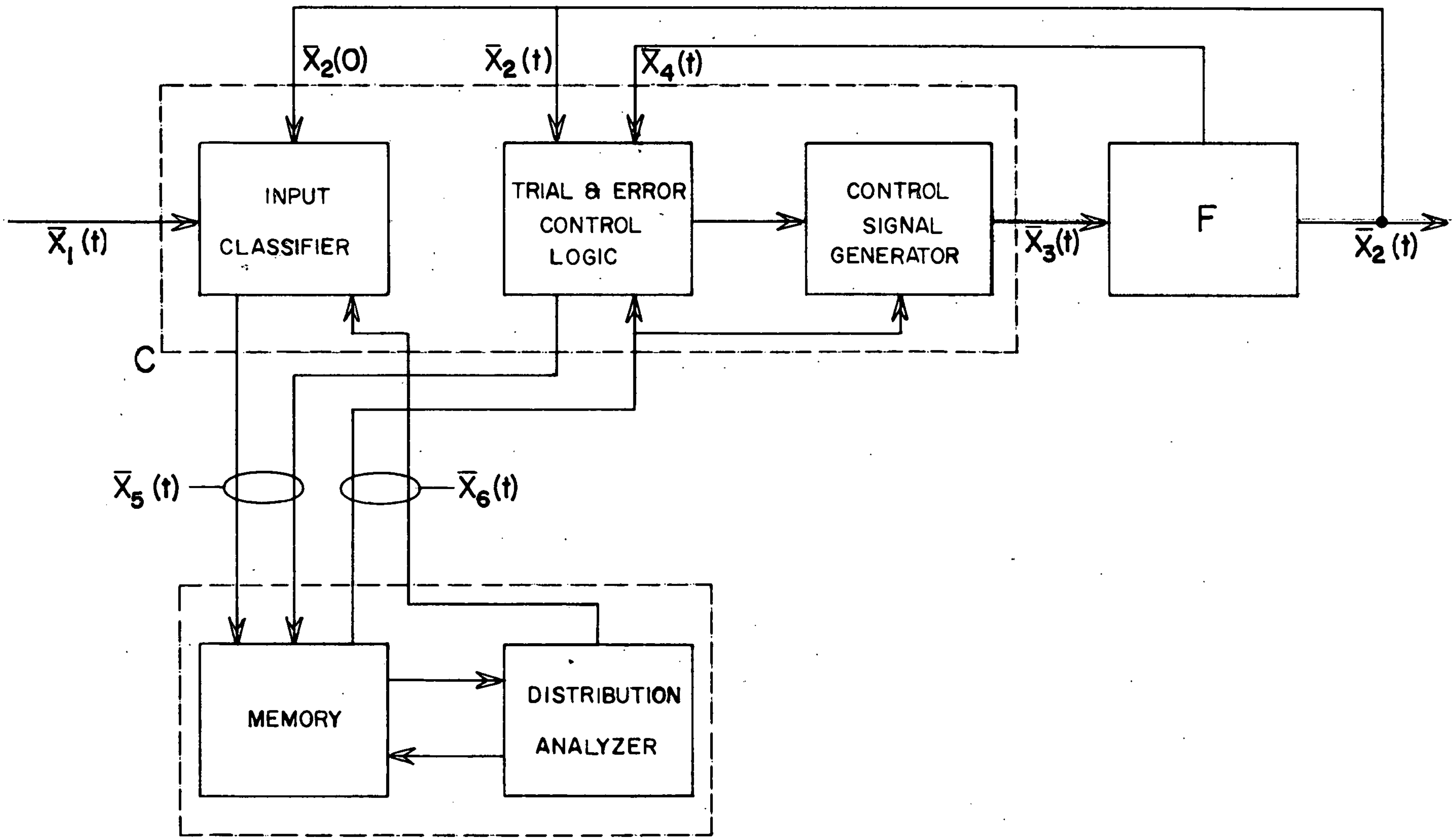


FIG. 8. TYPE I-B LEARNING CONTROL SYSTEM

associated. As the machine continues to operate, its first step when encountering a particular input is to search its memory to try to find a stored solution. In those cases where it has not seen the particular input before, it must go back to the trial-and-error method.

