AD656083

FINAL REPORT

INFORMATION THEORY, LEARNING SYSTEMS, AND SELF-ORGANIZING CONTROL SYSTEMS

CONTRACT NONR-4467(00), Amendment No. 1
Authority-NR 048-195/12-30-64
SCOPE Job No. 623
31 March 1967

Sponsored by:

INFORMATION SCIENCES BRANCH
OFFICE OF NAVAL RESEARCH
WASHINGTON, D. C.

D D C
RECEIVED
AUG 10 1967
RECEIVED
B

Prepared by:

SCOPE INCORPORATED
2816 FALLFAX DRIVE
FALLS CHURCH, VIRGINIA 22042

REPRODUCTION IN WHOLE OR IN PART IS PERMITTED
FOR ANY PURPOSE OF THE UNITED STATES GOVERNMENT.

RECEIVED
AUG 15 1967
CFSTI

This document has been approved
for public release and sale; its
distribution is unlimited.

51

FINAL REPORT

INFORMATION THEORY, LEARNING SYSTEMS, AND SELF-ORGANIZING CONTROL SYSTEMS

CONTRACT NONR-4467(00), Amendment No. 1

Authority-NR 048-195/12-30-64

SCOPE Job No. 623

31 March 1967

Sponsored by:

INFORMATION SCIENCES BRANCH
OFFICE OF NAVAL RESEARCH
WASHINGTON, D. C.

Prepared by:

SCOPE INCORPORATED
2816 FALLFAX DRIVE
FALLS CHURCH, VIRGINIA 22042

REPRODUCTION IN WHOLE OR IN PART IS PERMITTED
FOR ANY PURPOSE OF THE UNITED STATES GOVERNMENT.

TABLE OF CONTENTS

ILLUSTRATIONS

I. INTRODUCTION

I. INTRODUCTION

This final report covers the results obtained under Contract NCNR4467(00). Two areas of investigation, relatec in basic concept but disparate in approach and application, have been considered in this program. The initial effort involved a conceptual modeling of learning and self-organizing systems in information theoretic terms. The second effort entailed transferring the conceptual form developed in the initial study into a control-system framework and resulted in an attractive form of model-reference control system.

Publications and Lectures:

Mr. Malcolm R. Uffelman gave a series of lectures on learning machines in the "Pattern Recognition-Models, Learning, Decision Theory," seminar conducted by the Information Sciences Institute of the University of Maryland June 28 through July 1, 1965. Mr. Uffelman also published a paper "Learning Systems and Information Theory" [1] 1966 IEEE International Communication Conference, Philadelphia, Pennsylvania, June 1966.

[1] Reproduced in the Appendix to this report.

II. TECHNICAL DISCUSSION

Part I. Information Theory and Learning Systems

In this part of the report, we shall consider the basic forms and functions of learning systems. Learning systems can be considered on three levels of complexity and all three are defined herein. However, it is shown that the adaptive system forms the core of each level and it is to this system that most of our attention is directed. A theorem specifying the necessary order of complexity of an adaptive system is presented and proven and some of its implementations are discussed.

Definitions:

1. Trained System: A system which learns to perform a desired task through some training procedure, but whose internal state is frozen at the completion of the training process. An example of a trained system is a linear threshold device made up of fixed resistor weights where the values of the resistors used are determined via a least-mean-square training algorithm using typical inputs. The learning process can be performed off line.

2. Adaptive System: A system that learns to perform a desired task through some training procedure and retains the ability to learn throughout the life of the system. The continued ability to learn implies the ability to improve performance via further training (i.e., on-the-job training),

to unlearn tasks, and to learn new or additional tasks. The numerous examples of adaptive systems include the CONFLEX I, MINOS II, and the MARK I PERCEPTRON.

3. Self-Organizing System: An adaptive system coupled with an automatic evaluation system and a built-in set of goals. The purpose of the evaluator is to direct the adaptive portion of the self-organizing system so that it develops a set of responses satisfying the goals*. Examples of self-organizing systems include the Homeostat and the MIT model reference-control system.

4. Learning Systems: Systems which can learn (with or without a teacher) to do jobs. Types of learning systems are trained, adaptive, and self-organizing systems.

Information Theoretic Models:

Assume the existence of a trained system. To introduce this approach to modeling learning systems, assume that the function to be performed is print reading (multifont). Figure 1 shows the information theoretic model to be employed.

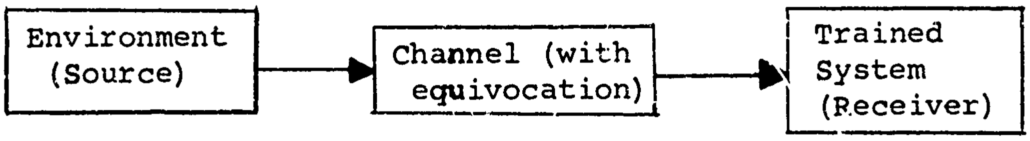


Figure 1
Simple Information Theory Model

* It should be noted that the goal system itself can be a self-organizing system that develops the higher goals of the overall system based on primitive goals. In fact, the final goal system can be composed of a hierarchy of self-organizing systems.

The source, called the environment, produces outputs that are the basic concepts involved in the problem at hand. In other words, the environment output is not reality but the abstract essence of physical reality. For example, when the environment output is some letter, say "B", it introduces into the channel only the concept "B" and not an IBM ELITE BACKSLANT "B" or a PICA bold face "B" or any other physical representation of "B".

For the message (i.e., the output of the environment) to reach the receiver, here taken as a trained system, it must be transmitted through a channel. As noted in figure 1, the channel is a noisy one, that is, it has equivocation. The output of the channel is a corrupted version of the environment output. In our model, the channel output has physical meaning and attributes.

The environment defined above is quite like the philosophy which Bishop Berkeley, an eighteenth century Irish philospher, put forth to refute materialism; matter does not exist except as a bundle of perceptions. Ultimate reality is the concept of the perceived reality. It is not necessary to accept this philosophy to use the model being proposed. The important idea is that all sensing machines, man included, can work only with signals taken at the output of the channel; but the general objective of the sensing machine and its superior parts, the combination of these being the trained system in our example, is to produce as an output the original concept or something functionally related to the original concept. In our print reader example, the environment output might be the "R" concept. Due to the equivocation of the channel, the input to the trained system might be a bold face block "R" with a broken cusp and surrounded with carbon smudges. The output of the trained system should be a code representing "R", without any indication of font or condition of print, in other words, the original concept.

A trained system is merely an adaptive system that has been taught a job and then had the ability to change internal states destroyed. Figure 2 shows the information theoretic model of an adaptive system.

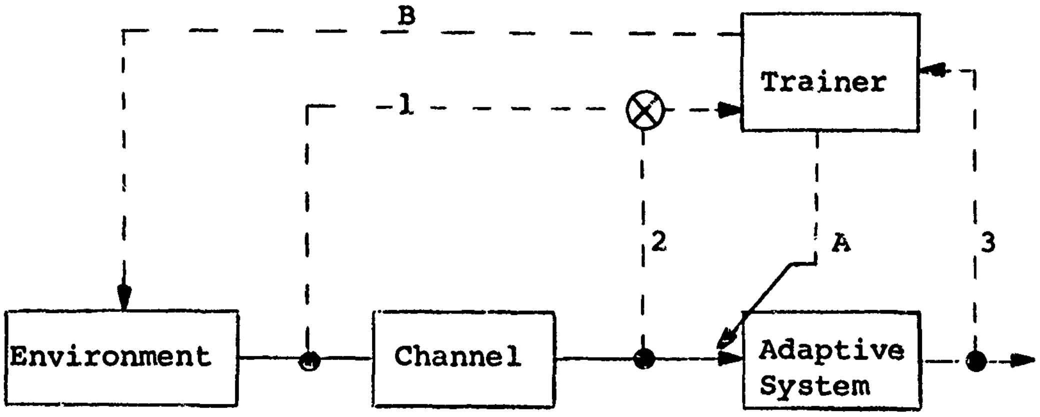


Figure 2. Information Theory Model, Adaptive System

The model is essentially the same as that for the trained system except that the receiver is now an adaptive system. The small arrow in the corner signifies that changes can be made in its internal state. The superstructure shown in dashed lines is the trainer and its lines of information-flow and control. All information paths and control paths need not be present in every situation. As shown, the trainer knows the truth (i.e., the environment output), physical reality (i.e., the channel output), and the system response; it can also control the environment and the state of the adaptive system. The trainer and its lines of communication are shown as dashed lines to remind us that they (or some part of them) are needed only when training is taking place or when the system performance is being monitored. The various combinations of information and control, such as AB23, A13, or A23 can each offer an interesting study into the behavior of the system. However, for the time being these studies will be postponed.

Figure 3 shows the model for a self-organizing system.

