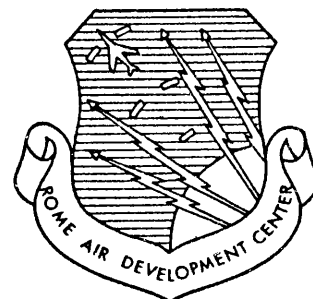


Nils Nilsson

**RADC-TDR-64-145
FINAL REPORT**

Nils J. Nilsson
Computer Science Department
Stanford University
Stanford, CA. 94305-4110



**LINEAR SEPARABILITY OF SIGNAL SPACE AS A BASIS FOR
ADAPTIVE MECHANISMS**

TECHNICAL DOCUMENTARY REPORT NO. RADC-TDR-64-145

May 1964

**Information Processing Branch
Rome Air Development Center
Research and Technology Division
Air Force Systems Command
Griffiss Air Force Base, New York**

Project No. 5581, Task No. 558104

**(Prepared under Contract No. AF30(602)-2943 by N. J. Nilsson,
Stanford Research Institute, Menlo Park, California. Approved:
Charles A. Rosen, Manager; J. D. Noe, Director)**

When US Government drawings, specifications, or other data are used for any purpose other than a definitely related government procurement operation, the government thereby incurs no responsibility nor any obligation whatsoever; and the fact that the government may have formulated, furnished, or in any way supplied the said drawings, specifications, or other data is not to be regarded by implication or otherwise, as in any manner licensing the holder or any other person or corporation, or conveying any rights or permission to manufacture, use, or sell any patented invention that may in any way be related thereto.

Qualified requesters may obtain copies from Defense Documentation Center.

Defense Documentation Center release to Office of Technical Services is authorized.

Do not return this copy. Retain or destroy.

Suggested Keywords: Artificial intelligence, Learning machines,
Adaptive mechanisms, Pattern recognition, Perceptron

ABSTRACT

This report reviews the research results of the program entitled "Linear Separability of Signal Space as a Basis for Adaptive Mechanisms." The major contributions of this program have been two fold: (1) the notion of discriminant functions has been employed in constructing a framework for organizing past and present knowledge into a basis for further theoretical research on trainable pattern classifying machines, and (2) some significant new results have been obtained on trainable pattern classifying machines. The specific research efforts reported here fall into the following categories:

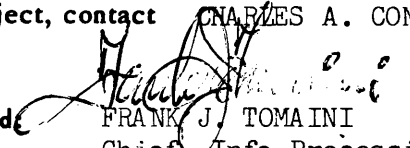
- (1) Investigation of the properties of various families of discriminant functions to be used by a pattern dichotomizer;
- (2) Investigation of various network structures for the implementation of useful families of discriminant functions; and
- (3) Investigation of various training rules to be used in selecting the appropriate discriminant function for a pattern dichotomizer.

Some of the research conducted during this program has been reported in technical notes. In reviewing the total program, this report references the technical notes for the details they contain, but presents detailed discussions of material not previously reported in technical notes.

PUBLICATION REVIEW

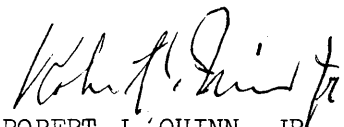
This report has been reviewed and is approved. For further technical information on this project, contact CHARLES A. CONSTANTINO, EMIID, extension 5146.

Approved:


FRANK J. TOMAINI

Chief, Info Processing Branch

Approved:


ROBERT J. QUINN, JR.

Col, USAF

Chief, Intel and Info Processing Div

FOR THE COMMANDER:


IRVING J. GABELMAN

Chief, Advanced Studies Group

CONTENTS

I	TRAINABLE PATTERN CLASSIFYING MACHINES	1
	A. Introduction	1
	B. The Basic Model	2
	C. On the Selection of Discriminant Functions--Training	5
	D. A Specialization	6
II	PROPERTIES OF FAMILIES OF DISCRIMINANT FUNCTIONS	9
	A. Linear Discriminant Functions	9
	B. Φ -Functions	12
	C. The Utility of Φ -Functions for Classifying Patterns	13
	D. Machine Capacity	14a
	E. Piecewise Linear Discriminant Functions	16
III	PROPERTIES OF NETWORK STRUCTURES FOR IMPLEMENTATION OF PIECEWISE LINEAR DISCRIMINANT FUNCTIONS	19
	A. Layered Machines	19
	B. A Theorem about Two-Layer Machines	19
	C. Committee Machines	22
	D. A Summary of Results of Majority Logic Machines	25
	E. An Example Dichotomy Illustrating the Use of Majority Logic Machines	29
IV	TRAINING METHODS FOR THE SELECTION OF DISCRIMINANT FUNCTIONS	30
	A. Error Correction Training Methods	30
	B. Generalization	33
	C. Applications of Φ -Machines	37
	D. The Training Problem for Majority Logic and Other Piecewise Linear Machines	38
V	CONCLUSIONS	41
	APPENDIX A	43
	APPENDIX B	54
	REFERENCES	59
	ILLUSTRATIONS	61

TECHNICAL DOCUMENTARY REPORTS
PREPARED DURING THE PROGRAM

H. D. Block, N. J. Nilsson, and R. O. Duda, "Determination and Detection of Features in Patterns".

T. M. Cover, "Classification and Generalization Capabilities of Linear Threshold Units".

D. J. Kaylor, "A Mathematical Model of a Two-Layer Network of Threshold Elements".

N. J. Nilsson and R. C. Singleton, "An Experimental Comparison of Three Learning Machine Training Rules".

B. Efron, "The Perceptron Correction Procedure in Non-Separable Situations".

I TRAINABLE PATTERN CLASSIFYING MACHINES

"The objective of this program is to develop a mathematical basis for the analysis and synthesis of learning machines. The program would have the following ultimate goals:

- (1) Identification of the pertinent problems,
- (2) Solution of the pertinent problems, and
- (3) Organization of the results into a general theory of learning machines."

A. INTRODUCTION

This report will review the research results of the program entitled, "Linear Separability of Signal Space as a Basis for Adaptive Mechanisms." The review will evaluate these results in the light of the program's stated objectives quoted above.

The main conclusion of this report will be that a basis for a general theory of trainable pattern classifying machines^{*} has been developed. This theoretical basis will be described in this section of the report; it provides a framework in which to organize present knowledge and displays clearly those questions for which answers have not yet been obtained. After discussing the general theory we shall outline those aspects of the theory that have been studied in detail during this program. Subsequent sections of this report will present the results of these studies.

*

We shall use the phrase "trainable pattern classifying machines" instead of "learning machines" in this report. This substitution was made in the interest of naming more precisely the intended area of research.

B. THE BASIC MODEL

The theory to be presented deals with trainable pattern classifying machines. We shall first discuss what is meant by a pattern classifying machine. A pattern classifying machine is a device which accepts input data and responds with an output indicating the classification of this data. Such a device could be used to perform tasks such as weather prediction, speech and character recognition, medical diagnosis, and speaker identification. We shall assume that each set of data to be classified is a set of d real numbers, x_1, x_2, \dots, x_d . Such a set we shall call a pattern, and we shall call the individual numbers components of the pattern.

An immediate problem confronting the designer of a pattern classifying machine for a specific application, such as character recognition, is the problem of what physical measurements should be made on the character to be recognized. The values of these measurements constitute the components of the pattern. Several authors have distinguished between these two aspects of recognition problems: measurement of important properties yielding a pattern, followed by classification of the pattern. These aspects are often called, respectively, the measurement problem and the decision problem. With one exception, the research conducted in this program was addressed to the decision problem. The exception was the feature detection research (the results of which were reported in full in a technical note^{1*} submitted during the program) which could have applications in the selection of measurements. Otherwise, we have assumed that the d measurements yielding the pattern to be classified have been selected as wisely as possible while remembering that

* References are listed at the end of this report.

the pattern classifier cannot, itself, compensate for a careless selection of measurements.

Suppose that there are R categories into which the patterns must be classified. We shall label these categories by the integers $1, 2, \dots, R$. One of these integers, perhaps R , might correspond to a "reject" or "null" category.

It will be convenient to represent a pattern as a point in a d -dimensional Euclidean space, E^d , called the pattern space. The rectangular coordinates of the point are the real numbers x_1, x_2, \dots, x_d . The vector, \vec{X} , extending from the origin to the point $\{x_1, x_2, \dots, x_d\}$ can also be used to represent the pattern. For notational convenience we shall represent the point $\{x_1, x_2, \dots, x_d\}$ by the symbol " \vec{X} " also.

A pattern classifying machine is thus a device which maps the points of E^d into the category numbers, $1, 2, \dots, R$. The mapping can be described by the surfaces, called decision surfaces, separating the space into R regions: R_1, R_2, \dots, R_R . The i -th region, R_i , is the set of points which are mapped into Category i .

The mapping achieved by a pattern classifying machine can also be described by a set of functions, R in number, which we shall call discriminant functions. Let $g_1(\vec{X}), g_2(\vec{X}), \dots, g_R(\vec{X})$ be scalar and single-valued functions of the pattern \vec{X} . These discriminant functions are chosen such that for all \vec{X} in R_i , $g_i(\vec{X}) > g_j(\vec{X})$ for $i, j = 1, \dots, R, j \neq i$. That is, in R_i the i -th discriminant function has the largest value. The decision surface separating contiguous regions R_i and R_j is given by the equation

$$g_i(\vec{X}) - g_j(\vec{X}) = 0 \quad (1)$$

The location and form of the decision surfaces do not uniquely specify the discriminant functions. For one thing, the same arbitrary constant can be added to each discriminant function without altering the decision surfaces. In general, any monotonic increasing function (e.g., logarithmic) can be used to convert a set of given discriminant functions into an equivalent set.

The notion of discriminant functions suggests a convenient method of implementing the decision surfaces for a pattern classifying machine. This method is illustrated by the block diagram in Fig. 1.* Each of the discriminators computes the value of a discriminant function. The outputs of the discriminators will be called discriminants. In classifying a pattern, \vec{X} , the R discriminants are compared by a maximum selector which indicates which one is largest. If the i_0 -th discriminant is the largest, \vec{X} is placed in Category i_0 .

Using the model in Fig. 1 we can state that the central problem in the design of pattern classifying machines is the specification of the discriminant functions g_1, g_2, \dots, g_R . In this program we are interested in one particular method of design which we call training. This method will now be discussed in more detail.

*The figures are at the end of this report.

C. ON THE SELECTION OF DISCRIMINANT FUNCTIONS--TRAINING

The discriminant functions for a pattern classifying machine can be specified in a variety of ways. Sometimes they can be calculated with precision on the basis of complete a priori knowledge about the patterns to be classified. At other times, reasonable guesses are made on the basis of qualitative knowledge about the patterns. In each of these cases, especially in the second, it may be necessary to "touch-up" or "debug" the discriminators to achieve acceptable performance on actual patterns. This debugging should be performed by using a set of patterns which are representative of the actual patterns which the machine must classify.* Debugging is always an important phase in the design of any equipment. In this program we are interested in those cases in which it is the major phase.

Here, we are interested in a particular form of debugging known as training. The training process proceeds as follows: a large number of patterns are chosen as typical of those which the machine must ultimately classify. This set of patterns is called the training set. The desired classifications of these patterns are assumed to be known. Discriminant functions are then selected from among the members of a given family of discriminant functions by a process

* The debugging can occur after the machine is constructed by making adjustments in the organization, structure, or parameter values of the parts of the machine, or it can occur before hardware construction by making these adjustments on a simulated machine using, for example, a digital computer.

which iteratively adjusts the parameters of the discriminant functions. The parameters are adjusted in such a way that the pattern classifying machine converges to a state in which all of the patterns in the training set are correctly classified.* When the pattern classifying machine reaches this state it is said to have been trained on the pattern set. A pattern classifier employing discriminant functions with adjustable parameters shall be called a trainable pattern classifier.

Several procedures for the iterative adjustment of discriminant function parameters have been proposed. The training rules of the α -perceptron² and the Madaline³ are examples of such procedures; these and others shall be discussed later in this report.**

D. A SPECIALIZATION

In this program we have been primarily interested in the case $R = 2$; that is, we have assumed that the number of pattern categories is equal to 2. In this case we call a pattern classifier a pattern dichotomizer. Only pattern dichotomizers will be considered from now on in this report.

*

We shall assume throughout this report that the family of discriminant functions originally chosen for the pattern classifying machine is such that correct classification of all patterns in the training set is possible.

**

Statistical techniques have also been suggested as a means for selecting the appropriate discriminant functions for a pattern classifier. Many of these techniques make use of a set of "typical" patterns whose categories are known. For an excellent survey of these statistical methods, see Harley, et al.⁴

The pattern classifier shown in Fig. 1 takes an interesting form when there are only two categories. Here, the maximum selector must decide which is the larger, $g_1(\vec{X})$ or $g_2(\vec{X})$. It turns out that this decision can be implemented by evaluating the sign of a single discriminant function $g(\vec{X}) \triangleq g_1(\vec{X}) - g_2(\vec{X})$. If $g(\vec{X})$ is positive, \vec{X} is placed in Category 1; if $g(\vec{X})$ is negative, \vec{X} is placed in Category 2. $g(\vec{X}) = 0$ is the equation of the decision surface separating the regions R_1 and R_2 . The sign of $g(\vec{X})$ can be evaluated by a threshold element whose threshold value is equal to zero. For this reason the threshold element assumes an important role in pattern dichotomizing machines.

The basic model of a pattern dichotomizer is shown in Fig. 2. Based on this model the main research efforts undertaken in this program were the following:

- (1) Investigation of the properties of various families of discriminant functions, $g(\vec{X})$, to be used by a pattern dichotomizer.
- (2) Investigation of various network structures for the implementation of useful families of discriminant functions.
- (3) Investigation of various training rules to be used in selecting the appropriate discriminant function of a pattern dichotomizer from among a given family of discriminant functions.

Much of this work has already been reported in technical notes issued during the program. In this final report of the program we shall draw this and related material together, in summary form,

referring the reader to the appropriate technical notes for the details. We shall also present some new material not previously reported in the technical notes.