We will consider this to be the first level of learning in the situation where the system is unknown but everything else is known. Higher level learning might come about in the following manner. Initially, since we knew the inputs and knew everything about their distribution, we would have set up a search procedure based upon the probability distribution and such knowledge of the system as is available. If we find from experience that the number of the possible control responses is smaller than anticipated, then we might be able to improve the search efficiency by a new classification or search procedure based on this new knowledge. This would require again some method of statistical analysis, and it would also require keeping records as to how many different responses there were.

On an even higher level it is possible that as we accumulated data on the proper response to various inputs it might be possible to identify the causal relationships and actually develop a method of computing the response, rather than having to search for it. This is on the assumption that the system was not known at the start, rather than known but too complex for calculation. This third level will obviously require complex logical capabilities.

This method of storing solutions and later using them, either directly or for further computation, is based on the assumption that the system itself is not changing. That is, once a suitable response has been found, that response will always be suitable. This invariance of the system is actually implied by the specification that <u>all</u> inputs are completely known, since we assume that any change in the system is always caused by something. We must separate the system from the influences acting upon it. If we have a system and we find that it changes in a manner which does not seem to have any relationship to any external influences on the system of which we are aware, then we must assume that there are external influences of which we are <u>not</u> aware, since something has to cause these changes in the behavior in the system. In other words, we regard the system as a fixed thing, a device whose behavior can in theory be predicted if all the forces acting upon it are known. The situation where we find that the system is changing in some unpredictable manner with time actually reduces to a situation where we do not know all the inputs. This is the case we wish to consider next. It may seem that this should be a Class I situation since we have unknown inputs. However, the effect of these unknowns is seen as a change in the relationships between the known inputs and outputs, i.e., as an apparent change in the system parameters. Also, the methods of handling this situation are more like those of Case II-A than Case I-A or I-B.

Case II-B - System Apparently Changing

In this situation some of the $\bar{X}_{1e}$ inputs are unknown. There is a significant difference between this situation and Case I-B. In that case (I-B) we assume that all components of $\bar{X}_{1e}$ have been identified in the sense that they have been taken into account in the equations of the system. But we do not know, or cannot catalog, all the possible values which they may take on. In Case II-B we assume that there are components of $\bar{X}_{1E}$ that we have not taken into account, either because we are unaware of their influence on the system, or because we cannot measure them.

However, the $\bar{X}_{1w}$ inputs, the inputs which are involved in the performance criteria, are totally known, i.e,, they are totally catalogued and their distributions are known, and the desired response of the system is known for each one of these inputs. In this situation some of the procedures used above would still be applicable. The system would initially have to function on a search basis and would record any solutions it had found corresponding to various input situations. But there is a complication in that we can no longer assume the time invariance of the system. The system is being influenced by factors which are not known and which may be causing the system to change with time. This factor makes it necessary to continually check on the system to be sure that it is still performing properly. In Case II-A, we assumed the system was totally known and time invariant, so that once the proper response for a given input had been found (and stored in association with that input pattern) the system could operate on an open loop basis. If the system does not change, any solution will continue to be a solution, so that further checks on the performance will not be necessary.

In Case II-B, the possibility of change in the system behavior will make continued rechecking necessary. In terms of addtional equipment, the main difference between this case and Case II-A will be additional memory. In Case II-A, the memory will initially contain the proper output in association with each input class. When the proper solution for a particular input is found, it can replace the output information, which will no longer be needed. For Case II-B, the output information must be retained for future checks, so extra memory space must be available for storing the solutions. In operation, the system, on its first encounter with a particular input pattern will operate just as in the previous case. It will search for a proper solution and it will store this solution in conjunction with the input pattern in question. On recurrence of this input pattern, it will, as before, use the stored solution of this input, but it will check the results to see that the system is still performing as it did before. If it is not, it will then have to enter again into a search procedure to find the proper response corresponding to the new situation. We may assume that the previous solution will be the starting point for the new search. If our description of the system is at all reasonable, there should be some correlation between the old and new responses. It should be noted that if the system is changing quite rapidly the provision of this learning facility may not actually result in improved performance since the "experience" of the system may not in such a case be of value.