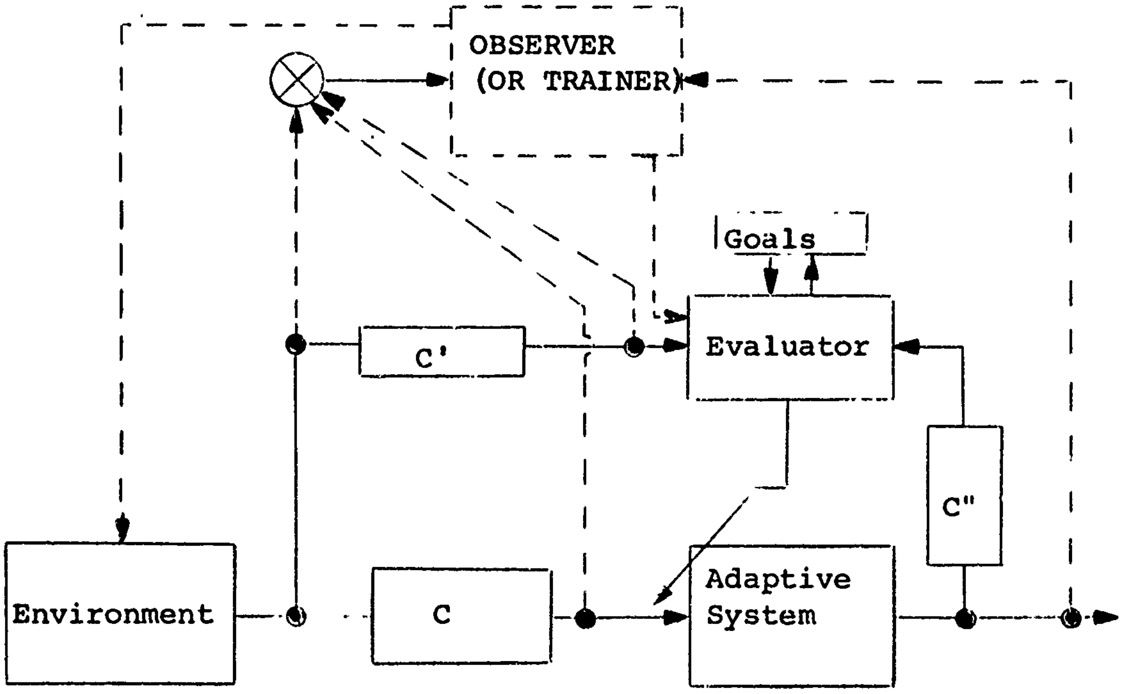


Figure 3
Information Theory Model, Self-Organizing System

Again, dashed lines show information and control lines for the trainer. (In a self-organizing system, the dashed lines more likely show the observer.) In the main structure, the combination of the evaluator and the goals are termed the goal system. The input to the goal system from the environment is through a channel, C', which is in general different from channel C. The goal system also has an input, different from C and C', from the adaptive system output through channel C". All channels can introduce noise. It is the purpose of the goal system to consider the response of the adaptive system and change the internal structure of the adaptive system as required to make the responses tend to satisfy the goals.

It should be noted that the goal system itself can be a self-organizing system based on more primitive goals. Such an organization would allow the main self-organizing system to develop higher-level goals based on the primitive goals established for the goal system.

Also notice that when a trainer is used to give instructions to the self-organizing system, the trainer does not have direct control of the adaptive portion as it did in the strictly adaptive system model of figure 2. In figure 3, the trainer can only affect the decision of the goal system. This is as it is in a biological teacher-student situation.

Analysis:

Assume the existence of the trained system illustrated in figure 1. The purpose of the system is to produce outputs related to the environment outputs. Without loss of generality, the system can be assumed to be a pattern-recognition system with the function of producing an output that is a coded form of the environment output. In general, the environment has a limited repertoire of outputs.

The output function of the environment can be represented as $P(E_i)$, the probability that output E_i will occur at a given time (a discrete ergodic source is assumed)*. The E_i are, as stated before, the concept of class; in other words, the E_i are the classes (or categories) to which the outputs of the channel belong and, in effect, each channel output is a noisy member of E_i. For simplicity, $P(E_i)$ at time t_k is taken to be independent of $P(E_i)$ at t_{k-1} (i.e., the same as selection with replacement). Consequently, $P(E_i)$ is the a priori probability of class E_i.

The output of the channel can be represented by $P(S_j|E_i)$, the probability of the physical stimulus, S_j, given the concept, E_i. Thus, as stated above, the channel introduces noise via

* A discrete source is assumed for convenience; an ergodic source is assumed because: (a) nature must have a high degree of stationarity or how could anything learn about it, and (b) nature must have its main concepts remain fixed over the ensemble or, again, how could anything learn about it?

a mapping of the concept into physical reality. The trained system must take measurements on the physical stimulus, and based on these measurements produce the concept as an output. Thus, the function of the receiver (in this case, the trained system) is to remove noise.

Assume that the form taken by physical reality at the channel output is a binary code; this removes the problem of noise being introduced by the measurements.

At this point, let us summarize. The function performed by a learning system is the removal of noise; thus, a learning system, after training (another name for designing and debugging), is a filter. After a filter is designed and working, it is of small interest. The design is the interesting part of a filter's life. Consequently, let us now turn our attention to the training of an adaptive system.

As done previously, we define the input pattern to the adaptive device as an array of n binary variables (1, -1). The set of patterns (S_{11}, S_{12}...S_{rn}) represents the corrupted versions of the classes (C_1, C_2,...C_r). During training, the adaptive device is adjusted so that it maps any input, S_i, onto the proper class concept, C_j. "Adjustment" means finding, by some procedure, an internal state of the device that performs the desired filtering function.

If we restrict our attention to a two-class problem, the number of possible filter functions (i.e., dichotomies) for N patterns is 2^N. In other words, with two concepts being emitted by the environment and with the noisy channel producing N patterns in response to the two concepts, the variability of the problem confronting the adaptive device is 2^N. Since we can physically grasp the signals only at the input of the adaptive device, this variability is the effective source variability. We call it the transfer variability, V_t, which can be considered the number of transfer states possible for N patterns. The base 2 logarithm V_t is called the transfer entropy, H_t. By some training procedure, we hope to adjust the adaptive device so that the output of the device, given the desired transfer state and the input pattern, S_a, is completely predictable by an outside observer.

The adaptive device has a number of possible internal states, each a different decision surface. We must be careful to distinguish between distinct internal states and the number of structural states. For example, in a simple linear-threshold device having two inputs, we can have more than one set of weights (i.e., more than one structural state) forming the same separating plane. Each different set of weights is a component of the number of structural states, but taken as a group, the weights form only one distinct internal state. The base 2 logarithm of the number of internal states is called the adaptive capacity, H_a, of the classifier.

THEOREM:

For an adaptive classifier to be able to perform all of the possible dichotomies of N patterns, it is necessary for H_a to at least equal H_t.

PROOF:

Assume that some form of adaptive classifier can achieve perfect classification for any and all dichotomies of N patterns with an adaptive capacity, H_a, less than the transfer entropy, H_t. This means, of course, that the classifier can dichotomize the patterns using fewer than 2^N internal states. If a table is made that relates each internal state to the dichotomy performed, there will be one or more internal states having more than one dichotomy listed with it. Therefore, each internal state can form more than one decision surface (i.e., can perform more than one dichotomy), or the dichotomies related to each of those internal states are the same. Both of the conclusions contradict the definitions; therefore, the initial assumption is wrong, and no form of adaptive classifier can perform all possible dichotomies if H_a is less than H_t.

Application:

Let us consider the application of the theorem stated above to a popular form of adaptive classifier, the linear threshold classifier shown in figure 4 on the following page.*

* Another form of adaptive classifier is also analyzed in the appendix.

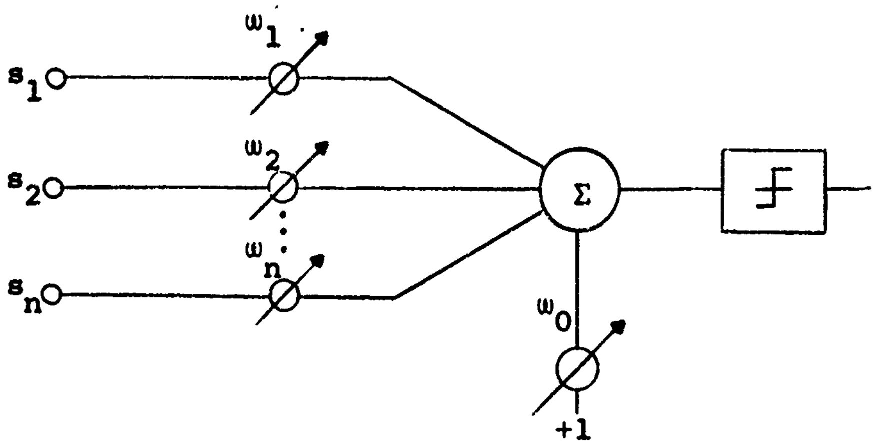


Figure 4
Linear Threshold Classifier

First let us assume, for the moment, that all of the dichotomies of N equal 2^n patterns are possible. Thus, the number of transfer states (or dichotomies) is 2^{2^n} and

$$\begin{aligned} H_t &= \log_2 2^{2^n} \\ &= 2^n \text{ bits} \end{aligned} \qquad (1)$$