II PROPERTIES OF FAMILIES OF DISCRIMINANT FUNCTIONS

A. LINEAR DISCRIMINANT FUNCTIONS

The task of selecting a discriminant function for use in a pattern classifying machine is made simpler if we first limit the class of functions from which we make this selection. For this reason our attention is drawn to consider families of discriminant functions. A discriminant function family can be defined through the use of parameters whose values determine the members of the family. For example, suppose a discriminant function $g(\vec{X})$ depends also on the values of the M parameters w_1, w_2, \dots, w_M . We make this dependence explicit by writing $g(\vec{X})$ in the form

$$g(\vec{X}) = g(\vec{X}; w_1, w_2, \dots, w_M) \quad (2)$$

The set of functions obtainable by varying the values of the parameters throughout their ranges, is called a family of functions. A particular function belonging to this family can be selected by choosing the appropriate values of the parameters. The training of a machine bound to employ discriminant functions belonging to a particular family can then be accomplished by adjusting the values of the parameters. We shall call these parameters weights. In this program, we are interested only in those pattern classifying machines whose discriminant functions are obtained by selecting or adjusting the values of weights.

Let us consider first the family of discriminant functions of the form

$$g(\vec{X}) = w_1x_1 + w_2x_2 + \dots + w_dx_d + w_{d+1} \quad (3)$$

This function is a linear function of the components of \vec{X} ; we shall denote discriminant functions of this form by the term linear discriminant function. A complete specification of any linear discriminant function is achieved by specifying the values of the weights or parameters of the function family.

A pattern dichotomizer employing a linear discriminant function can be simply implemented by using weights, a summing device, and a threshold unit. Such a machine is depicted in Fig. 3. This form for a pattern dichotomizer has been variously called an artificial neuron, a linear input logic unit, an Adaline, and an A-unit. In this program we have called it a threshold logic unit (TLU). The properties of TLUs, when the input pattern, \vec{X} , is restricted to binary components have been well studied by switching theorists. For an excellent survey of this literature, see Winder.⁵ As a device for pattern recognition, the TLU has also received much attention. In this connection, see Widrow and Hoff.⁶

We shall ordinarily assume that the TLU responds with a +1 signal if $g(\vec{X}) > 0$ and a -1 signal if $g(\vec{X}) < 0$. We must then associate a TLU output of +1 with pattern Category 1 and a TLU output of -1 with pattern Category 2. The last term in the expression for $g(\vec{X})$, w_{d+1} , is usually provided by a weight whose value, w_{d+1} , is energized by a signal of +1. Usually this +1 signal is associated with

the pattern as a $(d+1)$ -th input, x_{d+1} , whose value is always equal to $+1$.

The TLU implements a hyperplane decision surface given by the equation

$$w_1x_1 + w_2x_2 + \dots + w_dx_d + w_{d+1} = 0 \quad (4)$$

This hyperplane divides the pattern space into two regions, R_1 and R_2 . In R_1 the TLU responds with a $+1$. On the other side of the hyperplane, in R_2 , the TLU responds with a -1 . Some of the properties of hyperplane decision surfaces have been catalogued by Highleyman.⁷

If a set of patterns, χ , is dichotomized into two subsets χ_1 and χ_2 such that these subsets can be separated by a hyperplane decision surface, we say that the subsets are linearly separable, and that the dichotomy of χ is a linear dichotomy. That is, χ_1 and χ_2 are linearly separable if and only if there exists a set of weights $\{w_1, w_2, \dots, w_d, w_{d+1}\}$ such that

$$x_1w_1 + x_2w_2 + \dots + x_dw_d + w_{d+1} > 0$$

for \vec{X} in χ_1

and

$$x_1w_1 + x_2w_2 + \dots + x_dw_d + w_{d+1} < 0$$

for \vec{X} in χ_2

B. Φ -FUNCTIONS

A Φ -function with parameters (weights) w_1, w_2, \dots, w_{M+1} , is any function $\Phi(\vec{X}; w_1, w_2, \dots, w_{M+1})$ which depends linearly on the parameters. A Φ -function can be written

$$\begin{aligned} \Phi(\vec{X}) &= w_1 f_1(\vec{X}) + w_2 f_2(\vec{X}) + \dots \\ &+ w_M f_M(\vec{X}) + w_{M+1} \end{aligned} \quad (6)$$

where the $f_i(\vec{X})$ are independent of the weights. Since there are $M+1$ weights we shall say that the number of degrees of freedom is equal to $M+1$.

Specific examples of Φ -functions are:

- (1) Linear functions: $f_i(\vec{X}) = x_i$ for $i = 1, \dots, d$.

In this case, $M = d$.

- (2) Quadric functions: $f_i(\vec{X})$ is of the form $x_k^n x_\ell^m$;

$k, \ell = 1, \dots, d$ and $n, m = 0$ and 1 .

In this case, $M = \frac{d(d+3)}{2}$.

- (3) r -th order polynomial functions: $f_i(\vec{X})$ is of the form

$$x_{k_1}^{n_1} x_{k_2}^{n_2} \dots x_{k_r}^{n_r}; \quad k_1, k_2, \dots, k_r = 1, \dots, d, \text{ and}$$

$n_1, n_2, \dots, n_r = 0$ and 1 .

In this case $M = \sum_{i=1}^r \binom{d+i-1}{i}$.

If Φ -functions are used as discriminant functions in a pattern dichotomizer, a wide variety of decision surface families can be achieved. These families, given by equations of the form $\Phi(\vec{X}) = 0$.

depend upon the form of the Φ -function used. That is, they depend upon the $f_i(\vec{X})$, $i = 1, \dots, M$. Examples of Φ -surface families are

- (1) Linear (i.e., hyperplanes)
- (2) Hyperquadrics (e.g., hyperspheres, hyperellipsoids, etc.), and
- (3) r -th order polynomial surfaces.

Let us define the M -dimensional vector

$$\vec{F}(\vec{X}) = \{f_1(\vec{X}), f_2(\vec{X}), \dots, f_M(\vec{X})\} \quad (7)$$

\vec{F} is a vector in an M -dimensional space which we shall call the Φ -space. A Φ -surface in the pattern space has corresponding to it a hyperplane boundary in Φ -space. Because of this fact, many of the results originally obtained in this program for hyperplane surfaces can be easily extended to the whole class of Φ -surfaces. We shall assume here that the mappings $\vec{F}(\vec{X})$ are one-to-one.

We shall call any pattern dichotomizer employing Φ -functions a Φ -machine. A Φ -machine consists of a Φ -processor (which computes \vec{F} from \vec{X}) followed by a TLU. A device for computing Φ -functions is shown in Fig. 4. This device followed by a threshold element would constitute a Φ -machine.

C. THE UTILITY OF Φ -FUNCTIONS FOR CLASSIFYING PATTERNS

The ultimate test of a discriminant function family is the question: how efficient are the members of this family for use in classifying patterns? A specific question of this nature has been formulated and answered for the whole class of Φ -function families.

Suppose we have a set of N patterns. Clearly, there exist a total of 2^N distinct classifications of these patterns into 2 categories (dichotomies); each pattern may independently be assigned to Category 1 or Category 2. One measure of the effectiveness of a discriminant function family would be the total number of dichotomizations of N patterns that its members could effect. We have shown in this program that if the positions of the N pattern points satisfy some quite mild conditions, then the number of dichotomies implementable by a Φ -function family will depend only on the number of patterns, N , and the number of parameters, $M+1$ of the Φ -function family. It does not depend on the configuration of the patterns or on the specific form of the Φ -function family. For details of the derivations see the technical note by Cover.⁸ We shall summarize the results here.

The N patterns are represented by a set, χ , of N points in the d -dimensional pattern space. We assume that these points are in Φ -general position, meaning that no $(M+1)$ or more of these points lie on the same Φ -surface. Examples of points in Φ -general position are the following:

- (1) The Φ -function family is the family of linear functions.
 Φ -general position means that no $(d+1)$ points lie on the same $(d-1)$ -dimensional hyperplane.
- (2) The Φ -function family is the family of quadric functions.
 Φ -general position means that no $\frac{(d+1)(d+2)}{2}$ points lie on the same $(d-1)$ -dimensional hyperquadric surface.

For the members of χ in Φ -general position we have the following expression for the number $\Phi(N,d)$ of dichotomies of χ achievable by any Φ -function family:

$$\Phi(N,d) = 2 \sum_{i=0}^M \binom{N-1}{i} \quad (8)$$

where

N is the number of d -dimensional patterns to be dichotomized,

$M+1$ is the number of degrees of freedom of the Φ -function family, and

$\binom{A}{B}$ is the binomial coefficient $\frac{A!}{(A-B)!B!}$

Thus, a pattern dichotomizer employing a Φ -processor and a TLU with $M+1$ adjustable weights, will be able to achieve $\Phi(N,d)$ of the 2^N dichotomies of N d -dimensional patterns. Some examples illustrating this statement are elaborated on by Cover.⁸

D. MACHINE CAPACITY

Suppose we are given a Φ -machine with $M+1$ adjustable weights and a set, χ , of N patterns in Φ -general position in the pattern space. There are 2^N possible dichotomies of these patterns; if one of these dichotomies is selected at random (with probability 2^{-N}), what is the probability, $P_{N,M}$, that it can be implemented (for some setting of the weights) by the given Φ -machine? The answer is obtained by dividing the number of Φ -dichotomies by 2^N .

$$P_{N,M} = 2^{1-N} \sum_{i=0}^M \binom{N-1}{i} \quad (9)$$

$P_{N,M}$ has a number of interesting characteristics. These can best be seen if we normalize by setting $N = \lambda(M+1)$. A plot of the function $P_{\lambda(M+1),M}$ for various values of M appears in Fig. 5. Note the pronounced threshold effect, for large $(M+1)$, around $\lambda = 2$. Also note that for each value of M

$$P_{2(M+1),M} = 1/2 \quad (10)$$

The threshold effect around $2(M+1)$ can be seen by the expressions

$$\lim_{M \rightarrow \infty} P_{(2+\epsilon)(M+1),M} = 0 \quad \text{for all } \epsilon > 0.$$

and

$$\lim_{M \rightarrow \infty} P_{(2-\epsilon)(M+1),M} = 1 \quad \text{for all } \epsilon > 0. \quad (11)$$

These characteristics of $P_{N,M}$ leads us naturally to define the capacity, C , of a Φ -machine as follows*

$$C = 2(M+1) \quad (12)$$

*

Based on experimental and theoretical results on the number of dichotomies achievable by a TLU, Koford⁹ and Brown¹⁰ had previously suggested that the capacity of a TLU was equal to twice the number of variable weights. Winder¹¹ and Cover⁸ present additional theoretical evidence.

That is, the capacity is twice the number of degrees of freedom or twice the number of weights in the Φ -machine. We can be almost certain of being able to achieve any specific dichotomy of fewer than C patterns with a Φ -machine having $(M+1)$ degrees of freedom, where M is large. On the other hand, we are almost certain to fail to achieve any specific dichotomy of more than C patterns.

The capacity of a Φ -machine is, then, a quite useful measure of its ability to dichotomize patterns. The capacities of some specific Φ -machines are listed in Table 1.

Table 1
THE CAPACITIES OF SOME Φ -MACHINES

Decision Boundary in Pattern Space Implemented by Φ -Machine	Capacity
Hyperplane	$2(d+1)$
Hypersphere	$2(d+2)$
General Quadric Surface	$(d+1)(d+2)$
r -th Order Polynomial Surface	$2 \sum_{i=0}^r \binom{d+i-1}{i}$

E. PIECEWISE LINEAR DISCRIMINANT FUNCTIONS

Another important family of discriminant functions is the one we shall call the piecewise linear discriminant function family. A restricted set of discriminant functions belonging to this family is

employed by the MINOS II trainable pattern classifying machine constructed at the Stanford Research Institute for the U.S. Army Electronic Research and Development Laboratories at Fort Monmouth, New Jersey.¹² The piecewise linear discriminant function family is important because rather complex families of decision surfaces result from a relatively small number of variable parameters.

Consider the general family of discriminant functions of the form

$$g_i(\vec{X}) = \max_{j=1, \dots, L_i} \{g_i^{(j)}(\vec{X})\} \quad (13)$$

where each $g_i^{(j)}(\vec{X})$, called a subsidiary discriminant function, is a linear function. That is, each $g_i^{(j)}(\vec{X})$ is given by

$$g_i^{(j)}(\vec{X}) = w_{i1}^{(j)}x_1 + \dots + w_{id}^{(j)}x_d + w_{i,d+1} \quad (14)$$

where the weights $w_{ik}^{(j)}$ $i = 1, 2; j = 1, \dots, L_i; \text{ and } k = 1, \dots, d+1;$ are the parameters of the family. Here, we again adopt the convention of deciding in favor of Category 1 if $g_1(\vec{X}) > g_2(\vec{X})$, etc.

Since $g_1(\vec{X})$ and $g_2(\vec{X})$ as defined by Eq. (13) are piecewise linear functions of the components of \vec{X} we shall call them piecewise linear discriminant functions.*

*

Our piecewise linear discriminant functions are not completely general piecewise linear functions since Eq. (13) constrains them to be convex.

Any dichotomizer employing piecewise linear discriminant functions shall be called a piecewise linear machine. The structure shown in Fig. 6 is an implementation for a piecewise linear discriminator. The subsidiary discriminators are organized into two banks. If, for any pattern, \vec{X} , the i -th bank ($i = 1, 2$) contains the largest subsidiary discriminant, then \vec{X} is placed in Category i . The decision surfaces for piecewise linear machines consist of sections of hyperplanes.

In this program we have considered the properties of a special variety of piecewise linear machine consisting of a layered network of TLUs. The results of this research will be summarized in a subsequent section of this report. Except for special cases, the number of dichotomies of N d -dimensional patterns achievable by a given piecewise linear machine is still unknown. For that reason, the capacity of piecewise linear machines is also unknown, however, digital computer simulations indicate that it is of the order of twice the number of variable weights.

III PROPERTIES OF NETWORK STRUCTURES FOR THE IMPLEMENTATION OF PIECEWISE-LINEAR DISCRIMINANT FUNCTIONS

A. LAYERED MACHINES

Pattern classifying machines have often been proposed which consist of networks of interconnected TLUs. Figure 7 is an example of a TLU network in which the responses of some TLUs are used as inputs to the others. If $R = 2$, the output of one of the TLUs is taken to be the response of the whole machine.