For this case (II-B) there are several higher levels of learning which might be useful. First, we might provide for observation of the long-time patterns of behavior in order to deduce the probability distributions or patterns of the unknown variables. For example, if we observed a cyclic variation of certain system characteristics in time, we could take this into account in our control mechanism and provide for improved behavior. Another possibility is that long term observation of the performance of the system might provide a means of deducing the relationships between the measurable quantities and certain unknown state variables. For example, in certain chemical process situations there are devices which can be completely described by equations, but the variables needed in solving these equations are inaccessible for measurements. It is possible that in some cases long-term observation of the behavior of the system might provide clues as to computational methods which would enable us to deduce these unknown state variables.

This in turn would enable us to do more computing about the behavior of the system, with a resultant reduction or even elimination of the search procedure.

The case of unknown initial conditions is particularly interesting in that this may be considered, in some cases, similar to the case of hereditary systems. A hereditary system is defined as one in which the behavior at any time is dependent not only upon the present state and inputs but also upon what has happened to the system in the past. In other words, behavior is dependent upon how the system got to where it is, as well as on where it is now. The simplest example of this is any kind of a system exhibiting hysteresis. We may regard a system with unknown initial conditions as similar. If we have a situation where some of the initial conditions which are important in specifying the behavior of the system are unknown, we might be able to predict them, or account for them, on the basis of knowledge of the system behavior on prior cycles of operation. Suppose, for example, that it had been observed that, whenever input situation A occurred, response X was appropriate if the previous input situation had been situation B, whereas response Y was appropriate if the previous input situation had been C. Then we could account for the situation by making the response to input A dependent not only upon that input but upon stored information as to the previous situation encountered.

We note that in this example we have two different responses appropriate to situations which appear to be identical on the basis of current information. Actually, the problem is that the system was not totally identified by the current information. There were variables unaccounted for, and we could effectively deduce what these variables were, in an operational sense, by knowledge of what had happened before. The hysteresis type of situation may be considered to be logically equivalent. For examples, consider a magnetic device with hysteresis. With the substance in one particular state of magnetization, we cannot predict, on the basis of any measurements we can make, where it will go upon the application of some new magnetizing force. This depends, as we know, on how it got to where it is now. However, we could regard this as a lack of information. It is quite probable that if there were some way that we could measure the molecular state of the magnetic domains or something of this sort, that we could make this prediction. It doesn't matter whether it would even be theoretically possible to acquire such information or not, the logical idea is the same. This point is interesting, not only because it provides an alternate way of looking at the hereditary system, but also because it indicates that a possibly successful way of handling the hereditary situation is through the use of systems memory.

Next, let us consider the structural arrangements of the systems designed to cope with the situations of unknown systems (Cases II-A and II-B). One device that is obviously required is a logical control mechanism capable of instituting an organized search for the proper response patterns. The exact nature of the search procedure to be used will depend upon the particular situation. However, we may assume that in general we will have something on the order of either hill-climbing, or successive approximations, or some variation of these techniques. The implementation of such procedures will require among other things a fairly complex logical capability. We will require also some short-term memory with sufficient capacity to keep track of whether one try was better than the previous try.

A capability for random action will be necessary, although we may suppose that in most cases the search will not be completely random. Even though we may not understand the system fully, we usually would be able to derive some clues from the input situation as to what types of solutions should be tried. However, as discussed in an earlier report, there is a finite amount of information that we can provide, and we may therefore expect that situations will occur in which there is no correspondence between the situation we are encountering and anything about which we have any knowledge. Therefore a random trial will be necessary. Again, as pointed out before, this does not mean necessarily a "coin-flipping" operation, but rather the ability to simply try a perfectly arbitrary method if there is no method indicated by any previous knowledge or experiences.

Next, we can see that we must always provide for output measurement in systems of this type. Any trial-and-error system by its very nature requires output measurement so that we can judge whether or not the method we applied is successful. As noted above, for the case where the inputs are totally known and the system is unknown but invariant, the output measuring device will not be needed after the system has learned to cope with all possible inputs. As a matter of fact, in _any_ situation where the problem is bounded and it can be meaningfully said that there is a solution that can be found or learned, then we may expect that certain parts of the learning apparatus will not be necessary after the device has gained a certain degree of experience. One exception to this would be cases where the system varies in time in an unpredictable fashion, but at a rate slow enough that the learning capability of the system can keep up with it. In such a case the learning capability in its entirety may be required for the entire life of the device.