Therefore, the device must have an adaptive capacity of 2^n bits. Since there would be no reason to expect any weight, W_i, to need more range than any other weight, W_j, we can find the capacity required by each weight, H_w.

$$H_w = \frac{2^n}{n + 1} \quad \text{bits per weight.} \qquad (2)$$

Of course, all of the dichotomies are not possible using a linear threshold classifier. Cameron (1) has established the upper bound on the number of linearly separable dichotomies of 2^n patterns, $R(n)$, as

$$R(n) < \sqrt{\frac{2}{\pi n}} \left(\frac{e2^n}{n} \right)^n \qquad (3)$$

If we don't try impossible dichotomies, then $\log_2 R(n)$ is less than the true H_t:

$$\begin{aligned} H_t &< \log_2 \sqrt{\frac{2}{\pi n}} \left(\frac{e2^n}{n}\right)^n \\ &< \tfrac{1}{2} \log_2 \left(\frac{2}{\pi n}\right) + n \log_2 \left(\frac{e2^n}{n}\right) \qquad (4) \\ &< \tfrac{1}{2} \left[\log_2 2 - \log_2 \pi n\right] + n \left[\log_2 e2^n - \log_2 n\right] \\ &< \tfrac{1}{2} \left[1 - \log_2 \pi n\right] + n \left[\log_2 e + n - \log_2{}^{n}\right] \end{aligned}$$

Rearranging the foregoing,

$$H_t < n^2 + n \log_2 e - (n + \tfrac{1}{2}) \log_2 n - \tfrac{1}{2} \log_2 \pi \qquad (5)$$

And, if n is large compared to unity, this is, without great error,

$$H_t \le n^2 - n \log_2 n \qquad (6)$$

Therefore, to a reasonable approximation,

$$H_a \le n^2 - n \log_2 n \qquad (7)$$

Again, having no reason to assume otherwise, we can compute that each weight needs

$$\frac{n^2 - n \log_2 n}{n + 1} \quad \text{bits} \qquad (8)$$

Or., if n is large, the classifier needs about

$$n - \log_2 n \text{ bits per weight.} \qquad (9)$$

Now, let us take a more reasonable approach. Several investigators (2) have shown that the natural capacity of a linear threshold device is 2(n+1) patterns. In other words,

if N is equal to or less than 2(n+1), and n is large, a linear threshold classifier can perform any desired dichotomy with probability near unity. Let us assume that a linear-threshold classifier can perform any dichotomy of n equal to or less than 2(n+1) patterns:

$$H_t = \log_2 2^{2(n+1)} = 2(n+1) \text{ bits} \qquad (10$$

therefore, H_a must at least equal 2(n+1) bits. Again, we can compute the average capacity required by each weight:

$$H_w = \frac{2(n+1)}{(n+1)} \qquad (11$$

$$= 2 \text{ bits per weight.}$$

Notice that this is, by the above development, only an average value and that it is a necessary condition.

Therefore, we can conclude that to perform dichotomies involving all possible patterns, a prohibitively large memory-capacity ($n-n \log_2 n/n+1$ bits per weight) is required. If a linear-threshold classifier is used within its natural capacity, the necessary capacity of the weights is quite modest (2 bits per weight).

Further Considerations:

One interesting form of adaptive system based on a perception-like organization is the Multivac (3), which uses a memory cell having a one-bit capacity (i.e., 1 bit per weight). Since our theorem says that for n weights, at 1 bit per weight, the machine can learn, at most, n patterns, let us consider under what conditions it can learn any dichotomy of n patterns.

THEOREM:

Given an n-dimensional space and n binary patterns in that space, then if the patterns are linearly independent, any dichotomy of them can be performed by a modulo 2 threshold-classifier.

PROOF:

Arrange the patterns in an n x n matrix, called the pattern matrix, with each row being a pattern. The problem can now be stated as follows: given a pattern matrix with linearly independent rows, a column matrix containing n binary elements (called the weight matrix) and a column matrix containing n binary elements (called the classification matrix) then for the following:

$$PW = C; \text{ Mod } 2$$

where: P is the pattern matrix,
W is the weight matrix, and
C is the classification matrix

the elements of W can be uniquely specified for any arrangement of ones and zeros in the C matrix.

This can be proven as follows. Since the P matrix has linearly independent rows, its left inverse (Mod 2) exists (see Peterson, "Error Correcting Codes," Wiley, 1961). Thus, we can write:

$$W = P^{-1}C; \text{ Mod } 2$$

which presents a means for computing the elements of W. Since we have found a way to compute W, the theorem is true.

The theorem can be translated from Modulo 2 to real positive numbers by having a threshold unit which decides even or odd rather than greater than.

Thus, the Multivac cannot, in general, classify in any dichotomy of (N+1) or more patterns, but for N or less linearly independent patterns and an odd-even threshold device it can perform any dichotomy.

Part II. A Self-Organizing Control System

Figure 3 of Part I shows the model of a self-organizing system. ' If we consider only linear systems, notice that as long as "C" communicates to the evaluator the output of the linear channel C in series with the adaptive system, then their order can be reversed and adaptation not. be affected. This reversal is shown in figure 5, where we now call the channel a closed-loop plant, and the adaptive system is a preprocessor of the input. Thus, we can translate the general model of the self-organizing system of Part I into a form of model reference-control system. The basic concept of this form of system is that the required adaptation takes place in the preprocessor located in the signal path.

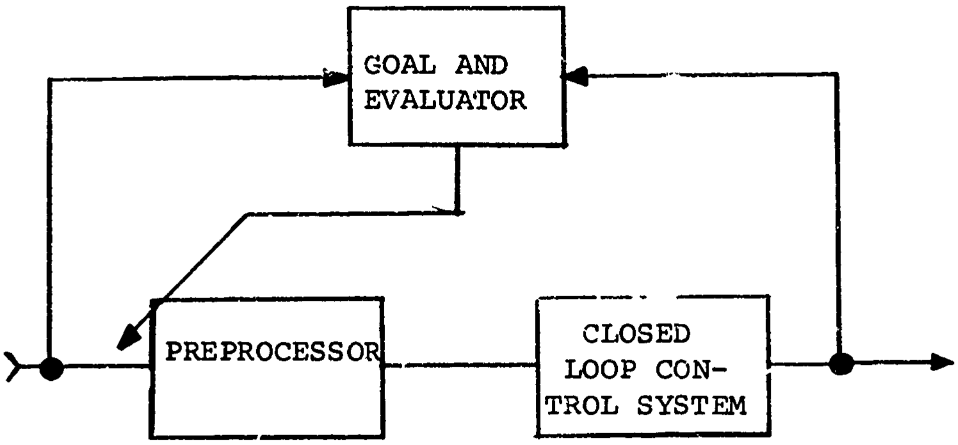


Figure 5
Block Diagram, Self-Organizing Control System

Two general conditions are imposed for this discussion: the closed-loop system is unconditionally stable for all changes in the plant function, and the closed-loop system is linear.

Initial Assumptions:

Since we shall be concerned with using a statistical measure to evaluate the behavior of the system, we assume that all signals are bounded real functions of time and that the plant function and the input signal are stationary over the time regions of interest. If y(t) is one of the signals, then $y_T(t)$ is

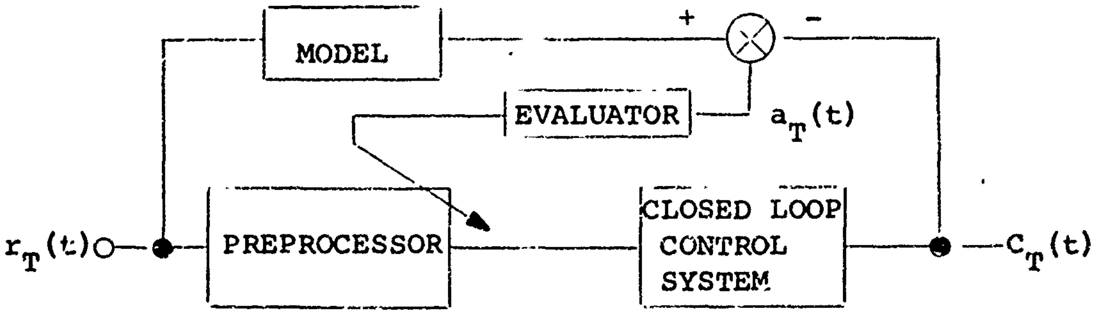
$$\begin{aligned} y_T(t) &= y(t) \; ; -T \geq t \geq T \\ y_T(t) &= 0 \quad ; \; t < -T, \; t > +T \end{aligned} \qquad (12)$$

and the Fourier transform $Y_T(jw)$ is defined by

$$Y_T(jw) = \int_{-T}^{+T} y_T(t) e^{-jwt} \, dt \qquad (13)$$

The System:

Figure 6 is a detailed block diagram of the self-organizing control system



Model Transfer Function M(jw)
Preprocessor Transfer Function K(jw)
Control System Transfer Function P(jw)

Figure 6
Detailed Block Diagram

The Mean Square Error Measure:

The mean-square error-measure is defined as:

$$\overline{\langle a^2(t) \rangle} = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{T} \left(d_T(t)-c_T(t)\right)^2 dt \qquad (14)$$

where d(t) is the desired system output, and
c(t) is the actual system output.