An important type of TLU network is one in which the TLU's are arranged in layers. In this program we have studied pattern dichotomizers consisting of two layers of TLUs. The first layer consists of a number of TLUs each of which has as its input the components of the pattern vector, \vec{X} . A single TLU in the second layer has as its inputs the outputs of the first layer TLUs. The response of this second layer TLU is taken to be the response of the machine. Such a pattern dichotomizer is illustrated in Fig. 8. It is a straightforward matter to verify that the discriminant function of this type of layered machine is piecewise linear. In one form or another this structure has been well-discussed in the literature.^{2,3,12} In this section we shall summarize some of the new results obtained in this program.

B. A THEOREM ABOUT TWO-LAYER MACHINES

The following theorem about two-layer machines was proved during this program. In this theorem we restrict our attention to binary patterns.

Let χ be a set of vertices of the unit d -dimensional cube. These vertices are given by binary d -tuples with $(0,1)$ components. χ is partitioned into the two subsets χ_1 and χ_2 . We are given K hyperplanes which do not intersect within the unit hypercube such that for each pair of vertices belonging to different subsets there is at least one hyperplane separating them. Furthermore, we assume that this separation property could not be met with fewer than K hyperplanes. These K hyperplanes are implemented by K TLU's whose $(0,1)$ binary responses for each pattern vector can be represented by a K -dimensional vector, \vec{Y} . Thus, these K TLU's map the sets χ_1 and χ_2 into image sets which we shall call \mathcal{Y}_1 and \mathcal{Y}_2 , respectively.

Theorem:

\mathcal{Y}_1 and \mathcal{Y}_2 are linearly separable. That is, a single TLU in the second layer can complete the dichotomy given by the partition of χ into χ_1 and χ_2 .

Proof:

In sketching the proof we shall first show that there are exactly $K+1$ distinct vectors in the union, \mathcal{Y} , of \mathcal{Y}_1 and \mathcal{Y}_2 . Then we shall show that the matrix whose columns are the vectors in \mathcal{Y} has rank K , which will permit us to observe that \mathcal{Y}_1 and \mathcal{Y}_2 are linearly separable.

The K hyperplanes partition the hypercube into exactly $K+1$ regions if the hyperplanes do not intersect within the hypercube. Corresponding to each of these regions is an element of \mathcal{Y} ; therefore \mathcal{Y} consists of $K+1$ vectors.

Let \underline{M} be the matrix whose columns are the vectors in \mathcal{V} . We can assume, without loss of generality that the zero vector is one of the members of \mathcal{V} . The columns of \underline{M} are, of course, all distinct. Assume that \underline{M} does not have rank K and thus has a row, say the k -th, which is a linear combination of the other rows. If we delete this k -th row the columns of the reduced matrix, \underline{M}^* , are still distinct. (For two columns to be identical after deletion of the k -th row means that these columns could have differed only in the k -th row: a contradiction of the assumption that the k -th row was a linear combination of the others.) We may interpret \underline{M}^* as a matrix of vectors produced by $(K-1)$ non-intersecting planes. But under these conditions, \underline{M}^* could have only K distinct columns thus contradicting the assumption that the rank of \underline{M} is not equal to K . Therefore, the rank of \underline{M} is equal to K .

Since the rank of \underline{M} is equal to K , we can solve the equations

$$\vec{Y}_i \cdot \vec{U} = d_i \quad i = 1, \dots, K$$

where the \vec{Y}_i are the non-zero vectors in \mathcal{V} and the d_i are arbitrary numbers. We have only to select the sign of each d_i to agree with the desired output of the second layer TLU. The fact that a vector, \vec{U} , exists for any specification of the d_i proves that \mathcal{V}_1 and \mathcal{V}_2 are linearly separable.

C. COMMITTEE MACHINES

A committee machine is another name given to a two-layer machine as shown in Fig. 8. It is called a committee machine for the following reasons: the TLU's in the first layer can be considered to be a committee of TLUs that votes on patterns. Each individual TLU, called a committee member, votes either +1 or -1. These votes are weighted by the weights of the second-layer TLU which then announces the final decision: either +1 or -1. If there are K TLUs in the first layer, the committee is said to be of size K .

In this program we have considered only the case in which K is odd and the second layer TLU has weights, w_1, w_2, \dots, w_K , each equal to +1. We also assume that the $(K+1)$ -th weight of the second layer TLU is equal to zero. In this case, each of the first-layer TLU's has an equal vote strength, and the second layer TLU merely announces the majority decision. Such a specialized committee machine has been called a "majority-logic" machine at SRI;¹² it is identical with one form of Madaline machine proposed by Widrow.³

Some of the properties of majority logic machines can be more easily understood after we have introduced a special geometric representation for the TLU. We shall discuss this representation now and then return to our discussion of majority logic machines.

Suppose the TLU has d weights,* w_1, w_2, \dots, w_d . This set of weights can be presented by a point in a d -dimensional weight-space. The rectangular coordinates of the point are given by the weight values. The

*

In this section we shall assume, for simplicity, that $w_{d+1} = 0$. This assumption constrains the TLU decision surface to pass through the origin of pattern space. This constraint can be lifted by repeating the following development in a space of $(d+1)$ dimensions.

d-dimensional vector, \vec{W} , with components w_1, w_2, \dots, w_d , extending from the origin to this point, can also be used to represent the set of TLU weight values. We shall use the symbol, \vec{W} , to denote both the weight vector and the weight point.

A linear discriminant function can now be written in the simple form:

$$g(\vec{X}) = \vec{X} \cdot \vec{W} \quad (15)$$

For any pattern, \vec{X} , there exists a hyperplane in weight space which is the locus of all weight vectors for which

$$\vec{X} \cdot \vec{W} = 0 \quad (16)$$

The hyperplane in weight space defined by Eq. (16) for a given pattern vector is called the pattern hyperplane. This hyperplane separates the space of weight points into two classes; those which for the pattern, \vec{X} , produce a TLU response of one are on one side of the hyperplane, called the positive side, and those which produce a TLU response of minus one are on the other, or negative side. Note that the point representing the weight values ($w_1 = 0, w_2 = 0, \dots, w_d = 0$) satisfies Eq. (16) regardless of \vec{X} . Therefore all pattern hyperplanes pass through the origin of weight space.

Corresponding to a finite set χ of N patterns there will exist a set of pattern hyperplanes which divide weight space into a number of different regions. The response of a TLU to any pattern depends upon

which side of the pattern hyperplane the TLU weight point, \vec{W} resides. As \vec{W} is varied, the TLU response to some of the patterns will change if \vec{W} crosses any of the pattern hyperplanes and thus leaves a region. Therefore, each region in weight space corresponds to a different linear dichotomy of the N patterns, and conversely, a dichotomy of the patterns in X is a linear dichotomy if and only if there is a region in weight space corresponding to it.*

These ideas are illustrated in Fig. 9 for a two-dimensional weight space. In this example, the small arrows attached to the pattern hyperplanes (lines) indicate the positive side of each hyperplane. The encircled numbers attached to the hyperplanes indicate the number of the pattern. Thus the weight point \vec{W} , or any weight point in the shaded region represents a set of TLU weights which would cause the TLU to respond with a +1 to patterns number one, two, and three.

We have stated that a dichotomy of N patterns is linear if there exists a region in weight space corresponding to it. If a dichotomy is not linear, a single TLU cannot achieve it. Now that we have introduced the weight space representation, it is easy to state the conditions under which a majority logic machine can implement a given dichotomy.

A committee of size K , voting by majority rule, can implement a given dichotomy if and only if a set of K weight points can be found such that the majority of the weight points is on the correct side of every pattern hyperplane.

*Here, our definition of a linear dichotomy has been restricted to one which can be achieved by a hyperplane decision surface passing through the origin of pattern space.

Consider the example of Fig. 10. There exists no single weight point on the positive side of each of the pattern hyperplanes number one, two, and three. Yet the majority of the "committee" of weight points, \vec{w}_1 , \vec{w}_2 , and \vec{w}_3 is on the positive side of each of these hyperplanes. Thus, a committee of size 3 could give a positive response to each of these patterns.

Some interesting properties of majority logic machines have been derived during this program. Many of these properties were reported in a technical note by Kaylor.¹³ Some recent results, not previously reported, are included in Appendix A of this report. We shall now summarize some of the more important of these properties.

D. A SUMMARY OF RESULTS ON MAJORITY LOGIC MACHINES

(1) It has been shown in reference 13 and in Appendix A of this report, that for any dichotomy of N d -dimensional patterns, there exists a majority logic machine of finite size, K , that can implement the dichotomy. Such a machine is called a solution machine. K never needs to be larger than N if N is odd or $N-1$ if N is even. There do exist dichotomies requiring committees of these sizes for $d = 2$, but, in general, tighter bounds on committee sizes are not well understood for $d > 2$.

(2) We define a simple region as a region in weight space bounded by pattern hyperplanes but not intersected by any pattern hyperplanes. A avored region is a simple region on the positive side of each of the pattern hyperplanes bounding it. We have the following results on favored regions (the proofs are to be found in reference 13).

- (a) In a majority logic machine each committee weight point may reside in a favored region.
- (b) There is exactly one favored region if and only if there is a solution majority logic machine of size $K = 1$.

(3) The capacity of a majority logic machine of given size is not known in general. We do have a result for the case of $d = 2$ which shall be derived here.

If there are a total of N two-dimensional patterns, then there are 2^N ways in which these patterns can be dichotomized.

We desire to know the number, $C_{N,2}^{(K)}$ of these dichotomies which can be implemented by a K -member committee voting by majority logic.

Each pattern $\{x_1, x_2\}$ is represented by a line (the two-dimensional equivalent of a pattern hyperplane) passing through the origin and normal to the vector from the origin to the point (x_1, x_2) . We assume that no two patterns are represented by lines with the same slope. The dichotomy of the patterns is represented by designating one side of each line "positive".

For two-dimensional patterns the number of committee members needed to implement a given dichotomy of N patterns is equal to the number of favored regions formed by the set of polarized pattern lines.¹³ The number of dichotomies of N two-dimensional patterns implementable by a K -member committee is therefore equal to the number of ways in which N distinct lines can be polarized to yield K or fewer favored regions.

For any two-dimensional diagram of N polarized lines write a sequence of N pluses and minuses using the following rule.

Draw a unit circle centered on the origin, begin at any point on this circle and proceed clockwise around the circle until exactly N lines have been traversed. Every time a line is traversed in its positive direction add a "+" to the sequence; every time a line is traversed in the negative direction add a "-" to the sequence.

The result will be a sequence such as $\underbrace{+ + - \dots + +}_N$.

Suppose there are i sign reversals in the sequence. It can easily be shown that the number of favored regions is equal to i if i is odd and equal to $i + 1$ if i is even. Note that the number of favored regions is thus always odd.

The number of distinct N -length sequences with exactly i sign reversals is equal to $2 \binom{N-1}{i}$. The number of N -length sequences with K or fewer sign reversals is therefore equal to $2 \sum_{i=0}^K \binom{N-1}{i}$. But this quantity is exactly equal to the number of dichotomies yielding K or fewer favored regions and is therefore equal to the number of dichotomies implementable by a K -member committee voting by majority logic. That is,

$$C_{N,2}^{(K)} = 2 \sum_{i=0}^K \binom{N-1}{i} \quad \text{for } K \leq N-1 \quad (17)$$

$K \text{ odd}$

If a dichotomization is selected at random according to a uniform probability law over all possible 2^N dichotomizations, then the probability, $P_{N,2}^{(K)}$ that this dichotomization is implementable by a K -member committee is given by the following expression:

$$p_{N,2}^{(K)} = \frac{C_{N,2}^{(K)}}{2^N} = \frac{1}{2^{N-1}} \sum_{i=0}^K \binom{N-1}{i} \quad (18)$$

Therefore, in analogy with Eqs. (9) and (12), the capacity of a two-dimensional majority machine, of size K , is equal to $2(K+1)$.

(4) The general problem of the optimum placement of committee weight points for any given dichotomy is as yet unsolved, but some work conducted in this program on cycles suggests a sub-optimum procedure. This work is reported in detail in Appendix A, and we shall only summarize it briefly here.

To form a cycle of regions in weight space we begin in any region and trace a path through weight space by crossing every plane exactly once. The regions containing the path and their reflections in the origin constitute a cycle. There are many different possible cycles, but any cycle can be used to place committee members.

Each region of the cycle has, in general, many boundary pattern hyperplanes, but for each region only two of the boundary hyperplanes are crossed by the cycle path. If the positive directions of both of these two hyperplanes point into the region, the region will be called positive with respect to the cycle. We have the following major result for the general d -dimensional situation:

A majority logic machine for any dichotomy can always be obtained by selecting any cycle and placing one committee member in each region which is positive with respect to the cycle.

Thus, there are many solutions that can be obtained in this way; one for each possible cycle. Some of these solutions might be identical, and some may be optimum. Future work in this area could provide some interesting and important results.

E. AN EXAMPLE DICHOTOMY ILLUSTRATING THE USE OF MAJORITY LOGIC
MACHINES

In Appendix B it is proved that the complete parity function in d -dimensions requires a majority logic machine of size $K = d$ if d is odd and size $K = d+1$ if d is even. By the parity function we mean that function of binary $(0,1)$ d -tuples used to place the d -tuples into one category if the number of ones in the d -tuple is even and to place it into the other category otherwise. By a complete parity function we mean that the complete set of 2^d d -tuples is to be classified by this rule.

The parity function has proven to be a useful one as a test problem for majority logic machines because it is one which has a solution of known size and character.

IV TRAINING METHODS FOR THE SELECTION OF DISCRIMINANT FUNCTIONS

A. ERROR CORRECTION TRAINING METHODS

If a dichotomy of a set, X , of pattern vectors is linearly separable, various procedures exist for specifying the set of weight values (called the solution weight vector) of the TLU which will implement the dichotomy. These procedures, called error correction methods, have the following characteristics:

- (1) The pattern vectors are presented to an adaptive TLU one at a time (in any sequence in which each of them occurs infinitely often) to determine the TLU response.
- (2) If the TLU response to a pattern is incorrect, an adjustment (adaptation) is immediately made in the weight values. Otherwise, the weight values are left unchanged.