As for all learning systems, there must be long-term memory for storage of the possible inputs together with the associated system output desired, and/or the proper controller response. As discussed above, Case II-B will in general require more memory, since information on the desired outputs must be retained until, and unless, enough information is acquired for the system to be regarded as time-invariant.

Input classification will be useful only if the number of possible controller responses is considerably less than the number of possible inputs. If there is a unique response pattern associated with each and every input, then it will not be possible to reduce the number of input classes, since there must then be a separate item in memory for each input and its associated unique response. On the other hand, if there are many inputs associated with any particular response, then all those inputs are automatically grouped together, as pointed out earlier. The search procedure will, therefore, be more efficient if the inputs are classified before search. Even if there is a unique output for each input, classification may be involved in the sense that efficient search involves a systematic reduction of the area in which a match is sought, but this is logically different from an actual reduction in the number of input classes.

The various types of higher-order learning systems discussed will obviously require greater logical complexity. For the case where we may wish to revise the input classification on the basis of observations of system responses, we will need the same devices we needed for the Type I-A system. We will need a system capable of making a statistical analysis of the response and of altering its own search and classification procedures. The higher level of operation beyond this, the type of device which may be able to deduce the equations of the system from the accumulated experience, we cannot say much about at this time. Very little is known about the type of logical operations that are involved in these extremely complex activities. All we can safely say is that this type of operation will require logical capability of extremely high order. It will also require more memory since we will have to keep track not only of what inputs have been associated with what outputs but also of the rate of occurrence of various types of inputs. The logical process of drawing conclusions and generalizations from them will also require intermediate storage.

The same comments apply to the type of system specified for the situation where there are unknown influences acting. We will have to keep track of not only what input patterns and behaviors occur but also their time sequence. This will require additional memory capacity as well as fairly high level computational and logical abilities. The case of the hereditary system or its logical equivalent, the system in which we don't know the initial conditions but can deduce them on the basis of what came before, will also require additional memory. We have to keep track not only of where the system is now, but also of where it has been on some number of previous steps. So we can make the fairly obvious statement that, the higher the level of operation we expect of the machine, the more logical capacity and memory capacity we will have to provide.

In terms of block diagram organization, the various Type II systems will be similar to the Type I-B system (Fig. 8). The main differences will be in what is going on inside the various blocks. The first level Type II-A system will be identical to Fig. 8, except that the distribution analyzer will be missing and the input classification block will be optional. The second level Type II-A system will be identical to Fig. 8. The difference between this system and the I-B system is that the distribution analyzer will be analyzing the response patterns for Case II-A, rather than the input patterns.

The third level Type II-A system will include all the blocks of Fig. 8 as well as a high-order logical facility in the M section.

The Type II-B system at the first, second, and third levels will be identical in organization to the corresponding levels of Type II-A system. Again the differences will be in what is going on in the various blocks. These activities have been discussed in detail above, so we need not consider them here.

Case III - Partially Unspecified Goal (Optimizing Systems)

Finally, we should consider the case where the adaptive goal is not totally specified. This may at first seem contradictory, since in the first section of this report we indicated that we were not interested in the system where the goal is unknown. By this, we meant that we are not interested in the situation where there is no possible way of knowing whether or not we are improving our performance. We do not mean by this restriction to rule out the optimizing system. For this type of system we do not know specifically what our ultimate goal is. We may know that the performance has to be above some level, but our ultimate goal is to do the best possible, and in many cases we don't know what the best possible output may be.

This case of the goal not being totally specified is actually a special case of the unknown system, because it is fairly obvious that, if we knew the system totally and knew the inputs, we could specify what the best possible performance was just as well as we could specify what should be done to get to this desired response. Thus we may expect the system organization will be similar to the organization in those cases where the system is not totally defined. The main difference is that there would be a continuing trial-and-error activity over successive intervals of the application of any particular input situation. In the case where the goal is specified as a definite known level of performance, we have trial-and-error during any one interval. We apply input A for the first time and the system hunts around until it finds the controller response that gives the desired output. The system stores the response and does not again go into the trial-and-error mode unless the system changes in some manner so that it no longer is achieving the desired output. However, in the case where we are looking for an optimum it is possible that the finding of the optimum may occur over many successive intervals of time.