Following Wiener (4), we shall use this measure to evaluate the performance error, $a_T(t)$ of Figure 6. Our method of using the mean-square-measure is standard; we shall at all times attempt to adjust the parameters at our disposal in such a way as to minimize the mean-square error.

In adopting the mean-square measure, we are stating that we are willing to accept many small differences between the desired output and the actual output, but that large differences are to be heavily penalized. Obviously, there are cases where this is not a good measure (i.e., cases for which a miss is as good as a mile).

However, for most applications, the simplicity of the mean-square measure makes it attractive enough to use even if it results in suboptimal goal achievement. It is to these applications that the system described here is addressed.

The Adaptive Preprocessor:

The adaptive preprocessor is a network based on the synthesis procedure of Wiener (4). Its structural form is indicated in Figure 7.

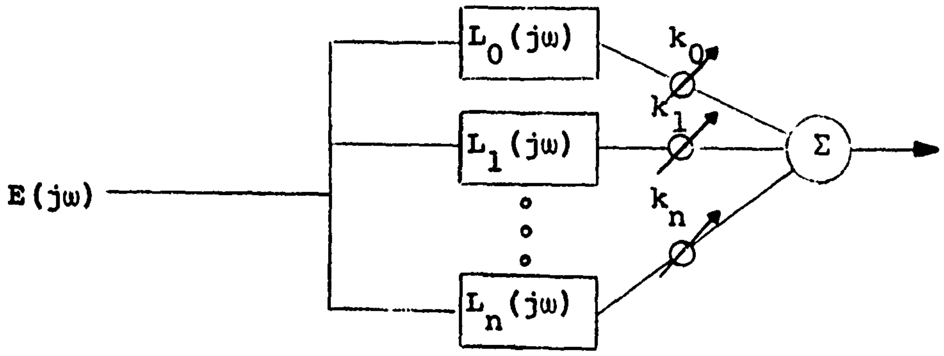


Figure 7
Preprocessor

The transfer functions, $L_i(jw)$, of the parallel filters are orthonormal functions (3) such as the Laguerre functions. Orthonormal functions are defined, for our purposes by

$$\int_{-\infty}^{+\infty} L_i(jw) L^*_j(jw)\,dw = \begin{cases} 1 \text{ if } i = j \\ 0 \text{ if } i \neq j \end{cases} \qquad (15)$$

where the asterisk denotes the complex conjugate.

The preprocessor can be used to approximate a transfer function, K(jw) by

$$K(jw) = \sum_{i=o}^{N} k_i L_i (jw) \tag{16}$$

where: $$k_i = \frac{1}{2\pi} \int_{-\infty}^{\infty} K(jw) L_i^*(jw) dw$$

We shall consider only those sets of functions which are complete (3).

Analysis:

From Figure 6, we can write

$$\begin{aligned} A(jw) &= M(Jw)R(jw) - C(jw) \\ &= R(jw) \left[M(jw)-K(jw)P(jw)\right] \end{aligned} \tag{17}$$

We desire to find an expression for the mean-square error $\overline{a^2(t)}$. By using Parseval's theorem, we can express $\overline{2^a(t)}$ in terms of A(jw):

$$\overline{a^2(t)} = \lim_{T\to\infty} \frac{1}{2T} \int_{-T}^{+T} a_T^2(t) dt \tag{18}$$

$$\overline{a^2(t)} = \lim_{T\to\infty} \frac{1}{2T} \left[\frac{1}{2\pi} \int_{-\infty}^{+\infty} A_T(jw) A_T^*(jw) dw\right]$$

We can simplify the notation by defining that equations used herein of the form

$$\overline{a^2(t)} = \int_{-\infty}^{+\infty} A(jw)A^*(jw)\,dw \qquad (20)$$

are to be interpreted as in Eq.19.

Using Equations 17 and 19:

$$\overline{a^2(t)} = \int_{-\infty}^{+\infty} R(jw)R^*(jw)\Big[M(jw)M^*(jw) - M(jw)K^*(jw)P^*(jw) - M^*(jw)K(jw)P(jw) + K(jw)P(jw)K^*(jw)P^*(jw)\Big]dw \qquad (21)$$

Substitution of Equation 16 into Equation 21 yields

$$\overline{a^2(t)} = \int_{-\infty}^{+\infty} R(jw)R^*(jw)\Big[M(jw)M^*(jw) - M(jw)P^*(jw)\sum_{i=o}^{N} k_iL_i^*(jw) - M^*(jw)P(jw)\sum_{j=o}^{N} k_jL_j(jw) + P(jw)P^*(jw)\sum_{m=o}^{N} k_mL_m(jw)\sum_{n=o} k_nL_n^*(jw)\Big]dw \qquad (22)$$

Differentiation of Equation 22 with respect to k_i yields

$$\frac{\partial \overline{a^2(t)}}{\partial k_i} = \int_{-\infty}^{+\infty} R(jw)R^*(jw) \Big[-M(jw)P^*(jw)Li^*(jw) \\ -M^*(jw)P(jw)L_i(jw) \\ +2k_iP(jw)P^*(jw)L_i(jw)L^*_i(jw) \Big] dw \qquad (23)$$

Differentiating again with respect to k_i yields:

$$\frac{\partial^2 \overline{a^2(t)}}{\partial^2 ki} = \int_{-\infty}^{+\infty} \Big[R(jw)R^*(jw)P(jw)P^*(jw)L_i(jw)L^*(jw)\Big] dw \quad (24)$$

From equation 13, we can see that there is a single value for k_i that will cause $\frac{\partial \overline{{}_a{}^2(t)}}{\partial k_i}$ to equal zero. Further, from equation 14, we can see that the second derivative of $\overline{a^2(t)}$ with respect to k_i is always positive, indicating that the single extremum is a minimum point.

Therefore, the mean-squared performance-error for the system is simply shaped surface (hyperparabolic) with a single minimum point. This form, therefore, avoids the problem common to the other form of model reference systems in which nonsimple surfaces must be searched.

A Simple Search Procedure

Although there are several methods of searching for and finding the minimum of a simple quadratic surface, only one will be discussed here.

First, let us rewrite equation 23.

$$\overline{\frac{\partial a^2(t)}{\partial ki}} = \int_{-\infty}^{+\infty} R(jw)R^*(jw)\left[-M(jw)P^*(jw)L_i^*(jw) - M^*(jw)P(jw)Li(jw)\right]dw + 2ki\int_{-\infty}^{+\infty} R(jw)R^*(jw)\left[P(jw)P^*(jw)Li(jw)L^*i(jw)\right]dw \quad (25)$$

If we perform the indicated integration, we obtain

$$\overline{\frac{\partial a^2(t)}{\partial ki}} = I_1 + 2kiI_2 \quad (26)$$

where I_1 and I_2 are real numbers.

Integration of equation 24 reveals

$$\overline{\frac{\partial^2 a^2(t)}{\partial ki^2}} = I_3 \quad (27)$$

where I_3 is a real number.

Equation 26 indicates that the minimum mean square error can be found by independent adjustment of each parameter, ki. There is a simple way to make this adjustment. Consider the general form of a quadratic in one variable, x,

$$y = Ax^2 + Bx + C \quad (28)$$

$$\frac{dy}{dx} = 2Ax + B \quad (29)$$

$$\frac{d^2x}{dx^2} = 2A \quad (30)$$

Simple algebra tells us that $y_{(min)}$ occurs at

$$x = \frac{-B}{2A} \qquad (31)$$

and that

$$y_{(min)} = C - \frac{B^2}{4A} \qquad (32)$$

If, in general, we know the value of x_0 at time t_0, the change in x, Δx, required to move to $y_{(m)}$ is

$$\Delta x = \frac{-\frac{dy}{dx}}{\frac{d^2y}{dx}} = -\left[x_0 + \frac{B}{2A}\right] \qquad (33)$$

Substituting equation 33 in equation 29, we obtain

$$\begin{aligned} y &= A(x_0+\Delta x) + C \\ &= A\left(x_0 - x_0 - \frac{B}{2A}\right)^2 + B\left(x_0 - x_0 - \frac{B}{2A}\right) + C \qquad (34) \\ y &= C - \frac{B^2}{4A} \qquad (35) \end{aligned}$$

which is, of course, the same result obtained in equation 32. The extension of this to the multivariable case is obvious.

Thus, the problem of searching for the minimum becomes a problem of evaluating the first and second partial derivatives with respect to each k_i and calculating the Δki required. Methods for performing these operations are well known (5).

In summary, one simple search procedure is to sequentially vary each ki, evaluate the first and second partial deviates for each ki, using the data obtained from the variation, and independently calculate the change required in each ki to obtain the minimum mean-square error.

In practice, two factors must be considered. First, the actual computation of $a^2(t)$ will most likely be performed by a low-pass filter. Wiener (4) has shown that this yields a good estimation of the true mean, if the filter time-constant is long with respect to the bandwidth of the signal to be averaged. Second, noisy measurements will prevent an accurate computation of δki. This will, in general, result in a failure to minimize the mean-square error. Repeated application of the search procedure of averaging the results of several variations of the k_i can reduce the amount of misadjustment at the minimum.