Two error correction training methods that have been proven to be effective are the following:

(1) Motzkin-Schoenberg Procedure¹⁴

If the weight values are to be adjusted, they are adjusted according to the following rule:

$$\vec{W}' = \vec{W} - \lambda \frac{\vec{W} \cdot \vec{X}}{\vec{X} \cdot \vec{X}} \vec{X}$$

where

\vec{W}' = New weight vector

\vec{W} = Old weight vector

\vec{X} = Pattern vector inaccurately categorized
by TLU with weights given by \vec{W} .

For $0 < \lambda < 2$, this rule produces a sequence of weight vectors that converge to a point on the boundary of the region of solution weight vectors unless one member of the sequence is itself a solution weight vector, in which case the sequence then terminates. For $\lambda = 2$, the sequence of weight vectors terminates at a solution.

(2) Rosenblatt-Widrow Procedure^{2,3}

The following weight vector adjustment is made

$$\vec{W}' = \vec{W} - K \frac{\vec{W} \cdot \vec{X}}{|\vec{W} \cdot \vec{X}|} \vec{X}$$

where K is any positive constant. This rule is guaranteed to produce a solution weight vector (when one exists) in at most a finite number of steps. A short proof of the convergence of this rule has been given by Novikoff.¹⁵

In some applications of the error correction training rules the components of the pattern vectors are binary numbers. In these situations it is important to ask whether these training methods converge faster for the (1,0) mode of representation or for the (1,-1) mode. It was determined by experimental and theoretical investigations that the (1,-1) mode generally leads to faster convergence. The details supporting these conclusions were reported during this program in a technical note by Nilsson and Singleton.¹⁶

In other studies conducted during the program it was determined that the length of the TLU weight vector during the Rosenblatt-Widrow training procedure remains bounded even when the training set, χ , is not linearly separable. The proof of the boundedness of the weight

vector is contained in a technical note by Efron.¹⁷ Since this proof is rather long and involved we shall present a sketch of it here.*

Theorem 1 of Efron's paper shows that the sequence $\{\vec{w}_j\}$ of weight vectors resulting from applying the Rosenblatt-Widrow training rule remains bounded in length, whatever the initial weight vector \vec{w}_0 and training sequence $\{\vec{x}_j\}$. Although it is possible to find initial weight vectors of arbitrary length that can be increased in length for one or more correction steps, there exists a bound T such that if $\|\vec{w}_0\| > T$, $\|\vec{w}_k\| < \|\vec{w}_0\| + 1$ for all k for any training sequence of finite length. Thus a weight vector that is already long can increase in length by only a small amount.

The proof is by induction. The result is true for dimension $d = 1$, and is assumed true for dimension up to $(d-1)$ but false for dimension d . The assumption that the theorem is false for dimension d is then shown to lead to a contradiction, and thus the theorem is true for arbitrary dimension.

Under the assumption that the theorem is false for dimension d , there exists an unbounded sequence $\{\vec{w}_0^j\}$ of initial vectors of increasing length, with training sequences $\{\vec{w}_k^j\}$ such that $\|\vec{w}_k^j\| \geq \|\vec{w}_0^j\|$ for $k = 0, 1, \dots, K_j$ and $\|\vec{w}_{K_j}^j\| - \|\vec{w}_0^j\| \geq 1$ for each j . Considering direction, the sequence of initial weight vectors must have at least one accumulation point on the unit sphere, say \vec{v} , and thus must contain a sub-sequence of initial weight vectors converging

* This sketch of Efron's proof is due to R. C. Singleton.

in direction to that of \vec{V} . In the final steps of the proof only members of this sub-sequence are considered.

The set χ of possible pattern vectors is finite, and can be divided into three classes with respect to \vec{V} :

$$C = \{\vec{X} | \vec{X} \cdot \vec{V} < 0\}$$

$$D = \{\vec{X} | \vec{X} \cdot \vec{V} = 0\}$$

$$E = \{\vec{X} | \vec{X} \cdot \vec{V} > 0\}$$

For \vec{W}_0^j sufficiently long and close in angle to \vec{V} , the initial correction must be for a pattern in the set D , for otherwise

$\|\vec{W}_1^j\| < \|\vec{W}_0^j\|$. The set D has dimension $(d-1)$ at most, and thus the component of \vec{W}_k^j in the subspace spanned by D will be of bounded length after any finite number of corrections for patterns within D .

The increase $\|\vec{W}_k^j\| - \|\vec{W}_0^j\|$ in weight vector length and increase in angle with \vec{V} can be made as small as desired by choice of \vec{W}_0^j .

In order to satisfy the condition $\|\vec{W}_{K_j}^j\| - \|\vec{W}_0^j\| \geq 1$, some pattern outside of D must eventually be corrected for. But for \vec{W}_0^j sufficiently long and close in angle to \vec{V} , the next pattern corrected for after initial corrections within D must come from C , and the weight may be decreased in length by any single correction from C by more than it can be increased by a finite number of corrections for patterns in D . Thus the conclusion $\|\vec{W}_k^j\| \geq \|\vec{W}_0^j\|$ for $k = 0, 1, \dots, K_j$ is contradicted.

B. GENERALIZATION

Whatever method is employed to train a TLU on a given set, χ , of training patterns, there is always the question: how well will the trained TLU perform on related patterns not contained in the

the training set? Questions of this type refer to the capability of a trained TLU to make generalizations. In the technical note by Cover⁸ a question of this type was discussed. We shall summarize his results together with some previously unreported ones.

Cover discusses generalization capabilities as follows. Suppose a set, χ , consisting of N d -dimensional points has been correctly dichotomized by a hyperplane. We now add one more point, \vec{x}_{N+1} , and ask: what is the probability, $F(N,d)$, (for random dichotomizations of χ) that the category of \vec{x}_{N+1} is ambiguous? Using the fact that the category of \vec{x}_{N+1} is ambiguous if χ can be correctly dichotomized by a hyperplane passing through \vec{x}_{N+1} , we obtain

$$F(N,d) = \frac{C_{N,d-1}}{C_{N,d}} = \frac{\sum_{i=0}^{d-1} \binom{N-1}{i}}{d \sum_{i=0}^{N-1} \binom{N-1}{i}} \quad (19)$$

$F(N,d)$ is a measure of how well trained a TLU is after it can correctly dichotomize N d -dimensional patterns. The smaller $F(N,d)$, the more highly trained is the TLU and the better is its capability for making correct generalizations.

Cover observed that, for $\frac{N}{d} \gg 1$

$$F(N,d) \approx \frac{d}{N-d} \quad (20)$$

A new result due to B. Elspas states that this approximation is also valid for any $\frac{N}{d} \geq 2$ so long as d is very large. This result is derived as follows:

write $F(N,d)$ as

$$F(N,d) = \frac{\sum_{i=0}^d \binom{N-1}{i} - \binom{N-1}{d}}{\sum_{i=0}^d \binom{N-1}{i}} = 1 - \frac{1}{G(N,d)} \quad (21)$$

where

$$G(N,d) = \sum_{i=0}^d \binom{N-1}{i} / \binom{N-1}{d} \quad (22)$$

We observe that $G(N,d)$ satisfies the recursion:

$$\begin{aligned} G(N,d) &= 1 + \frac{d}{N-d} G(N,d-1) \\ &= 1 + \frac{1}{\beta-1} G(N,d-1) \end{aligned} \quad (23)$$

where $\beta = N/d$. For $\beta-1 > 1$, this recursion clearly yields a finite limiting value, $G^*(\beta)$, for the function, $\lim_{d \rightarrow \infty} G(\beta d, d)$. In fact,

$$G^*(\beta) = 1 + \frac{1}{\beta-1} G^*(\beta) \quad (24)$$

yields

$$G^*(\beta) = (\beta-1)/(\beta-2), \text{ for } \beta > 2 \quad (25)$$

Hence,

$$F(N,d) = 1 - 1/G^* = 1/(\beta-1) \text{ for } \beta > 2 \quad (26)$$

While for $\beta \leq 2$, $\lim_{d \rightarrow \infty} G(\beta d, d) = \infty$, since $G(2d, d) \rightarrow \infty$ with $d \rightarrow \infty$; and for any fixed d , $G(\beta d, d)$ is monotonic decreasing with β in the range $1 < \beta \leq 2$. Hence

$$\begin{aligned} G^*(\beta) &= \infty && \text{for } 1 < \beta \leq 2 \\ &= \frac{\beta-1}{\beta-2} && \text{for } \beta \geq 2 \end{aligned} \quad (27)$$

and consequently,

$$\begin{aligned} F(N, d) &\rightarrow \frac{1}{\frac{N}{d} - 1} && \text{when } \frac{N}{d} \geq 2 \\ &\rightarrow 1 && \text{when } \frac{N}{d} \leq 2 \end{aligned} \quad (28)$$

The shape of the asymptotic curve for $F(N, d)$ is shown in Fig. 11.

The following series expansion for $G(N, d)$, due to M. W. Green may perhaps be of interest in that it gives some idea of how $G(N, d)$ --and hence also $F(N, d)$ --approaches its asymptotic form.

Put

$$G(N, d) = a_0 + a_1 d + a_2 d^2 + \dots \quad (29)$$

in the recursion

$$G(N, d) = 1 + \frac{d}{N-d} G(N, d-1) \quad (30)$$

and write

$$G(N, d-1) = G(N, d) - G'(N, d) \quad (31)$$

where G' is the derivative with respect to d . Equating coefficients of like powers of d one finds that

$$\begin{aligned} G(N, d) = 1 + \frac{d}{N+1} + \frac{2d^2}{(N+1)(N+2)} \\ + \frac{4d^3}{(N+1)(N+2)(N+3)} + \dots \quad (32) \end{aligned}$$

C. APPLICATIONS TO Φ -MACHINES

Since a Φ -machine consists of a Φ -processor followed by a TLU, the error correction methods can be applied to the training of Φ -machines. The Φ -processor, determined by the Φ -function family being used, remains fixed during training; only the weights of the TLU are adjusted, and those are adjusted exactly as we have described above. If the Φ -processor converts the input pattern vectors by the mapping $\vec{F} = \vec{F}(\vec{X})$, we replace \vec{X} by \vec{F} in the foregoing discussion of error correction training methods. Thus, adjusting the location of a hyperplane surface in the Φ -space, at the same time adjusts a Φ -surface in the pattern space. Therefore, any Φ -machine can be trained to dichotomize a set of training patterns if it is possible to dichotomize this set by some member of the same family

of Φ -machines. Since Φ -surface families constitute an important class (including linear, quadric, r -th order polynomial, etc.) this conclusion is of great importance.

D. THE TRAINING PROBLEM FOR MAJORITY LOGIC AND OTHER PIECEWISE LINEAR MACHINES

At present there do not exist any methods for training committee machines analogous to the error correction methods for training the TLU that are guaranteed to converge. There does exist, however, a training procedure for the majority logic machine that has been found satisfactory in a variety of different experiments. This method, originally proposed by Ridgway,¹⁸ is also discussed in the technical note by Kaylor.¹³ It is known that there are situations for which Ridgway's method does not converge when a solution does exist. Efforts during this program to find similar training procedures, for which convergence theorems could be proven, were not successful.

Rather than directing further effort toward finding convergent procedures for committee machines, it might be wiser to consider the problem of finding training methods for general piecewise linear machines as we have defined them in Sect. II-E. We shall suggest a possible training method here in the hope that it might stimulate further research on this topic. The method to be suggested has not been tested experimentally; neither have conditions been discovered under which it might converge to a solution when one exists.

Suppose we are given a piecewise linear machine which we wish to train to dichotomize correctly, the members of a training set,

χ (see Fig. 6). The only constraint is that the total number of subsidiary discriminant functions be equal to L . That is, the training problems is (1) to specify L subsidiary decision functions, and (2) to divide them up into 2 banks in such a way that when a pattern \vec{X} belonging to Category i is presented to the machine, the i -th bank contains the highest-valued function ($i = 1, 2$).

We desire a training procedure analogous to the error correction procedures. Such a training procedure implies a process in which the L subsidiary functions are being iteratively adjusted and simultaneously reshuffled among the various banks.

These two aspects of training piecewise linear machines have stubbornly resisted satisfactory solutions. In the case of layered machines, which are special cases of piecewise linear machines, these dual aspects of training have the following significances: adjustment of the subsidiary discriminant functions corresponds to modifying the weight values of the first layer of TLUs. The reshuffling of the subsidiary functions between the two banks corresponds to changing the weight values of the second layer TLU. The fact that no satisfactory methods have yet been proposed for simultaneously training two or more layers of TLUs in a layered machine reflects the difficulty of the general problem of training piecewise linear machines.

A somewhat simpler training problem is presented if the initial distribution of the subsidiary functions into two banks is left unaltered during training. Suppose we have L linear discriminant functions, L_1 of which are in the first bank and L_2 of which are in the second bank. We shall not transfer any subsidiary functions

from one bank to another during training. The training problem in this case involves only the adjustment of the subsidiary discriminant functions, no reshuffling. The simplified but less powerful training method corresponds, in a committee machine, to adjusting the weight values of the TLUs in the first layer only.

The following procedure is suggested as a training method for adjusting the subsidiary discriminant functions while leaving their distribution fixed. After presenting a pattern which the machine classifies correctly we make no changes in the values of the weights used to implement the subsidiary discriminant functions. Suppose, however, that a pattern, \vec{X} , belonging to Category i causes an incorrect response. Such would be the case if the j -th bank, $j \neq i$, contained that subsidiary discriminant function whose value, evaluated for \vec{X} , was largest. The suggested adjustment method first subtracts \vec{X} from the weight vector used by this subsidiary discriminant function in the j -th bank. Secondly, of those subsidiary discriminant functions in the i -th bank we determine which has the largest value for \vec{X} . The corresponding weight vector is adjusted by the addition of \vec{X} . This process is continued until convergence is reached.

V CONCLUSIONS

In this report we have suggested that the problem of the selection of discriminant functions constitutes the primary problem in the design of pattern classifying machines. Several classes of discriminant functions and some means of implementing them have been discussed. We have also presented some training methods for the selection of discriminant functions.