In a system of any realistic degree of complexity, the finding and identification of an optimum mode of behavior will be a complex and lengthy procedure. In most practical cases it may be expected that there will be a finite length of time we may allow the system to hunt around for a solution before we finally have to say, "Well, this is the best we can do for now, so we will have to buy it." The next time the same input occurs, the system may continue the search, using the previously found solution as a starting point in an attempt to find something better. This process will be repeated on each successive occurrence of a particular input, until the optimum is reached.

The identification of the optimum is something of a problem in itself. We do not know in advance what the optimum is, since if we did we would have a totally specified goal. It is conceivable that we might have a situation where we would not in advance know what the best possible output was, but that we could absolutely identify it when it was found. It is not certain whether or not this can happen, but experience suggests that, in any except a trivial case, it is not actually in practice possible to identify, with absolute certainty, optimum behavior in a system that is not totally known.

Nevertheless, in any practical case we are going to have to decide, sooner or later, that we have done as well as can possibly be done. In general, this will require looking for a "steady-state" situation. If we have been searching for the optimum for many trials and for the last n trials we have not been able to improve, then we will have to assume that we have found the optimum, although there may be no way of proving it. What n should be will depend on the situation. Once the system has decided that it has achieved the optimum, it will discontinue the search mode and use the stored solution on future cycles.

In terms of organization, the Class III system will be identical to the Class II system except that it will require considerably more memory in order to keep track of its long term progress towards optimization.

It is quite obvious that in most practical situations we will have various combinations of the classes of environmental situations described in the preceding paragraphs. Whenever these combinations occur, all of the capabilities described in each one of them will be necessary in order to provide the optimum learning system. It is also apparent that in many cases certain of the operations involved in one situation or another are similar, and the equipment could be shared in some manner to make construction more efficient.

The form of analysis we have suggested in this section of the report is not the only one that can be taken, nor are the descriptions given the only ones that can be applied to any given class or group of systems. However, we believe this method of analysis, the classification of a system in accordance with what is unknown, will provide a useful method of analysis in any case.

We believe that virtually any system could be put into this framework, and that to do so would provide a useful method of getting started on the problem of finding out what sort of apparatus will be required and whether or not the situation is of the type where learning characteristics may be of real value.

In this latter connection, we must note that what we have indicated here is what kind of capability would be required on the part of the machine if it were to achieve learning behavior. Our analysis will not answer the question as to whether it is economically or practically desirable to try to provide this capability. In the first place, in order for a learning device to be useful, there must be enough information, being presented at a rapid enough rate, that the machine can in a useful sense learn by experience. Just what this amount and rate should be will be dependent upon the individual situation. For example, suppose that experiences of the type from which the system might learn are going to be presented at the rate of one every ten minutes, and it takes 10,000 such experiences for the system to make any useful generalizations. Then you are certainly going to have to question very strongly whether you should put the learning ability "on the line", or whether you should simply put a recorder on the system to keep a record of system experience which can later be processed by a machine or human beings in a more efficient manner.

It is necessary to consider whether or not certain of these jobs, even though they could be done by machines, could be done better or more efficiently by human beings. Consider the type of system in which you provide for the capability of making a statistical analysis of the experience in order to determine better search procedures. It may be that a machine could do it, but perhaps a human being could do it better or more cheaply. This would of course depend upon the individual circumstances. This same philosophy of skepticism should be applied at every stage of the design and the question to be asked is not, "Is it possible for a learning machine to do the job?" but rather, "Is there any benefit to be gained from having a learning machine do the job rather than a human being?" This is a type of philosophy that any designer should apply, and it should not be necessary to warn good designers to apply this philosophy. However, there seems to be a very strong psychological attraction to the idea of building learning devices just to see if you can do it. Therefore, it is more important in this situation to caution the prospective designer to always consider the more conventional methods which might do the job.

SECTION 1.5 SELF-REPAIRING DEVICES

In this section we wish to consider briefly the relationships between the principles of machine learning and the problem of developing self-repairing devices. First we should comment on a fairly common misconception. Many people who are concerned with the problem of producing ultra-reliable systems will tell you that we need to duplicate the ability of human beings to learn from experience how to deal with new and unexpected situations. It should be recognized that there is a contradiction here. If a situation is truly new and unexpected, then our past experience will be of no value and our only possible method of approach is trial-and-error.