Experimental Study:

An experimental study of the self-organizing control system was conducted using general-purpose analog computers to simulate control plants and reference models. A special-purpose analog computer was built to provide a 10-stage Laguerre network. The general term of the Laguerre network, $L_n(s)$, is

$$\frac{\sqrt{2}p}{2\Pi} \frac{(p-s)^n}{(p+s)^{n+1}} \tag{36}$$

This can be rewritten as

$$\frac{\sqrt{2}p}{2\Pi(p+s)} \left[\frac{(p-s)}{(p+s)}\right]^n \tag{37}$$

The term $\frac{\sqrt{2_p}}{2\Pi(p+s)}$ can be realized by the circuit of figure

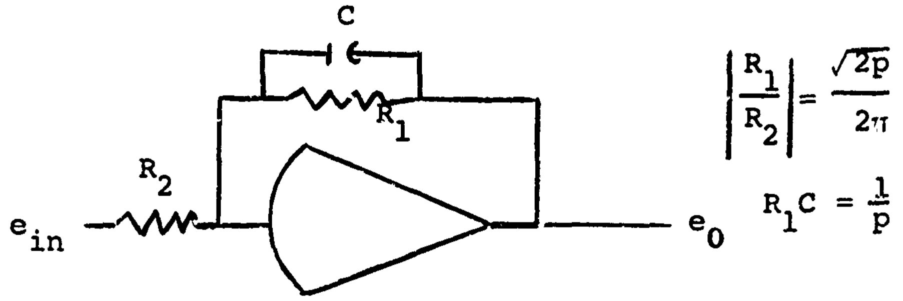


Figure 8. $L_0(s)$

The term $\frac{p-s}{p+s}$ can be realized by the circuit of figure

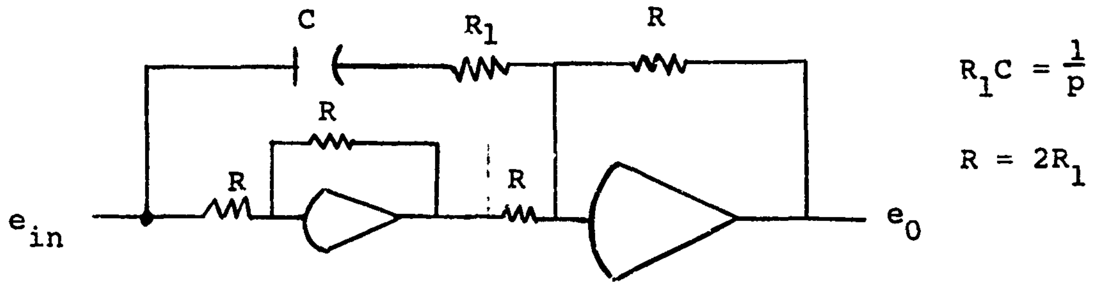


Figure 9

$\frac{(p-s)}{(p+s)}$ Realization

Thus any order function, $L_n(s)$, can be realized by one circuit, as shown in figure 8, followed by n cascaded circuits, as shown in figure 9. The simulator constructed for this study used a p equal to $\frac{1}{3 \times 10^{-2}}$.

A variety of plants and reference models were studied. Typical of these is the following:

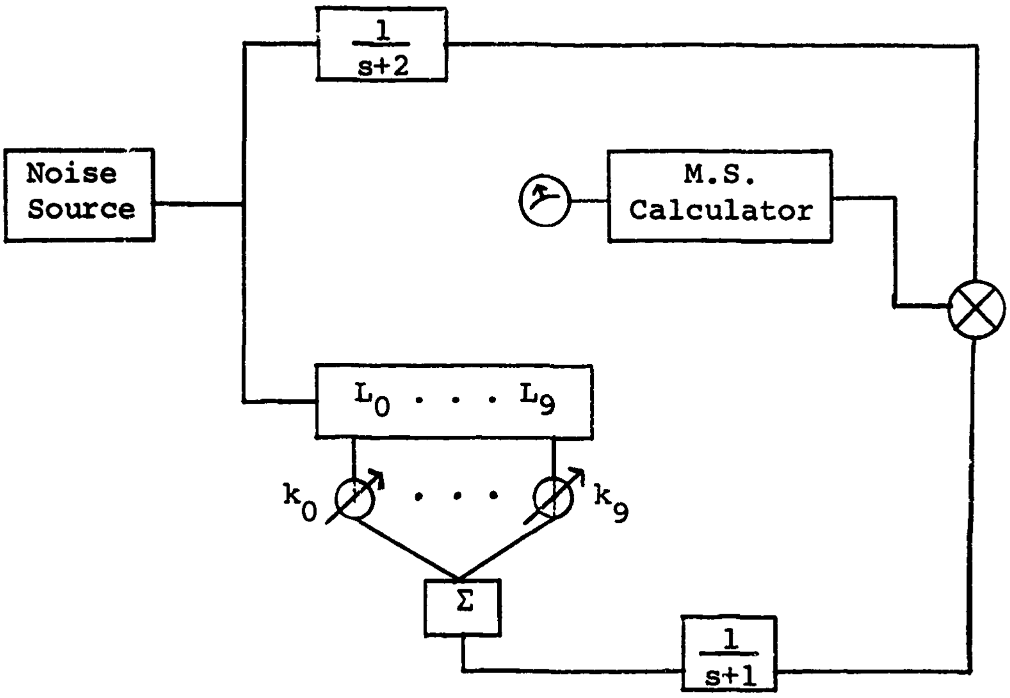
1. Plant open-loop transfer function

$$K(s) = \frac{1}{s}$$

2. Reference model

$$M(s) = \frac{1}{S+2}$$

The experimental set-up is shown in figure 10.



$$K(s) = \frac{1}{s} \; ; \; P(s) = \frac{\frac{1}{s}}{1 + \frac{1}{s}} = \frac{1}{s + 1}$$

Figure 10. Typical Experimental Set-Up

Figure 11 shows the mean-square error, $\overline{E^2(t)}$, for the variation around minimum for $L_0(t)$, $L_5(t)$ and $L_9(t)$ as a function of scale gain (i.e., as read off the indicators on the simulator.)

Figure 12 shows the $\overline{E^2(t)}$ for the same terms after the scale gains have been corrected for pot load. The pot loading curve is given in figure 13.

These results are typical of all cases tried and shown the quadratic nature of $E^2(t)$ as a function of the k_i's.