The accomplishments made during this program are two-fold. First, we have organized the new and the existing material on trainable pattern classifying machines around the framework provided by the notion of discriminant functions to provide a basis for further theoretical development. Second, we have contributed some specific new results to the theory. These new results include those on

- (1) the capacity and generalization capabilities of TLUs
- (2) Φ -functions and their use as discriminant functions
- (3) the comparison of the (1,0) versus the (1,-1) mode for pattern presentation
- (4) the boundedness of the length of the TLU weight vector during training
- (5) majority logic and committee machines, and
- (6) feature detection.

In the course of organizing past and present results several new questions arise. Each of these questions could serve as the basis for further research in this area. Their answers would form an integral part of a well-developed theory of trainable pattern classifying machines. Some of these questions are the following:

- (1) For any given application how should the measurements be selected to obtain suitable inputs to the pattern classifying machine?
- (2) How do the past and present results generalize to the case $R > 2$.
- (3) What are the capacities of piecewise linear machines in general and committee and majority logic machines in particular?
- (4) What are good training methods for piecewise linear machines in general and committee and majority logic machines in particular?
- (5) How should a pattern classifying machine be trained if the given dichotomy of the training set is known to be one not implementable by the given machine?
- (6) What bases are there for selecting one family of discriminant functions over any other for use in a pattern classifying machine for a specific application?

It is hoped that future work will provide the answers to these and related questions.

APPENDIX A

**GEOMETRICAL CONSTRUCTION
OF A MAJORITY
RULE SOLUTION COMMITTEE**

by

D. J. Kaylor

C. M. Ablow

APPENDIX A

GEOMETRICAL CONSTRUCTION OF A MAJORITY RULE SOLUTION COMMITTEE

The task is to find a majority rule solution committee for the classification of each of a number M of $(N-1)$ -dimensional pattern vectors into one of two categories. We interpret the problem in N -dimensional Euclidean space and construct a solution committee geometrically.

The pattern vector x^j determines an $(N-1)$ -dimensional plane $\pi_j = \{w: \sum_{i=1}^N x_i^j w_i = 0\}$ normal to x^j and passing through the origin.¹³ We assume that no two of the vectors x^j are equal. The halfspace $H_j = \{w: \sum_{i=1}^N x_i^j w_i > 0\}$ containing x^j is called the positive side of π_j . A majority rule solution committee consists of K points w^k in E^N with K odd located so that a majority of them lie on the positive side of every π_j , $j = 1, \dots, M$.

A region R is a polyhedral cone bounded by some of the planes π_j such that none of the planes intersects it [Sect. IV, Ref. 13]. We say a plane separates two regions if the regions are on opposite sides of the plane. The reflection in the origin of a region R is a region since the planes π_j pass through the origin, and the reflection of π_j is π_j . All of the planes separate the regions R and $-R$.

Two regions are neighbors if exactly one plane separates them. A boundary of a region is a plane separating the region from a neighbor.

Two distinct regions are separated by one or more boundaries of each region. For, let S be the set of boundary planes of region R_1 , and let R_2 be a distinct region. From the complete collection of M planes delete all planes not in S . Region R_1 is unaffected since its boundaries are unchanged. Region R_2 is perhaps enlarged to a region R_2' . R_2' is distinct from R_1 , and therefore separated from it by a member of S . But R_2 is contained in R_2' and so is separated from R_1 by a boundary of R_1 .

A favoured region is a region that is on the positive side of each of its boundaries. In a solution committee governed by majority rule, each committee member may reside in a favoured region [Thm. 3, Sect. IV, Ref. 13]. Suppose a committee member is in a not-favoured region R . Then R is on the negative side of at least one of its boundary planes. If π is such a plane, and we move the committee member across π , it now lies on the positive side of π and on the same side as before of all the other planes. Even before the move, a majority of the committee members were on the positive side of π ; the only change after the move is an additional member on the positive side of π .

The M distinct oriented planes π_j through the origin in general position in E^N determine Q regions, where $Q = 2 \sum_{i=0}^{N-1} \binom{M-1}{i}$ [Ref. 8]. We label these regions according to the orientation of the planes in the following way: the region R_k is assigned the M -tuple $(a_{1k}, a_{2k}, \dots, a_{Mk})$, where

$$a_{jk} = \begin{cases} +1 & \text{if } R_k \text{ lies on the positive side of } \pi_j \\ -1 & \text{if } R_k \text{ lies on the negative side of } \pi_j \end{cases} \quad (1)$$

Then we may represent the entire configuration of regions by an $M \times Q$ matrix $A = (a_{jk})$, with a_{jk} as in Eq. (1). The k -th column of A corresponds to the region R_k . We shall call the k -th column R_k or a_k .

Two matrices are equivalent if they represent the same geometric configuration of planes. Thus matrices obtained from a given matrix by row or column permutations are equivalent since such permutations merely correspond to a renumbering of the planes or regions. If the geometric configuration is at hand we may construct its matrix representation. The converse problem of determining whether a given matrix can represent a geometric configuration is open.

The committee problem may be restated as follows: Find a Q element column vector $C = (c_k)$ with non-negative entries such that every component of AC is greater than zero. Since the committee is governed by majority rule, $c_k = u$ if u committee members reside in R_k and $c_k = 0$ if no committee member resides in R_k .

We shall discuss the configuration of M planes forming Q regions in terms of the matrix A . No two columns of A are identical, since two distinct regions are separated by at least one plane π_j , and thus differ in the j -th coordinate. If the M -tuple a_k is a column of A , so is the M -tuple $-a_k$, since all M of the planes separate the region $-R_k$ from the region R_k . Two columns are neighbors with the common boundary π_j if they differ only in the j -th coordinate. That is, R_p and R_q are neighbors with common boundary π_j if

$$a_{ip} = a_{iq}, \quad i \neq j, \quad i = 1, 2, \dots, M$$

$$a_{jp} = -a_{jq}$$

A chain of length $n+1$ connecting two regions R_p and R_q is a set of regions $R_0 = R_p, R_1, R_2, \dots, R_n = R_q$ such that R_m and R_{m+1} are neighbors, $m = 0, 1, \dots, n-1$. The chain is said to cross the plane π_j if two neighbors in the chain are separated by π_j . There exists a chain connecting any two regions. [Intuitively, the regions fill up space, and the planes do not coincide, so we can step from one region to another crossing only one plane at a time.]

We prove the following theorem:

Theorem 1:

If two regions are separated by n planes, there is a chain of length $(n+1)$ connecting them.

Proof:

The proof is by induction on n . If $n = 1$, the two regions are neighbors forming a trivial chain of length two.

Let R_p and R_q be separated by a set S of n planes. At least one member of S is a boundary of R_p . For, if not, R_q would be on the same side of each boundary of R_p as R_p itself, and so would coincide with R_p . Choose such a boundary plane π_h of R_p that is in S , and let the neighbor of R_p across π_h be R_h . Let S_1 be the set S with π_h deleted. Region R_h is separated from R_q by the $(n-1)$ planes belonging to S_1 since R_h is on the same side of every plane as R_p , except for π_h . By the induction

hypothesis, there is a chain of length n connecting R_h and R_q ; this chain plus the initial region R_p is a chain of length $(n+1)$ connecting the regions R_p and R_q .

A cycle is a chain of neighbors containing exactly M regions and their reflections.

Theorem 2:

Any two regions can be joined by a cycle.

Proof:

Suppose the regions R_p and R_q are separated by n planes. By Thm. 1 there is a chain $R_p = R_0, R_1, \dots, R_n = R_q$ of length $(n+1)$ connecting them. Regions R_q and $-R_p$ are separated by the $(M-n)$ planes not separating R_p and R_q , so there is a chain $R_q = R_n, R_{n+1}, \dots, R_M = -R_p$ of length $(M-n+1)$ connecting them. The chain $R_p = R_0, R_1, \dots, R_n, R_{n+1}, \dots, R_M = -R_p$ contains $(M+1)$ regions, counting both R_p and $-R_p$. Thus the chain $R_p = R_0, R_1, \dots, R_M = -R_p, -R_1, \dots, -R_{M-1}$ contains exactly M regions and their reflections, and is a cycle joining R_p and R_q .

Intuitively, a cycle is constructed by starting in any region R_k , moving across a plane into a neighboring region, and continuing to move into neighboring regions without crossing any plane a second time until the region $-R_k$ is reached after exactly M crossings. These regions together with their reflections form a cycle. This procedure is simply the inductive construction used to prove Thm. 1, where the regions R_k and $-R_k$ are the two regions to be joined by a chain.

Suppose we have chosen a particular cycle. We renumber all of the regions so the members of the cycle are the regions R_1, R_2, \dots, R_{2M} , with $R_{M+m} = -R_m$, $m = 1, \dots, M$. We also renumber the planes so the regions R_m and R_{m+1} are separated just by the plane π_m , $m = 1, \dots, M$; then the plane π_m also separates the regions R_{M+m} and R_{M+m+1} , $m = 1, \dots, M-1$, and plane π_M separates R_{2M} and R_1 . The new matrix A represents the same configuration of planes and regions as before the renumbering. Now the cycle is represented by the submatrix B , consisting of the first $2M$ columns of A . For the remainder of the appendix we shall assume the regions and planes have been numbered in this way so that the particular cycle chosen consists of the regions R_1, \dots, R_{2M} , with regions R_m and R_{m+1} possessing the common boundary π_m , where the subscript on π_m is taken modulo M .

We say a region in a cycle B is positive with respect to the cycle if it lies on the positive side of both of its boundary planes separating it from its neighbors in the cycle. That is, R_k is positive with respect to the cycle B if $a_{k-1,k} = 1$ and $a_{k,k} = 1$, where the first subscript on a is taken modulo M .

Construct a committee C as follows. Let,

$$c_k = \begin{cases} 1 & \text{if } R_k \text{ is in } B \text{ and is positive with} \\ & \text{respect to the cycle } B. \\ 0 & \text{otherwise.} \end{cases}$$

Given the matrix A representing all of the regions and the matrix B representing the regions belonging to the cycle, we have

$$c_k = \begin{cases} 1 & \text{if } R_k \text{ is in } B, \text{ and } a_{k-1,k} = a_{k,k} = 1, \\ & \text{where the first subscript is taken modulo } M. \\ 0 & \text{if either } R_k \text{ is in } B \text{ but not both} \\ & a_{k-1,k} = 1 \text{ and } a_{k,k} = 1 \text{ (first subscript} \\ & \text{taken modulo } M). \\ & \text{or } R_k \text{ is not in } B. \end{cases}$$

We now prove that C is a majority rule solution committee.

Consider plane π_k and, for definiteness, suppose region R_{k+1} is on the positive side of π_k . Then regions $R_{k+1}, R_{k+2}, \dots, R_{k+M}$, subscripts reduced modulo $2M$, are all on the positive side of π_k . For, if not, the corrected set forming the cycle would cross π_k more than twice, which is contrary to cycle construction. A committee member in region R_p of the cycle votes correctly or not with respect to plane π_k according to whether region R_p is in the half of the cycle on the positive side of π_k or not. Thus, the order of regions and their separating planes in the cycle controls the correctness of committee member voting.

A simple pattern possessing the cycle order relation is given by M oriented straight lines lying in a plane and passing through its origin. Each line represents one of the M planes π_j . The sector S_m is the angular space between two adjacent half-lines or rays from the origin, and represents the region R_m belonging to the cycle. We label the rays and sectors so that as we proceed along a circle with center O we enter the sectors in the order S_1, S_2, \dots, S_{2M} , and cross the rays in the order L_1, L_2, \dots, L_{2M} . Because of the construction, S_{M+m} is the reflection in the origin of S_m ; they are

called vertical sectors. Similarly, L_{M+m} is the reflection of L_m . A sector S_m is called positive (negative) if it lies on the positive (negative) side of each of its boundary lines L_{m-1}, L_m . Otherwise it is called half-positive.

We construct a committee p consisting of one member in each positive sector. This committee is located in the sectors in precisely the same way that the committee C is located in the regions of the cycle, since a particular sector lies on the positive or negative side of each of the lines according to whether the corresponding region in the cycle lies on the positive or negative side of the corresponding plane. Thus the committee C is a majority rule solution committee if and only if the committee p is.

The following theorem is proved in Sect. III, Ref. 13.

Theorem 3:

For M distinct oriented lines through the origin in a two-dimensional plane the committee p composed of $(2k+1)$ members, one belonging to each positive sector, is a solution committee located so that exactly $(k+1)$ committee members lie on the positive side of each line.

We give an example for $N = 3$. We may represent planes and regions in E^3 by lines and polygons in a two-dimensional plane. Let Γ be a plane parallel to one of the pattern planes π_1 and on the positive side of π_1 . Consider the figure in Γ formed by the intersection of Γ and the planes π_j . For $j \neq 1$, $\Gamma \cap \pi_j$ is a line L_j in Γ , inheriting a positive side from the positive side of

π_j . No three of the lines in Γ are concurrent, since the planes were assumed to be in general position. The intersection of Γ and a region R_p on the positive side of π_1 is a polygon. It represents both the region R_p and its reflection $-R_p$. In placing a committee in the polygons, a $+$ will indicate a committee member in the region on the positive side of π_1 , and a $-$ will indicate a committee member in the region on the negative side of R_1 .

Consider a configuration of six planes, represented in a plane Γ parallel to π_1 in Fig. A-1. The positive direction of π_1 is out of the paper. Choose a cycle whose regions are $R_1, R_2, R_3, R_4, R_5, R_6$ and their reflections. A solution committee located in this cycle according to Thm. 3 consists of one committee member in each of the regions $R_1, R_3, R_6, -R_2, -R_5$. This committee is illustrated in Fig. A-1.

As previously noted, any committee member may be moved in the positive direction across a boundary from a region to a neighboring region. We move the committee members as follows:

from R_1 to R_7
 from R_3 to R_9
 from R_6 to R_{10} to R_9
 remain in $-R_2$
 from $-R_5$ to R_7

This committee of five, with two members in each of R_7 and R_9 and one in $-R_2$, votes the same as a committee C of three members, one in each of these regions. This committee of three can be obtained directly by choosing the cycle $R_1, R_7, R_8, R_9, R_{10}, R_6$ and their reflections.

A solution committee located in this cycle according to Thm. 3 consists of one member in each of regions $R_7, -R_8, R_9$. The member in $-R_8$ may be moved into region $-R_2$, giving the committee C.