Consider a refinery control system. We build a large and complex system to control a refinery. We include all sorts of emergency procedures to deal with malfunctions, but the ultimate emergency procedure is to call in a human overseer. When the control system encounters a situation that is unlike any it has ever encountered or been told about, it sends for human aid. But why? Why should the human be able to do more than the machine? It is mainly because the human being has more information at his disposal than we yet know how to put into a machine. He has more experience than the machine, so it is more likely that he can correlate the present situation with something he has seen before which suggests a proper procedure. If the situation is as new to the human as it was to the machine, the human must use the same procedure that the machine would use, trial-and-error.

So we see that when it comes to "unexpected situations", it is not the learning ability of the machine that is important, it is the amount of information it has available. If the machine can find a solution to the problem by trial-and-error, it can store it, so that if the situation occurs again, it will not be "unexpected", but this does not help us with the unique situation. Learning by its very nature involves repetitive situations, not "unexpected" situations. However, there is still the possibility that some of the capabilities or components associated with learning systems might prove of value in self-repairing systems, and it is this approach to the problem that we shall explore in the following.

It is convenient to break the problem of self-repair down into three fairly distinct problems: recognition, diagnosis, and correction. Recognition is the problem of recognizing that there is something wrong with the system. Diagnosis is the problem of locating the source of the trouble. Correction is the problem of actually eliminating the trouble. We shall discuss these problems in relation to the classes of environmental situations discussed in the previous section.

Recognition is a more difficult problem than it may at first appear to be, particularly for adaptive systems. Generally we consider that there is something wrong with a system when it fails to do what it is supposed to do. But in an adaptive system we may not know in advance what the system is supposed to do, so how do we know if there is something wrong? For Case I-A, the situation is fairly simple. We know all possible inputs, so there can be no unexpected situations. We know what the system is supposed to do for any input, so all we need to do is to provide a means of measuring performance and comparing it with that desired. That is, we must close the loop. This may require a considerable amount of equipment beyond that found in the normal Class I-A system, but the logical problem is simple.

For Cases I-B, II-A, and II-B, or any combination of them, the situation is similar. Even though more and more factors may be unknown, we still know what the adaptive goal is, so we can always check the performance against this goal. Note that we specify the adaptive goal. As discussed earlier, the learning goal is not definitely known. The detection of faulty behavior on the learning level will therefore be much more difficult, though not necessarily impossible. Fortunately, malfunctions at the learning level will not in general be as serious as at the adaptive level. The adaptive goal usually represents some level of behavior which the system must achieve to be considered successful in any sense. Learning behavior usually results in a lower cost for this successful behavior. This lower cost is desirable, but not often essential.

Case III presents the most difficult situation, for here we do not know what the system should be doing. If there is some minimum level of acceptable performance, in addition to the optimum criterion, we can use this as a first test of proper operation. With respect to the optimum, while we may not know exactly what the optimum is, we know that the performance should not deteriorate significantly from one cycle to the next. So we could test for malfunction in the "optimum-seeking" behavior by comparing performance on one occurrence of a given input with performance on the previous occurrence. It should be noted that this feature of comparison with previous behavior is already included in a Class III system, so no extra equipment will be necessary. This procedure of comparison with previous behavior will provide a general check on learning behavior, but will require additional equipment in the Class I and II systems.

Diagnosis is probably the most difficult problem of the three. Recognition as discussed above determines only that the system is not performing as it should. The first, and most difficult, part of the diagnosis procedure is to determine whether the faulty behavior is due to failures in the system or to external causes. We shall consider this problem first.

For Case I-A there is no problem. Unknown external influences are by definition excluded, so if the system performance is faulty, the system must be at fault. For Case I-B the situation may be more complicated. Suppose we know the range of inputs over which the system can perform satisfactorily even though we have not catalogued all possible inputs in this range. Then improper behavior for any input in this range will indicate a system malfunction, since unknown influences are still excluded by our definition.

On the other hand, suppose we encounter an input totally unanticipated by the designer and outside the range of inputs for which the system was designed. Then improper behavior may simply indicate a signal outside the system's maximum range of adaptivity. The most obvious procedure in such a case would be to test the system on a signal which we know it should be able to handle. Of course, any time a system is successful in adapting to an input outside the range specified by the designer, the range is accordingly enlarged.