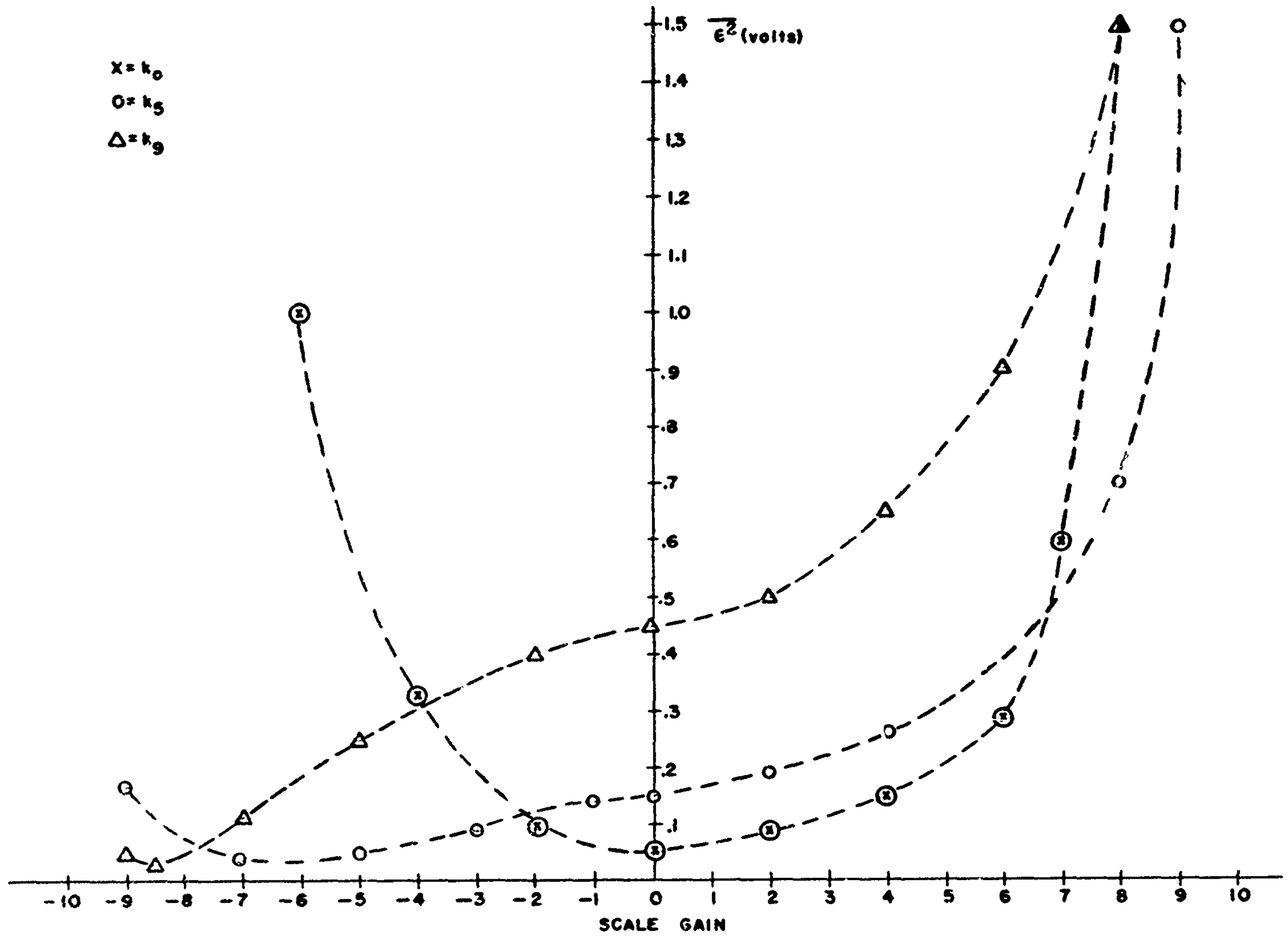


Figure 11. Mean-Square Error vs Scale Gain

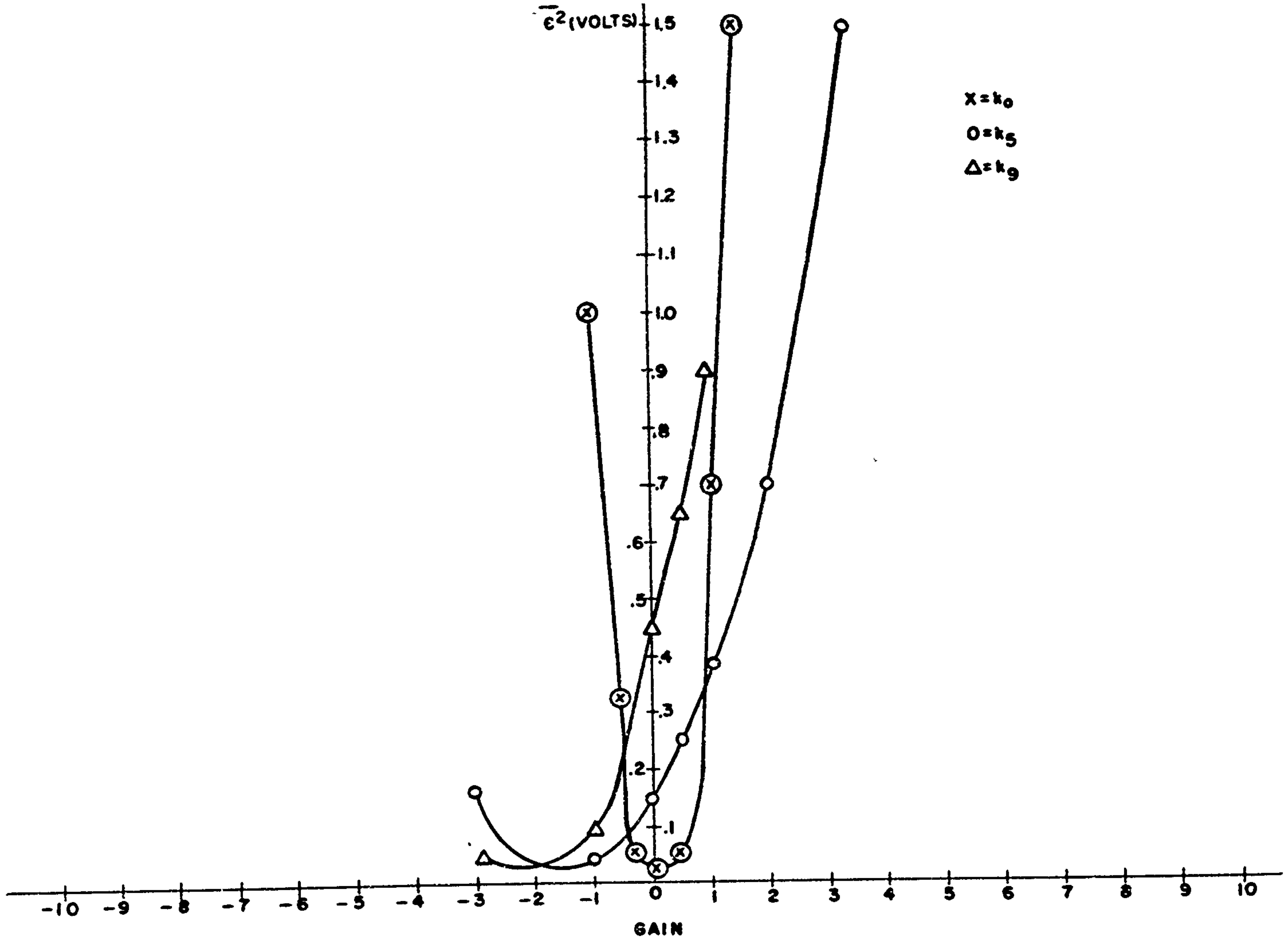


Figure 12. Mean Square Error vs Gain

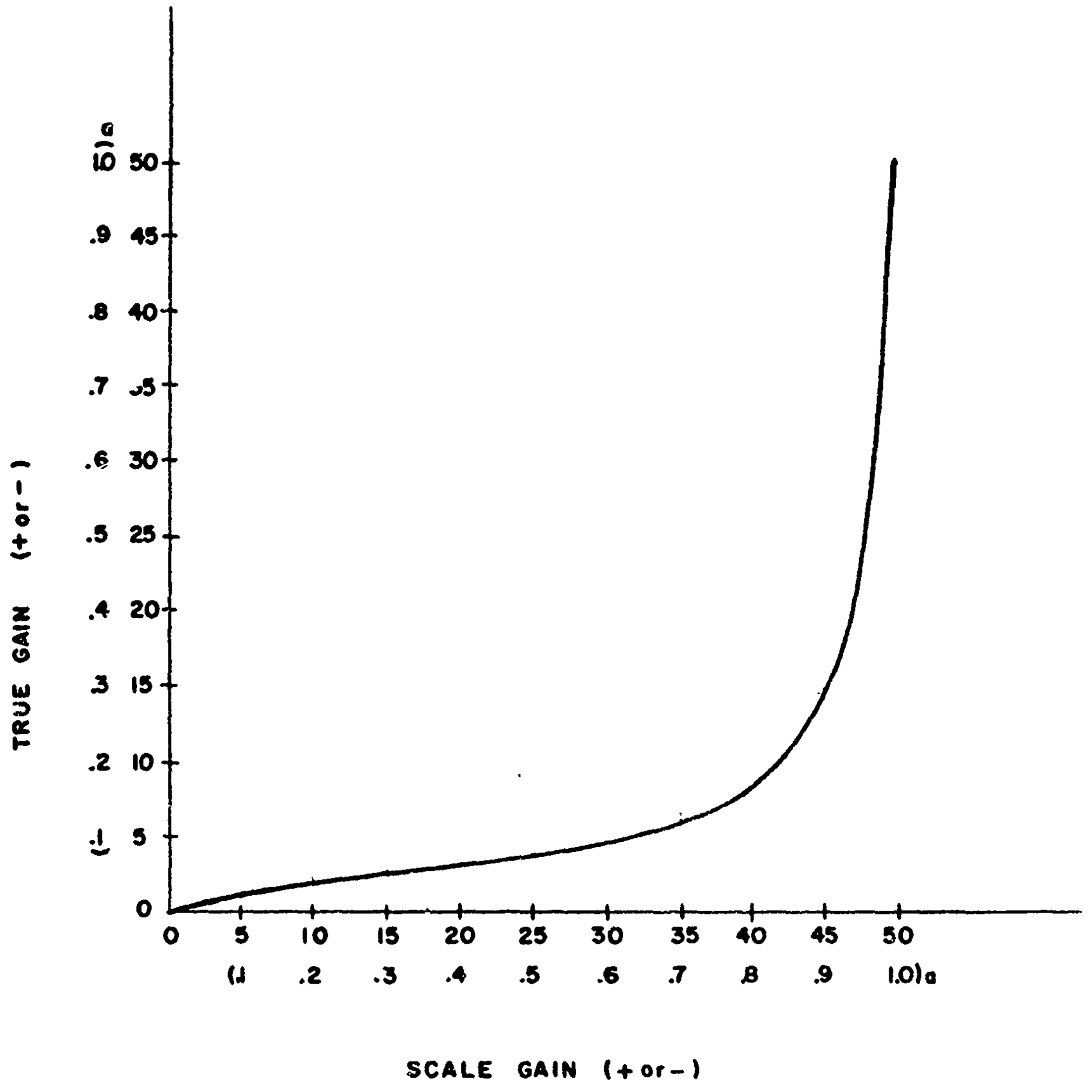


Figure 13. Loading Curve

LIST OF REFERENCES

1. Cameron, S.H., On Estimate of the Complexity Requisite in A Universal Decision Network, Proceedings of 1960 Bionics Symposium, Wright Air Development Division Technical Report 60-600 (December 1960).