The cycle $R_8, R_2, R_1, R_{14}, R_{15}, R_{16}$ and their reflections leads to a committee of three, one member in each of the regions $R_8, -R_2, R_{14}$. Moving the member in R_8 into the region R_7 , we see that this cycle produced a different committee from either of the first two cycles.

The example shows that different cycles produce committees of different sizes and residing in different regions, although every committee located in a cycle according to Thm. 3 is a solution committee.

APPENDIX B

**A MAJORITY SOLUTION
COMMITTEE FOR
THE PARITY FUNCTION**

by

D. J. Kaylor

N. J. Nilsson

APPENDIX B

A MAJORITY SOLUTION COMMITTEE FOR THE PARITY FUNCTION

In this appendix we prove that there exists in E^M an M -member majority solution committee for the N -dimensional parity function, where $M = N$ if N is odd and $M = N+1$ if N is even. The pattern vectors will be the vertices of a cube, and the committee members will be represented by planes. In a solution committee governed by majority rule, all the planes pass through the origin and points in class 1 lie on the positive side of a majority of the planes, while points in class 0 do not.

Let $x = (x_1, \dots, x_N)$ be a vertex of the unit N -dimensional cube; each $x_i = \pm 1$. Let $s(x) = \sum_{i=1}^N x_i$ be the sum of the components of x . Let $w(x) = \sum_{x_i=+1} 1$ be the number of components of x which are equal to $+1$. The parity function $p(x)$ is defined as $p(x) = w(x) \pmod{2}$. That is, x is in class 0 if $w(x)$ is even and in class 1 if $w(x)$ is odd.

Remarks:

1. If $w(x) - w(y) \equiv 1 \pmod{2}$, then x and y are in different categories.
2. $w(x) + w(-x) = N$.
3. $s(x) = w(x) - w(-x)$.
4. $p(x) = \frac{1}{2}[s(x) + N] \pmod{2}$.
5. $w(x) = w(y)$ if and only if $s(x) = s(y)$

A function is said to be partitioned by K planes if none of the regions formed by the planes contains points assigned to different categories by the function.

Theorem 1:

The N -dimensional parity function in E^N may be partitioned by N planes.

Proof:

Project the vertices of the cube orthogonally onto the major diagonal D connecting the points $(-1, \dots, -1)$ and $(1, \dots, 1)$. All vertices x with $s(x) = L$ have the image $y_L = (L/N, \dots, L/N)$. The $N+1$ points y_L located on the diagonal D and the category of all vertices mapping into y_L are shown in Fig. B-1. A set of partitioning planes for the parity function is the set of N planes P_j perpendicular to D and halfway between neighboring vertex images. P_j is the plane whose equation is $\sum_{i=1}^N x_i = \frac{2j-N-1}{N}$ ($j = 1, \dots, N$).

Theorem 2:

The $(2k)$ -dimensional parity function may be partitioned in E^{2k+1} by $2k+1$ planes through the origin so the committee votes by majority rule.

Proof:

Partition the parity function in E^{2k} with $2k$ $(2k-1)$ -dimensional planes P_j according to Thm. 1. Now imbed E^{2k} in E^{2k+1} on the flat $x_{2k+1} = -1$. Since no P_j passes through the origin, each P_j and the origin determine a $(2k)$ -dimensional plane P_j^* . Let P_{2k+1}^* be the plane whose equation is $x_{2k+1} = 0$. The planes P_j^* ($j = 1, \dots, 2k+1$)

partition the parity function, since each P_j^* ($j = 1, \dots, 2k$) separates class 0 points from class 1 points (see Fig. B-2). Let the positive side of P_j^* be that containing the nearby class 1 points ($j = 1, \dots, 2k$) and the positive side of P_{2k+1}^* be that for which $x_{2k+1} > 0$.

Each vertex x with $p(x) = 0$ is on the positive side of k planes. For, the vertex $\alpha = (-1, -1, \dots, -1)$ is on the positive side of the k planes P_{2n}^* ($n = 1, \dots, k$); to travel from α to a point x with $w(x)$ even, we cross m planes in the positive direction and m planes in the negative direction. Each vertex y with $p(y) = 1$ is on the positive side of $k + 1$ planes. For, the vertex $\beta = (1, -1, \dots, -1)$ is on the positive side of the $k + 1$ planes P_{2n}^* ($n = 1, \dots, k$) and P_1^* ; to travel from β to a point y with $w(y)$ odd, we cross m planes in the positive direction and m planes in the negative direction. Thus, points in class 1 lie on the positive side of a majority of the $2k+1$ planes, and points in class 0 lie on the positive side of a minority of the planes.

Theorem 3:

The $(2k+1)$ -dimensional parity function may be partitioned in E^{2k+1} by $2k+1$ planes through the origin so the committee votes by majority rule.

Proof:

The $(2k+1)$ -dimensional parity function may be viewed as two $(2k)$ -dimensional parity functions, one on the flat $x_{2k+1} = +1$, and the other on the flat $x_{2k+1} = -1$. Partition the parity function on the flat $x_{2k+1} = -1$ according to Thm. 2. We shall prove that the

planes P_j^* ($j = 1, \dots, 2k+1$) form a majority solution committee for the $(2k+1)$ -dimensional parity function.

For vertices x with $x_{2k+1} = -1$, the proof of Thm. 2 demonstrates that x is properly classified. The vertex $y = -x$ (with $y_{2k+1} = 1$) is in the opposite class from x , since

$$w(y) - w(x) = w(-x) - w(x) = 2k+1 - 2w(x) \equiv 1 \pmod{2}.$$

Since all the planes go through the origin, y is on the opposite side from x of every plane. Thus, if x is in class 0, it is on the positive side of k planes (by Thm. 1), and y is on the positive side of $(2k+1) - k = k+1$ planes, so is classified Category 1, as required. If x is in class 1, it is on the positive side of $k+1$ planes (by Thm. 1) and y is on the positive side of $(2k+1) - (k+1) = k$ planes, so is classified Category 0, as required.

REFERENCES

1. H. D. Block, N. J. Nilsson, and R. O. Duda, "Determination and Detection of Features in Patterns," Technical Documentary Report No. RADC-TDR-63-467, Rome Air Development Center, Griffiss Air Force Base, New York, Prepared under Contract AF 30(602)-2943, (Dec. 1963).
2. F. Rosenblatt, Principles of Neurodynamics: Perceptrons and the Theory of Brain Mechanisms, (Spartan Books, Washington, D.C., 1961).
3. B. Widrow, "Generalization and Information Storage in Networks of Adaline 'Neurons'," Self-Organizing Systems-1962, (Yovits, Jacobi and Goldstein (Eds.), (Spartan Books, Washington, D.C., 1962).
4. T. Harley, et al, "Semi-Automatic Imagery Screening Research Study and Experimental Investigation," Philco Reports Nos. VO43-2 and VO43-3, Vol. I, Sec. 6 and Appendix H, prepared for U.S. Army Electronics Research and Development Laboratory under Contract No. DA-36-039-SC-90742 (March 29, 1963).
5. R. O. Winder, "Threshold Logic in Artificial Intelligence", IEEE Publication S-142, Artificial Intelligence, (A combined preprint of papers presented at the Winter General Meeting, 1963), New York, 1963.
6. B. Widrow and M. E. Hoff, "Adaptive Switching Circuits," Stanford Electronics Laboratories Technical Report No. 1553-1, Stanford University, Stanford, California, (June 30, 1960).
7. W. Highleyman, "Linear Decision Functions with Applications to Pattern Recognition," Proc. IRE, Vol. 50, pp. 1501-1515 (June, 1962).
8. T. M. Cover, "Classification and Generalization Capabilities of Linear Threshold Units," submitted as a Technical Documentary Report to RADC under Contract AF 30(602)-2943 (November, 1963).
9. B. Widrow, "Generalization and Information Storage in Networks of Adaline 'Neurons'," Self-Organizing Systems-1962, p. 442 (Yovits, Jacobi and Goldstein (Eds.), (Spartan Books, Washington, D.C., 1962).

10. R. Brown, "Logical Properties of Adaptive Networks," Stanford Electronics Laboratory, Quarterly Research Review, No. 4, III-6 to III-9, 1963.
11. R. O. Winder, "Bounds on Threshold Gate Realizability," IEEE Trans. on Elect. Computers, Vol. EC-12, No. 5, pp. 561-564 (October 1963).
12. A. E. Brain, "Graphical Data Processing Research Study and Experimental Investigation," Report No. 12, Prepared for U.S. Army Signal Research and Development Laboratory under Contract DA-36-039-SC-78343, 1963.
13. D. J. Kaylor, "A Mathematical Model of a Two-Layer Network of Threshold Elements," submitted as a Technical Documentary Report to RADC under Contract AF 30(602)-2943, (Oct. 1963).
14. T. S. Motzkin and I. J. Schoenberg, "The Relaxation Method for Linear Inequalities," Can. J. of Math., Vol. 6, No. 3, pp. 393-404, (1954).
15. A. B. J. Novikoff, "On Convergence Proofs for Perceptrons," Technical Report Prepared for the Office of Naval Research, Washington, D.C., under Contract Nonr 3438(00), Stanford Research Institute, Menlo Park, California, (January 1963).
16. N. J. Nilsson and R. C. Singleton, "An Experimental Comparison of Three Learning Machine Training Rules," submitted as a Technical Documentary Report to RADC under Contract AF 30(602)-2943, (Nov. 1963).
17. B. Efron, "The Perceptron Correction Procedure in Non-Separable Situations," submitted as a Technical Documentary Report to RADC under Contract AF 30(602)-2943, (August 1963).
18. W. C. Ridgway, "An Adaptive Logic System with Generalizing Properties," Stanford Electronics Laboratories Technical Report, No. 1556-1, prepared under Air Force Contract AF 33(616)-7726, Stanford University, Stanford, California, April, 1962.

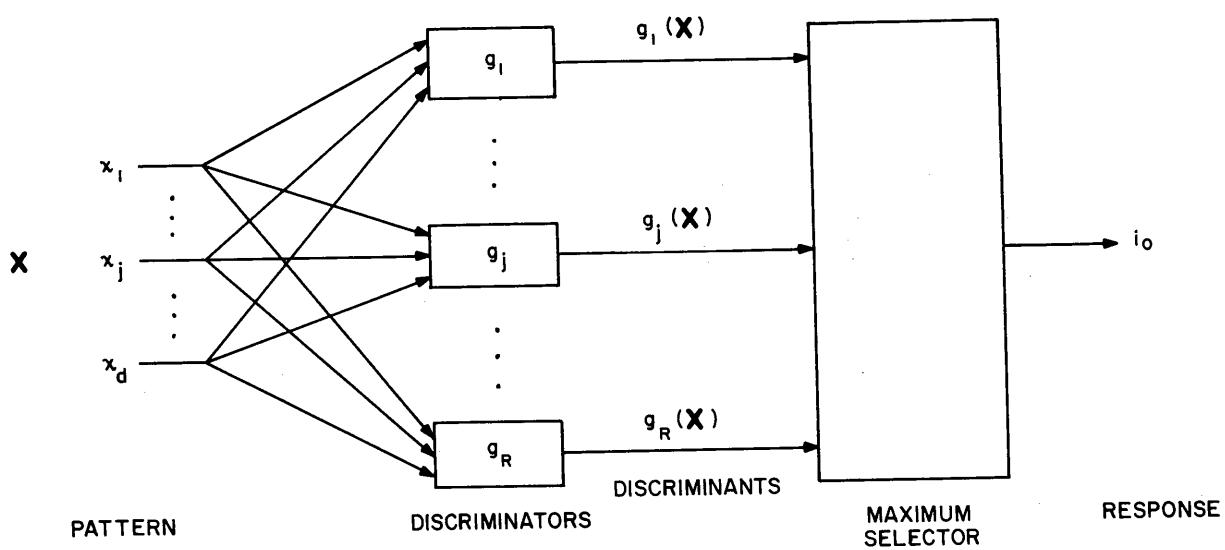


FIG. 1 BASIC MODEL FOR A PATTERN CLASSIFIER

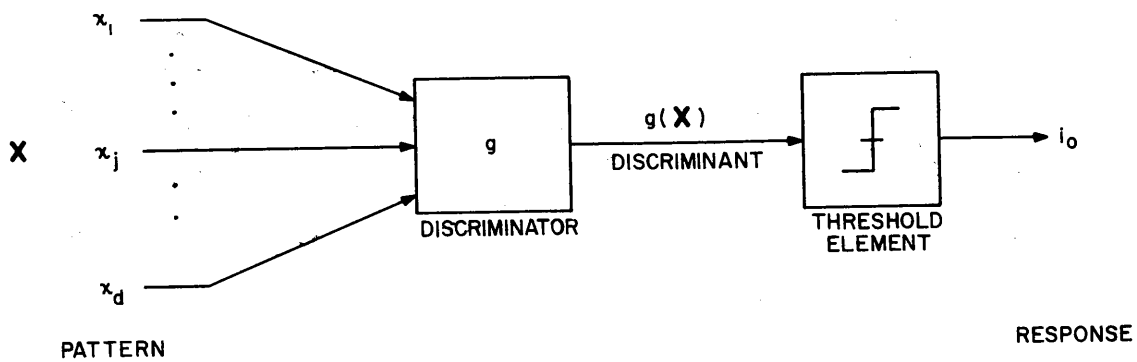


FIG. 2 BASIC MODEL FOR A PATTERN DICHOTOMIZER

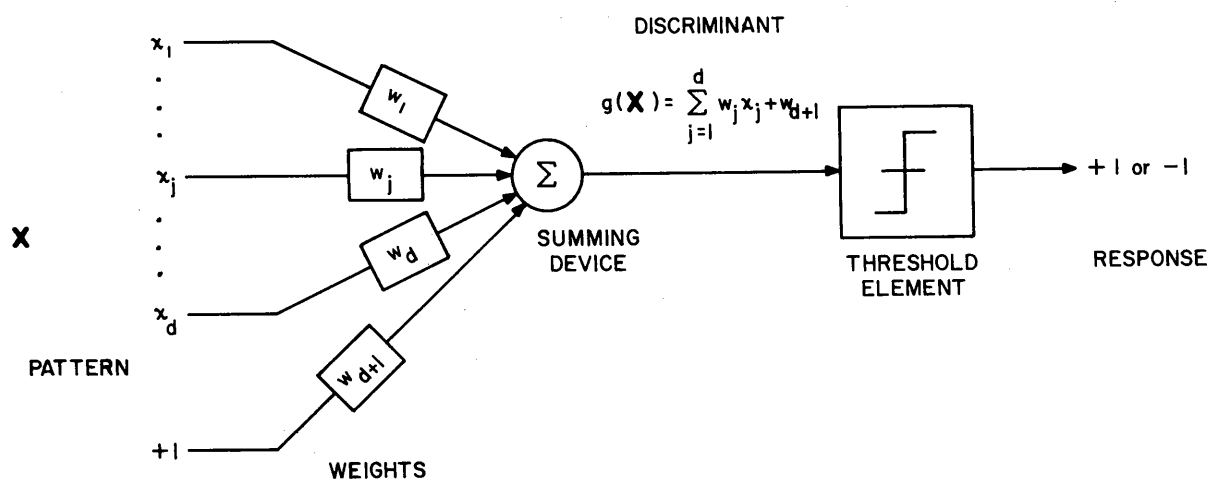


FIG. 3

THE THRESHOLD LOGIC UNIT (TLU)