It should be noted that the restriction of totally known goal implies that, if any components of the input are involved in W, these must be totally catalogued. If they were not, the system would have no knowledge of its goal and could not possibly operate. Thus, a cataloguing of inputs could be incomplete only with respect to those components of the input not involved in the goal structure. For example, suppose the inputs to a system had six components, with only two involved in the goal definition. On receipt of a particular input, the system would check to see if the exact input were catalogued. If it were catalogued, the proper controller response would be known. If it were not, the system would check the listing of all values of the two components involved in the goal in order to determine that goal. It would then go into a trial-and-error search for the proper controller response to achieve this goal.

In Case II-A we have the situation that we know all possible inputs and the corresponding outputs but do not have a complete identification of the system. This implies that though we know what the system should do for each input, we do not know that it will be capable of proper behavior in every case, even if all components are functioning properly. In such a case the first step would be a check to determine if the system has ever adapted successfully to the current input. An actual search would not be required, since a search for a previous solution is the first step taken by a Class II-A system. All we need to do is have the system set an indicator according to the result of its search. If the situation has not been encountered before, one possibility would be to test the system with a situation which has been encountered before and is as similar as possible to the present situation.

Case II-B presents the most difficult case. If there are completely unknown forces acting in the system, then by definition we don't know what effects they will have on the system. So we cannot reliably distinguish these effects from those of internal malfunction. One possible technique is to try to isolate the system from these influences, but to the degree that they are unknown this will not be possible. The most powerful technique, and sometimes the simplest, is to replace all or some part of the system with units assumed to be good. Of course, if this works, we have solved the whole self-repair problem.

The ultimate solution to any repair problem is replacement. To the degree that we have a large stock of spare parts and a rapid and economical method of putting them in the system, we eliminate the need for any other diagnostic techniques. The best way to test a tube is to try a new one. If you can afford a large enough stock of tubes, you don't need a tube tester. Of course, the failure of a tube may have been caused by some other failure, and if you put in a new tube you may burn it out too. Still, if you have a large enough stock of tubes, this may be the cheapest way to find the other failure. (It is necessary to know that the "new" tube is a "good" tube, also.)

Consider the techniques used in computers today. We have all sorts of checks for errors, and we have diagnostic routines to find the cause of the trouble. But why bother with diagnostic routines? The surest way to trouble-shoot a computer is to replace circuit cards until you correct the trouble. But this has to be done manually and, with the number of cards in a modern computer, this may take too long. But the computer can run logical diagnostic checks at electronic speeds. When you consider the problem in this light, it becomes apparent that the ultimate answer to the problem of self repair lies not in the development of more complex and ingenious logical techniques, but rather in the development of electronic techniques for replacing components. However, we may expect that the development of such techniques is some years away, so a consideration of logical techniques of diagnosis is still pertinent.

Case III seems to be essentially the same as Case II-A, with respect to the diagnosis problem. For previously encountered inputs, we will compare with past performance. For new inputs, we will test the system on similar, but previously encountered, situations. In summary, the main ability needed for this part of the diagnostic problem is the ability to compare present with past performance. This would be true even if the systems were not learning, but only adaptive, since it is primarily the adaptive behavior that we are checking on. Thus, the main component needed is memory, and our learning systems already have this. So we may expect that when we have produced a learning system, we will have taken a large step towards producing a self-repairing system, and vice-versa.

Next, assuming that it has been determined that the trouble is in the system itself, how do we locate the faulty component? With respect to this aspect of the problem, there does not seem to be much significant difference among the cases discussed earlier. As discussed above, replacement of parts until the trouble is corrected is the most powerful technique, but we will assume for the present that this is practical only to a limited extent. The alternative will be to run certain tests on the entire system and various parts of the system. The level at which we make the tests will depend on the level at which we can make repairs. If our repair capability is limited to total replacement of various major blocks of the system, then our test procedure should be capable only of isolating the faulty block. We will not waste equipment trying to find out why the block is faulty if we cannot repair it anyway.