2. Nilsson, N.J., Learning Machines, New York, 1965.

3. Asendorf, R.H., "A Modular Self-Organizing System," Biophysics and Cybernetic Systems, Washington, D.C., 1965.

4. Wiener, N., Extrapolation, Interpolation, and Smoothing of Stationary Time Series, New York, 1960.

5. Widrow, B., "Adaptive Sample-Data Systems," Technical Report No. 2104-1, Stanford Electronics Laboratories, Stanford University, Stanford, California, July 15, 1960.

APPENDIX

LEARNING SYSTEMS AND INFORMATION THEORY

M. Rucj Uffelman

SCOPE Incorporated
Falls Church, Virginia

INTRODUCTION

A learning system can be broadly defined as a system "whose actions are influenced by past experiences."[1] In this paper, we shall restrict our attention to adaptive pattern classifiers, a subclass of learning systems. It is generally accepted that the ability of adaptive pattern classifiers, either nonparametric or parametric, to classify correctly a set of patterns is limited by the structural or algorithmic composition of the system. An information theoretic model of an adaptive classifier and one limitation on performance are considered in this paper.

DISCUSSION

An elementary information theoretic model for an adaptive pattern classifier and its input mechanism is shown in Figure 1. For the purpose of this model, the source (or environment) is based on the philosophy which Bishop Berkeley, an 18th century Irish philosopher, put forth to refute materialism; matter does not exist as a bundle of perceptions. Ultimate reality, (represented by the source) is the concept of the perceived reality (represented by the output of the channel). Thus, in this model, the source produces only the abstract concept of a class. The perceived representation (i.e. the physical representation) which appears at the output of the channel is a corrupted or noisy version of the concept. Notice that the very process of taking on a physical representation is considered noise. The purpose of the adaptive classifier is to produce outputs which are either the concepts or related to the concepts. In other words, if the classifier is being used as a font reader, a chain of events might be: (The source produces as an output the concept of "R"; as a result, the channel produces an "R" in Canterbury Pica with a broken serif; and finally, the output of the classifier is a binary code representing "R". Thus, the classifier has, in effect, removed the noise introduced by the channel and returned the pure signal, the concept "R".

Based on the above, we can adopt the point of view that the classifier is a filter whose function is to remove noise. The degree to which it can remove the noise depends on the type and quantity of the noise and the functional complexity of the filter. By introducing two factors, akin to source entropy and channel capacity, it is possible to define a necessary condition on the ability of the classifier to remove noise.

ANALYSIS

For simplicity we shall take the channel output to be binary. If there are N patterns to be classified into 2 classes, there are 2^N possible dichotomies. In other words, the training amounts to adjusting the classifier to perform one of 2^N connective functions. The number of possible dichotomies is called the transfer variability and the base 2 logarithm of this number is called the transfer entropy, H_t. Notice that H_t is analogous to source entropy in that it is a measure of the variability forced upon the system.

The adaptive classifier has a number of possible internal states, each such state being a different decision surface. The training process involves finding the internal state which satisfies the desired dichotomy. The base 2 logarithm of the number of internal states is called the adaptive capacity, H_a, of the classifier.

THEOREM:

For an adaptive classifier to be able to perform all of the possible dichotomies of N patterns, it is necessary for H_a to at least equal H_t.

The proof of this involves assuming that a classifier can be trained to perform all dichotomies of N patterns where H_a is less than H_t. This leads to the conclusion that one internal state can perform more than one dichotomy of the patterns or that two or more of the dichotomies are identical. Both results contradict the definitions.

Now let us consider the application of this theorem to two well-known adaptive classifiers, the SOBLN[2] and the linear threshold classifier[1].

The SOBLN is a logical connective system having, in general, n binary inputs. As an example, a SOBLN for n = 3 is shown schematically in Figure 2. It computes the 2^n logical products and logically combines them using an OR gate and transmission weights to produce the output. It can be seen that the SOBLN can realize all dichotomies of N patterns up to the maximum N of 2^n. The maximum H_t is

$$H_t = \log_2 2^N = \log_2 2^{2^n} = 2^n \text{ bits}$$

Thus, H_a by the theorem must be at least 2^n bits. The SOBLN contains 2^n weights, W_i, which form the variable portion of its structure. Since there is no reason to expect any weight to require more variability than any other we can establish the variability necessary for each weight by

$$\frac{H_a}{\text{number of weights}} \geq \frac{H_t}{\text{number of weights}}$$

$$\geq \frac{2^n}{2^n} = 1 \text{ bit per weight}$$

Therefore, at least one bit per weight is necessary. And in this case, we can see by direct enumeration that one bit per weight is also sufficient.

Now let us proceed to the linear threshold classifier; an example is depicted in Figure 3. We know that for n binary inputs and $N = 2^n$ patterns, the classifier can not perform all 2^{2^n} dichotomies.[1] However, it has been shown by a number of investigators that the "natural capacity"[1] of a linear threshold device is 2(n + 1). In other words, if N is equal to or less than 2(n + 1) the probability that all dichotomies can be performed is approximately unity (for large n) and it is approximately zero if N is greater than 2(n + 1). Again taking a maximum case, we shall for the moment assume that for any 2(n + 1) binary patterns all dichotomies can be performed.

Thus,

$$H_t = \log_2 2^{2(n+1)} = 2(n + 1) \text{ bits}$$

And again, since we have no reason to suspect that any weight will require more flexibility than any other weight, we find that

$$\frac{H_a}{\text{number of weights}} \geq \frac{H_t}{\text{number of weights}}$$

$$\geq 2\frac{(n+1)}{(n+1)} = 2 \text{ bits per weight}$$

Thus, a linear threshold classifier must have at least 2 bits per weight to be able to perform all of the possible dichotomies within its "natural capacity" of 2(n + 1) patterns.

CONCLUSION

A necessary condition for an adaptive classifier to be able to be trained for a given task has been developed. In at least one case, the SOBLN, the theorem also leads to a sufficient condition. For the linear threshold classifier it has been shown experimentally and theoretically, that two bits or less per weight will, in many cases, do as well in a dichotomy problem as weights with greater variability.[3] This leads us to believe that for a linear threshold classifier with N equal to or less than 2(n + 1) and for those realizable dichotomies of the patterns, that 2 bits per weight are sufficient. This belief is, of course, still in the form of a conjecture, but we are working on its proof.