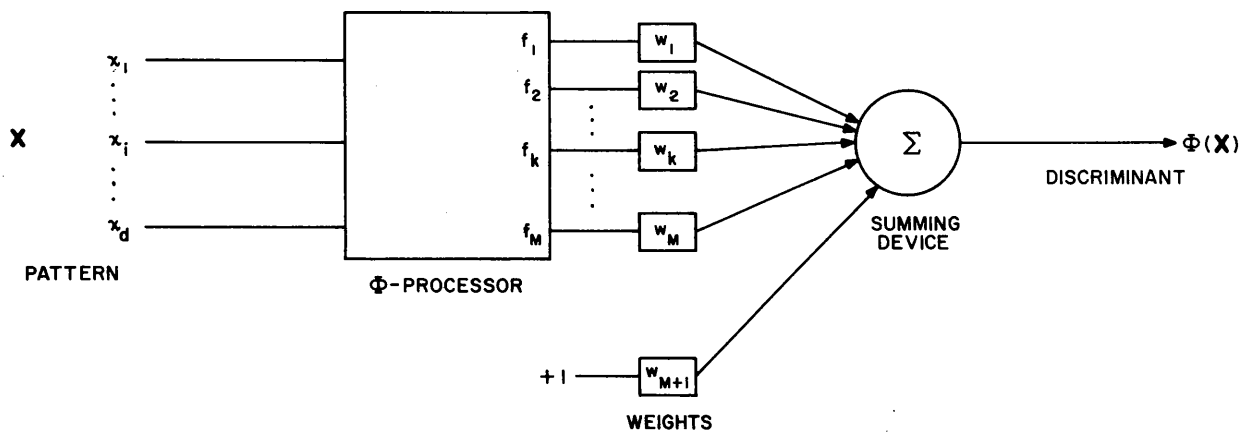


FIG. 4 A Φ -FUNCTION DISCRIMINATOR

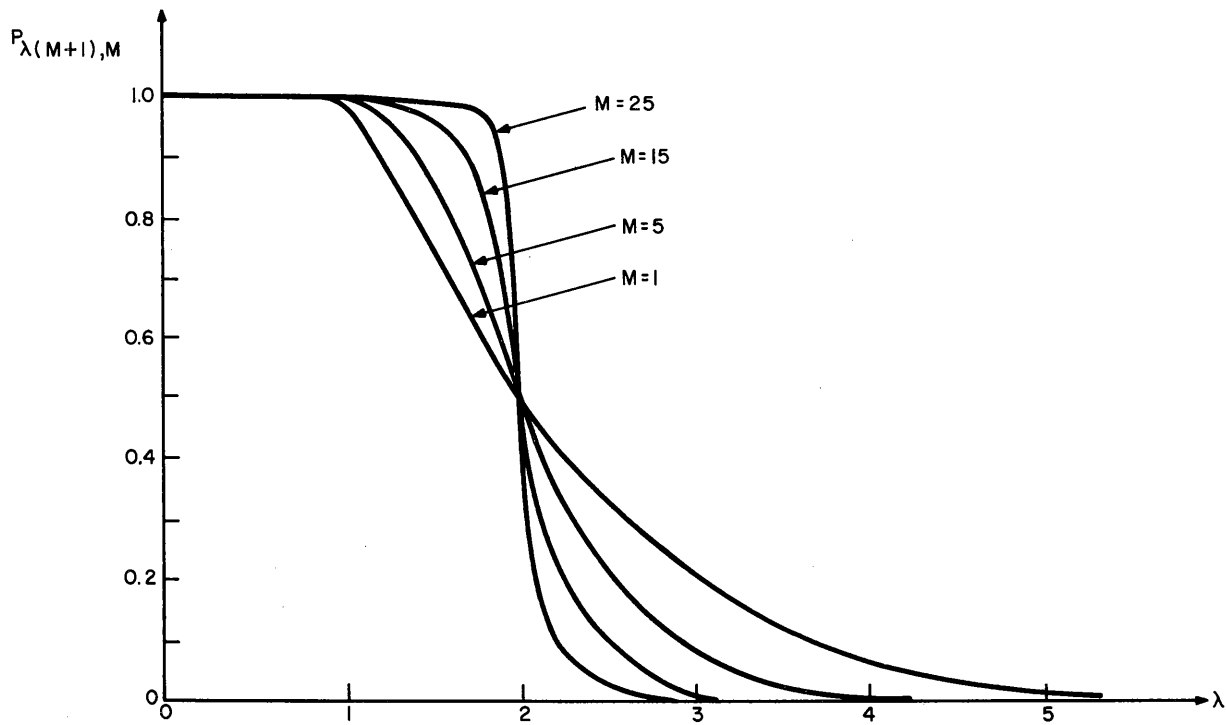


FIG. 5 $P_{\lambda(M+1),M}$ VS λ FOR VARIOUS VALUES OF M

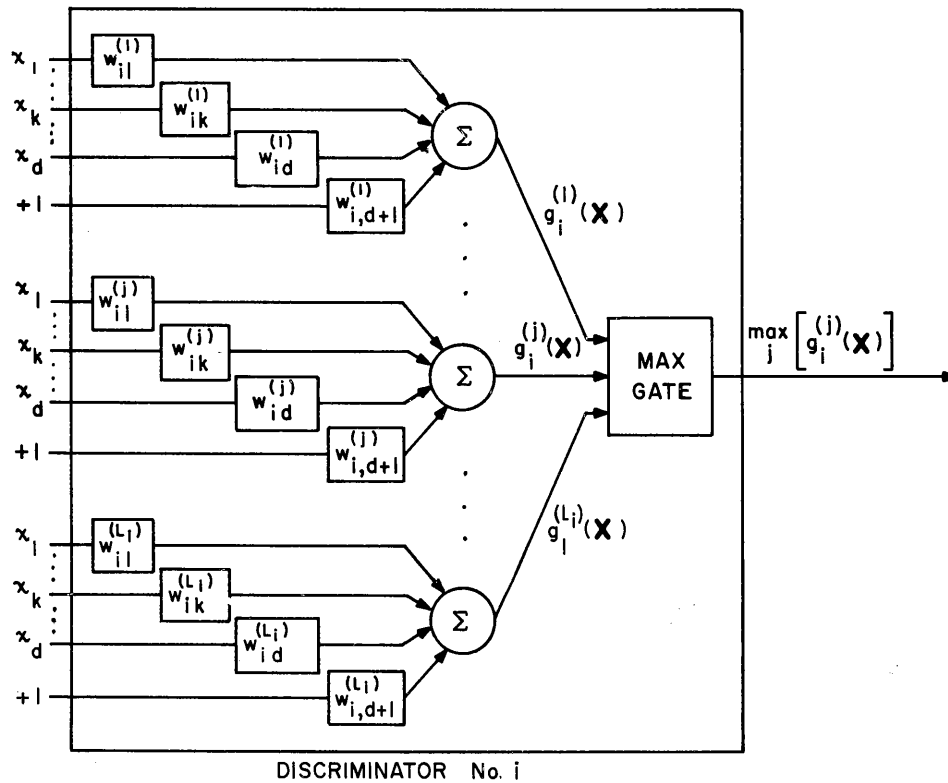


FIG. 6 A PIECEWISE LINEAR DISCRIMINATOR

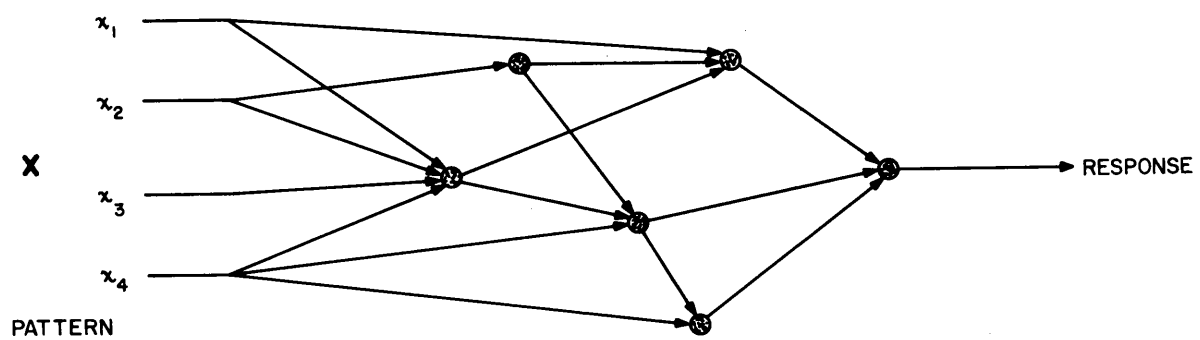


FIG. 7 A NETWORK OF TLU'S

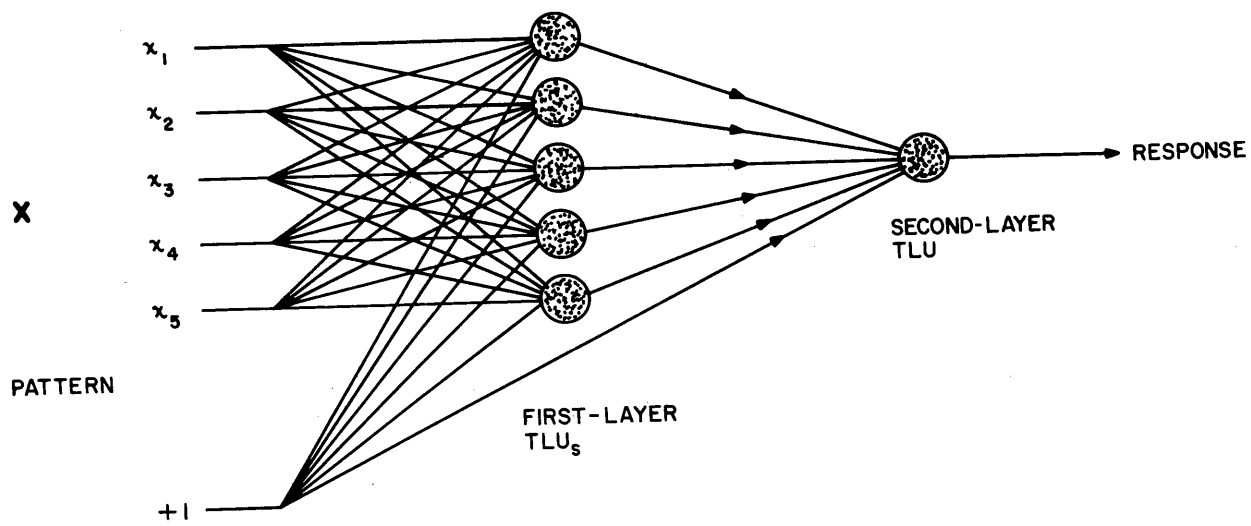


FIG. 8 A LAYERED NETWORK OF TLU'S

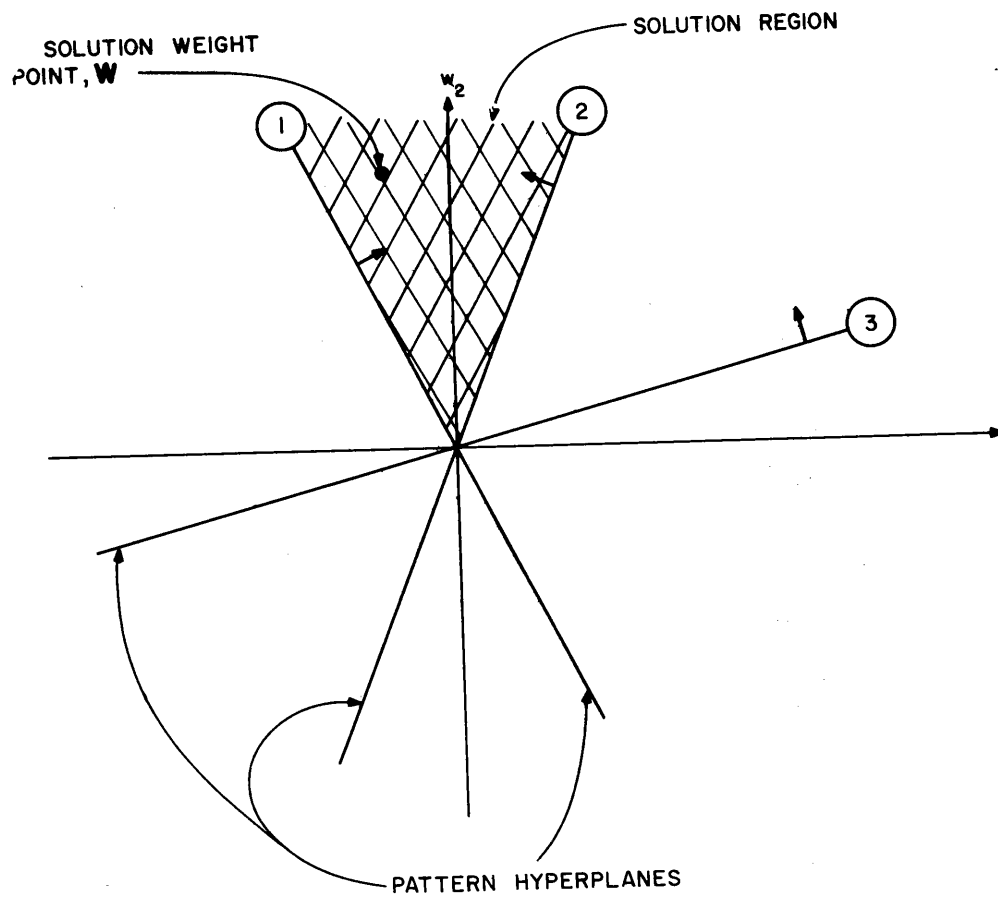


FIG. 9 A TWO-DIMENSIONAL WEIGHT-SPACE WITH THREE PATTERN HYPERPLANES