The main equipment requirements for this testing will be a considerable amount of memory, a means of generating and injecting test signals at various levels, and a means of measuring performance at appropriate levels. The need for signal generation and measurment is obvious. The memory will be required because it will be necessary to store a great deal of information about the system, such as tables of tests appropriate to various symptoms and lists of the proper values of various internal signals. It may be expected that the information needed will often be quite different from that required for operation of the system. For example, we can operate an oscilloscope without knowing much about what goes on inside of it. But if we are going to service it, we will need a schematic diagram and a list of the proper values of various voltages.

It might be possible to substitute logical capability for some of the memory, so that the system might "figure out" certain tests, rather than looking them up. Nevertheless, a considerable amount of memory would still be required. The success of the most ingenious technician in devising tests for a piece of faulty equipment will be directly related to the amount of information available to him.

In summary, we see that the capabilities required are not basically different from those involved in any learning system, but more of them will be required. Memory, the basic distinguishing characteristic of learning systems, is again an essential factor, and in this sense learning and self-repair overlap.

Next, we come to the problem of correction. Again there is no significant difference among the environmental classes. We assume that we may rule out the possiblity of a set of "mechanical hands" soldering in new parts. The only technique which appears generally reasonable is to have spare units available which can be switched into the system, either electronically or by means of relays. To the degree that we develop techniques for doing this economically for larger and larger numbers of parts, we solve the entire problem of self-repair, as discussed above.

An alternative approach to the correction problem would be the development of units which are self-healing in the manner of biological organisms. Unfortunately, practically nothing is known about such processes. There is some evidence that the proper structural forms are locally stored in some chemical code. Alternatively, the basic building blocks may be so related that the proper form is the only stable one. (This may be two ways of looking at the same phenomena.) In any event, we must adopt a "wait and see" attitude toward this approach and concentrate our attention on more conventional techniques.

Finally, we note that the actual learning ability might have value if we expect repetitive failures. If a system failed and was able to repair itself, it could store information on this experience which might make repair more efficient if there should be a recurrence of the same failure. However, the likelihood of single system being subject to a recurrent but unanticipated failure seems rather remote. A more likely situation might be that we would have a large number of identical independent units which could share their experiences. For example, we might have a hundred identical satellites which could communciate with one another and build up a common fund of failure information. We still have to decide if this is the best way to solve the problem. If we are going to provide communciation facilities anyway, why not send the information back to earth for processing by humans? We must recognize that, barring a major breakthrough in the areas of self-healing or electronic parts replacement, self-repair is going to be a very expensive proposition. It should generally be considered a last resort.

SECTION 1.6. SUGGESTIONS FOR FUTURE WORK

We feel that there are basically two jobs which remain to be done to bring this general study of machine learning to a logical conclusion. The first is a further study on the definition and model of the learning system discussed in Section 1.2. As indicated there, this definition is in the nature of a preliminary result, and we definitely feel that additional work is required.

The second job is to apply the concepts of Sections 1.2 and 1.4 to particular examples of large scale control problems. We feel that such an application is important, both to clarify the concepts and to test their validity. It is likely that modifications of the model and concepts would be suggested and our overall understanding greatly broadened. Among the areas that might be studied are air traffic control, missile and satellite control, process control and reactor control. We would welcome any suggestions from the contracting agency as to areas that might be of particular interest. It should be emphasized that we do not propose to become experts in any of these areas. Our intent would be to gain sufficient understanding of the general problems involved that we might meaningfully test our ideas on them.

We believe that the work outlined above would bring this <u>general</u> study of machine learning to a reasonable stopping point. Beyond this point, detailed study of specific problems and techniques would probably prove more profitable. Further general studies should await the accumulation of more experience and experimental data on practical problems

REFERENCES

1. Truxal, John G., "Trends in Adaptive Control Systems," Proceedings of the National Electronics Conference, 15:1-16, Oct. 1959.

2. Zadeh, L. A., "On the Definition of Adaptivity," Proc. IEEE, Vol. 51, pp. 470-471, March 1963.

3. Donalson, Dale D., The Theory and Stability Analysis of a Model Referenced Parameters Tracking Technique for Adaptive Automatic Control Systems, Ph.D. Dissertation, Univ. Of California at Los Angeles, 1961.

4. Gibson, John E., Non-Linear Automatic Control, McGraw-Hill, New York, 1963.

5. Bellman, Richard, Adaptive Control Processes, Princeton University Press, Princeton, N. J., 1961.