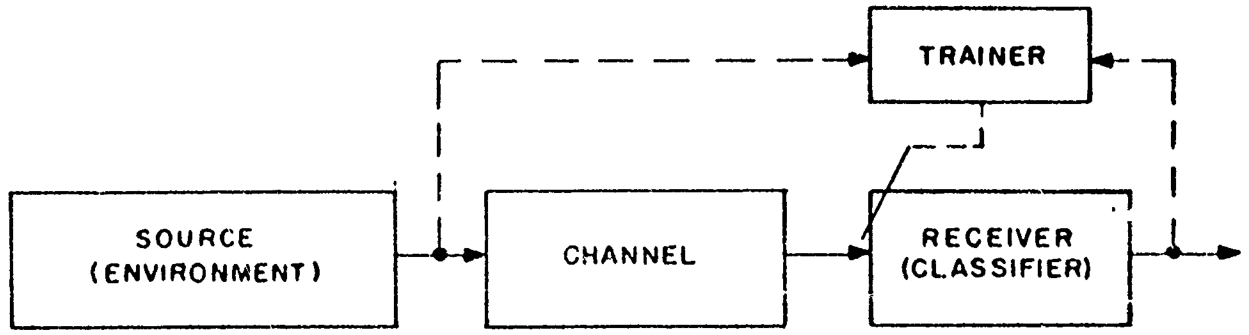


FIG 1-ELEMENTRY INFORMATION THEORETIC MODEL

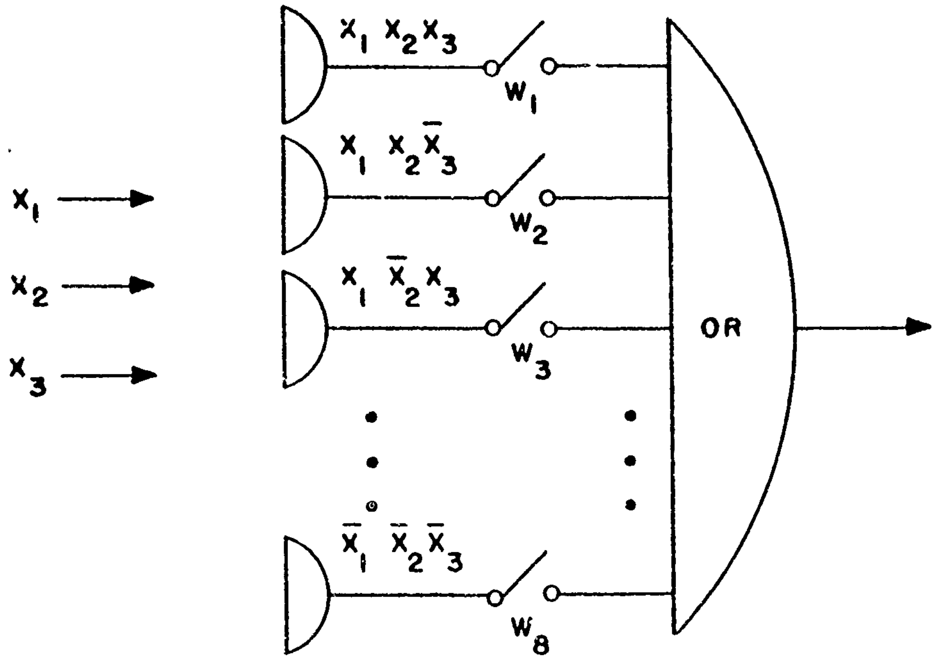


FIG 2-SOBLN CLASSIFIER

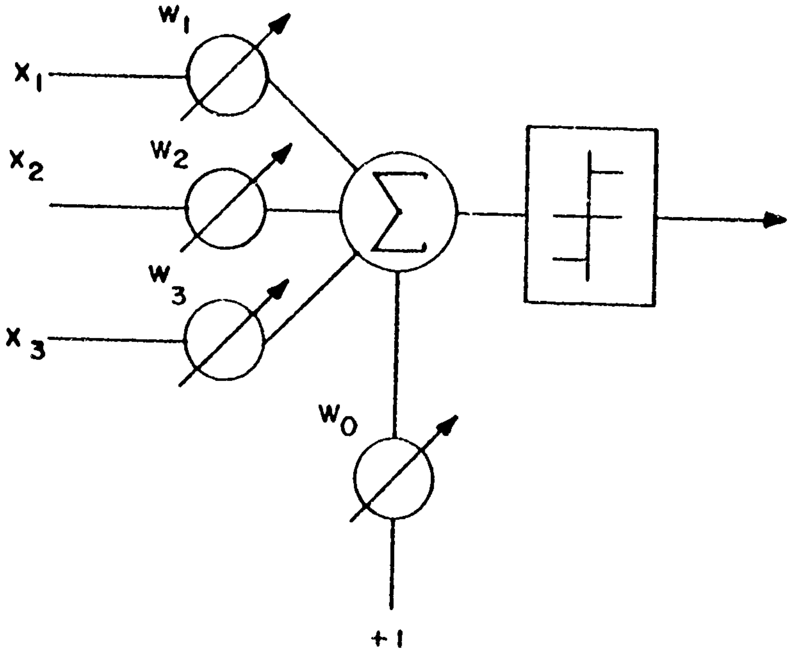


FIG 3-LINEAR THRESHOLD CLASSIFIER

ACKNOWLEDGEMENTS

I wish to express my appreciation to the Information System Branch of the Office of Naval Research for their support of this work. I also wish to thank Mr. Warren Holford, Mr. Norbert Kleiner, Mr. James Glenn and Dr. John Gerig for their helpful suggestions.

1. Nilsson, N. J., Learning Machines
   McGraw-Hill Book Company, New York, 1965.
2. Carne, E. B., A Study of Generalized Machine Learning,
   ASD-TDR-62-166 April 1962.
3. Uffelman, M. R., "CONFLEX I", International IRE Convention,
   Vol. 10, Part 4, New York March 1962.

UNCLASSIFIED

Security Classification

DOCUMENT CONTROL DATA - R&D

(Security classification of title, body of abstract and indexing annotation must be entered when the overall report is classified)

1 ORIGINATING ACTIVITY (Corporate author) SCOPE Incorporated	2a REPORT SECURITY CLASSIFICATION UNCLASSIFIED 2b GROUP

3 REPORT TITLE

A Self-Organizing Control System Study, Final Report

4 DESCRIPTIVE NOTES (Type of report and inclusive dates)

Final Report (1 April 1964 - 31 March 1967)

5 AUTHOR(S) (Last name, first name, initial)

Uffelman, Malcolm Rucj

6 REPORT DATE 31 March 1967	7a TOTAL NO OF PAGES 33	7b NO OF REFS 5

8a. CONTRACT OR GRANT NO.	9a ORIGINATOR'S REPORT NUMBER(S) SCOPE Job No. 623
b. PROJECT NO. c d	9b OTHER REPORT NO(S) (Any other numbers that may be assigned this report) None

10 AVAILABILITY/LIMITATION NOTICES

Yes

11 SUPPLEMENTARY NOTES None	12 SPONSORING MILITARY ACTIVITY Department of the Navy Office of Naval Research

13 ABSTRACT

Learning systems are defined on three levels of complexity, the trained system, the adaptive system, and the self-organizing system. The functional purpose of each is discussed in terms of the removal of noise and a theorem stating a necessary condition for adaption is stated and proven. The theorem is then applied to a simple form of adaptive system, and it is shown that for a linear threshold device employed within its "natural capacity", two bits per weight are necessary. The information theoretic model of the self-organizing system is translated into a goal directed control system model. This self-organizing control system model is analyzed and shown to have a simple performance surface that can be searched by relaxation methods. Experiments are discussed.

14 KEY WORDS	LINK A		LINK B		LINK C	
	ROLE	WT	ROLE	WT	ROLE	WT
Self-Organizing Systems						
Adaptive Pattern Recognition						
Adaptive Control						
Learning Systems						

INSTRUCTIONS

1. ORIGINATING ACTIVITY: Enter the name and address of the contractor, subcontractor, grantee, Department of Defense activity or other organization (*corporate author*) issuing the report.

2*a*. REPORT SECURITY CLASSIFICATION: Enter the overall security classification of the report. Indicate whether "Restricted Data" is included. Marking is to be in accordance with appropriate security regulations.

2*b*. GROUP: Automatic downgrading is specified in DoD Directive 5200.10 and Armed Forces Industrial Manual. Enter the group number. Also, when applicable, show that optional markings have been used for Group 3 and Group 4 as authorized.

3. REPORT TITLE: Enter the complete report title in all capital letters. Titles in all cases should be unclassified. If a meaningful title cannot be selected without classification, show title classification in all capitals in parenthesis immediately following the title.

4. DESCRIPTIVE NOTES: If appropriate, enter the type of report, e.g., interim, progress, summary, annual, or final. Give the inclusive dates when a specific reporting period is covered.

5. AUTHOR(S): Enter the name(s) of author(s) as shown on or in the report. Enter last name, first name, middle initial. If military, show rank and branch of service. The name of the principal author is an absolute minimum requirement.

6. REPORT DATE: Enter the date of the report as day, month, year, or month, year. If more than one date appears on the report, use date of publication.

7*a*. TOTAL NUMBER OF PAGES: The total page count should follow normal pagination procedures, i.e., enter the number of pages containing information.

7*b*. NUMBER OF REFERENCES: Enter the total number of references cited in the report.

8*a*. CONTRACT OR GRANT NUMBER: If appropriate, enter the applicable number of the contract or grant under which the report was written.

8*b*, 8*c*, & 8*d*. PROJECT NUMBER: Enter the appropriate military department identification, such as project number, subproject number, system numbers, task number, etc.

9*a*. ORIGINATOR'S REPORT NUMBER(S): Enter the official report number by which the document will be identified and controlled by the originating activity. This number must be unique to this report.

9*b*. OTHER REPORT NUMBER(S): If the report has been assigned any other report numbers (*either by the originator or by the sponsor*), also enter this number(s).

10. AVAILABILITY/LIMITATION NOTICES: Enter any limitations on further dissemination of the report, other than those imposed by security classification, using standard statements such as:

(1) "Qualified requesters may obtain copies of this report from DDC."

(2) "Foreign announcement and dissemination of this report by DDC is not authorized."

(3) "U. S. Government agencies may obtain copies of this report directly from DDC. Other qualified DDC users shall request through
________________."

(4) "U. S. military agencies may obtain copies of this report directly from DDC. Other qualified users shall request through
________________."

(5) "All distribution of this report is controlled. Qualified DDC users shall request through
________________."

If the report has been furnished to the Office of Technical Services, Department of Commerce, for sale to the public, indicate this fact and enter the price, if known.

11. SUPPLEMENTARY NOTES: Use for additional explanatory notes.

12. SPONSORING MILITARY ACTIVITY: Enter the name of the departmental project office or laboratory sponsoring (*paying for*) the research and development. Include address.

13. ABSTRACT: Enter an abstract giving a brief and factual summary of the document indicative of the report, even though it may also appear elsewhere in the body of the technical report. If additional space is required, a continuation sheet shall be attached.

It is highly desirable that the abstract of classified reports be unclassified. Each paragraph of the abstract shall end with an indication of the military security classification of the information in the paragraph, represented as (*TS*), (*S*), (*C*), or (*U*).

There is no limitation on the length of the abstract. However, the suggested length is from 150 to 225 words.

14. KEY WORDS: Key words are technically meaningful terms or short phrases that characterize a report and may be used as index entries for cataloging the report. Key words must be selected so that no security classification is required. Identifiers, such as equipment model designation, trade name, military project code name, geographic location, may be used as key words but will be followed by an indication of technical context. The assignment of links, roles, and weights is optional.