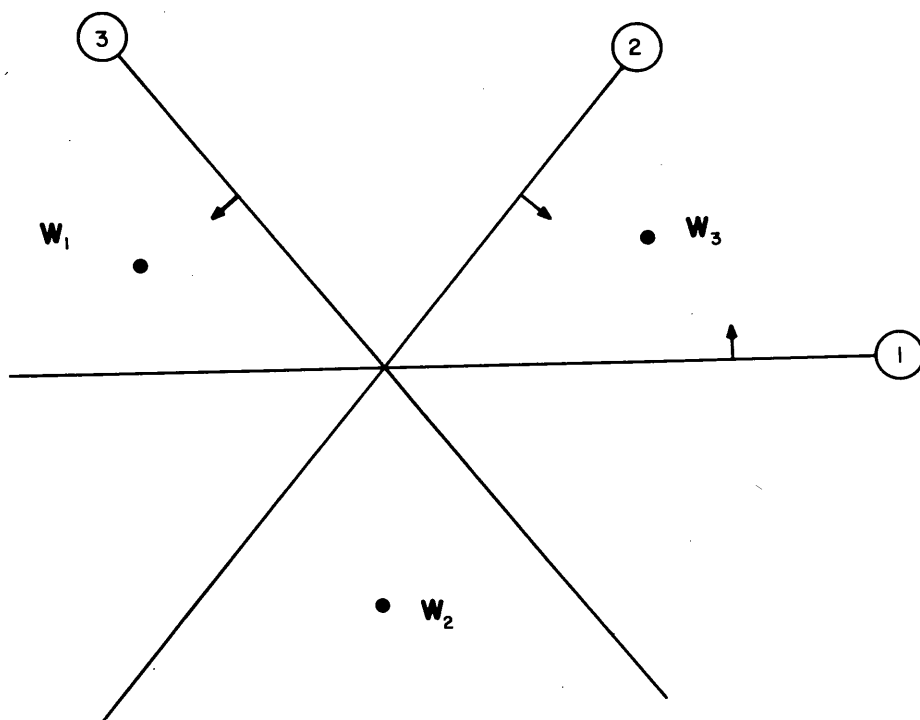


FIG. 10 A COMMITTEE OF WEIGHT VECTORS

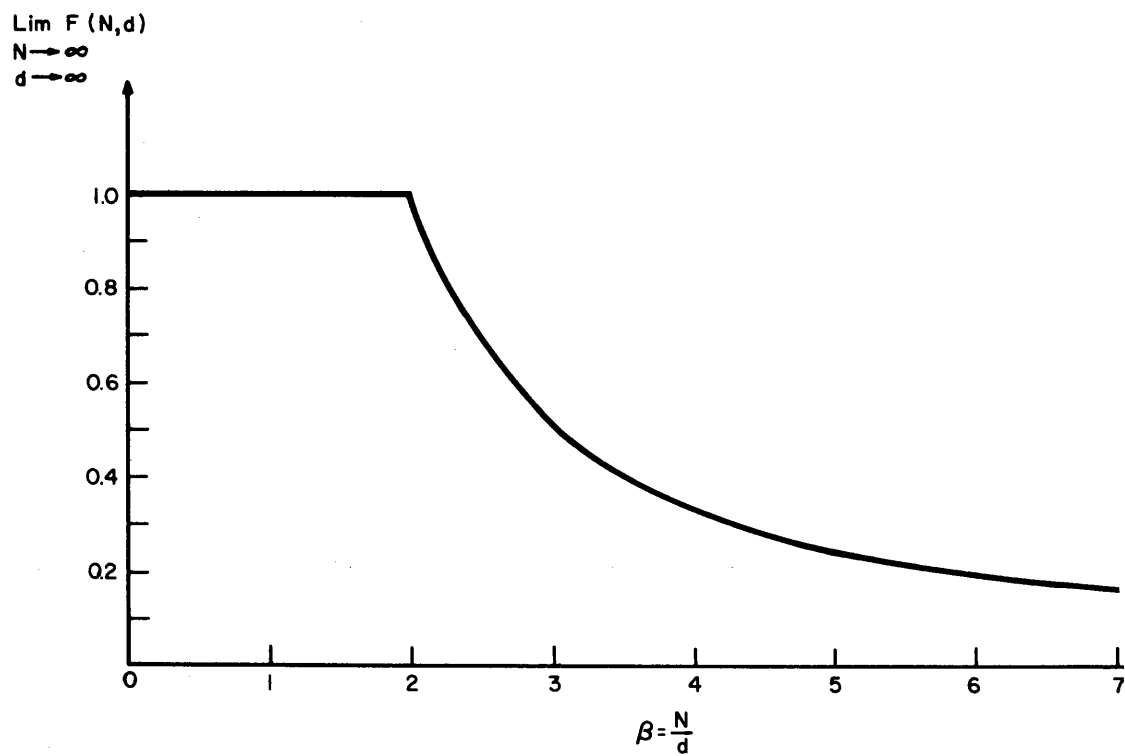


FIG. 11 ASYMPTOTIC CURVE FOR PROBABILITY OF AMBIGUITY

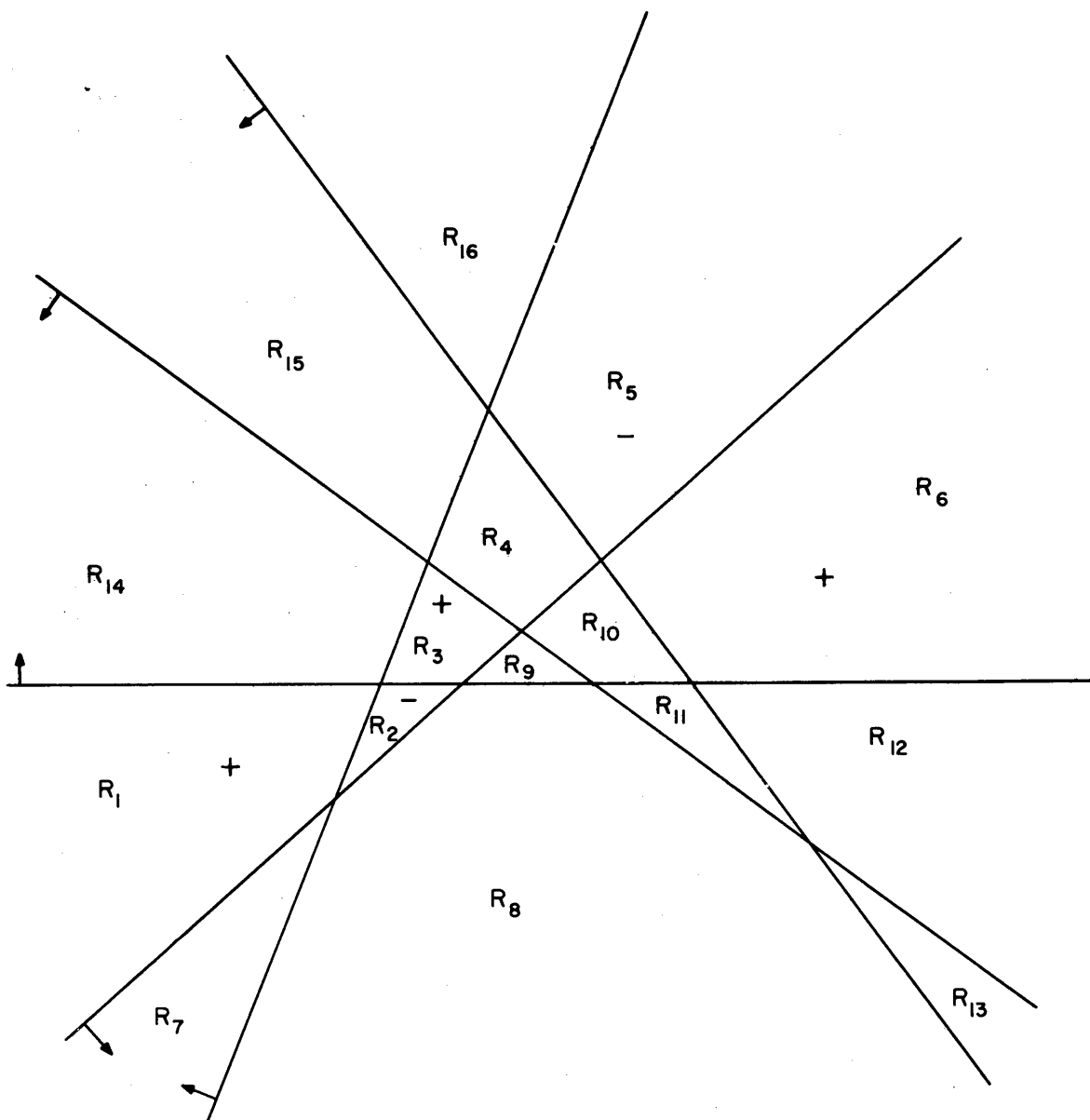


FIG. A-1 A CONFIGURATION OF SIX PLANES IN THREE DIMENSIONS

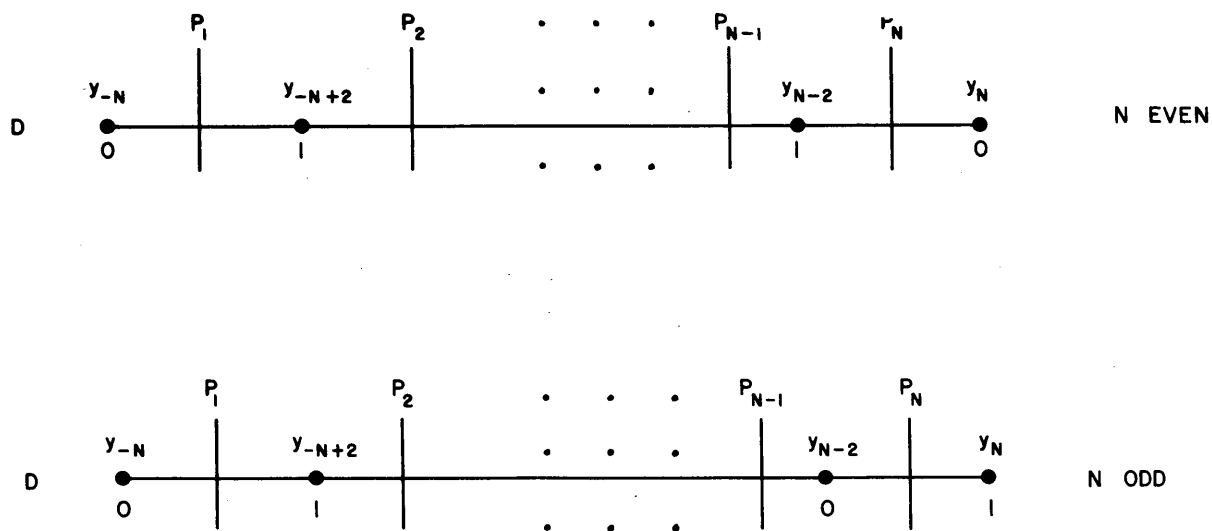


FIG. B-1 PROJECTION OF VERTICES OF THE N-CUBE ON THE MAJOR DIAGONAL

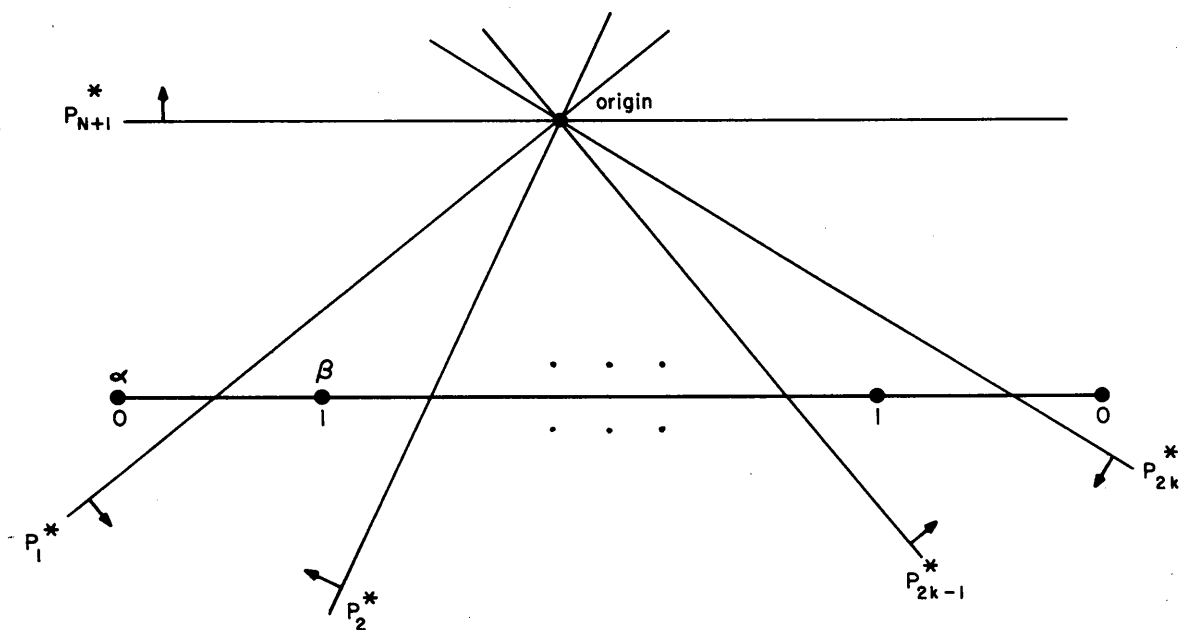


FIG. B-2 PARTITION OF THE $(2k)$ -DIMENSIONAL PARITY FUNCTION IN E^{2k+1}

