63-1-3

AIR FORCE INSTITUTE OF TECHNOLOGY

INSTITUTE OF TECHNOLOGY

AIR UNIVERSITY
UNITED STATES AIR FORCE

SCHOOL OF ENGINEERING

WRIGHT-PATTERSON AIR FORCE BASE, OHIO

AF-WP-O-MAY 62 3,500

THRESHOLDING
IN
SELF-ORGANIZING MACHINES

GA/EE/62-3 Robert Bradley Stuart
Capt USAF

THESIS

Presented to the Faculty of the School of Engineering

The Institute of Technology

Air University

in Partial Fulfillment of the

Requirements for the

Master of Science Degree

in Astronautics

THRESHOLDING
IN
SELF-ORGANIZING MACHINES

GA/EE/62-3 Robert Bradley Stuart
Capt USAF

Graduate Astronautics

August 1962

Preface

This thesis is the result of a study comparing self-organizing machines applicable to computation and pattern recognition. The combinatorial units in many of these machines are Boolean function generators which produce an output if the weighted algebraic sum of the binary inputs is greater than a specified value called the threshold. An example of a unit of this type is the toroidal ferrite core used in computer memories, which produces a pulse on the output or read-out line only when the direction of the core's magnetic field reverses. This reversal occurs when the magneto-motive force, produced by one or more linking turns of current carrying wire, exceeds a value called the coercive force. In this case, one may think of the inputs as unit current pulses in separate wires linking the core, the weights and their signs as the number of turns of each wire linking the core in a given direction, and the threshold as the number of ampere-turns required to reverse the magnetic field. Units of this type, including the magnetic core, relays with multiple solenoids, and various tube and semi-conductor circuits, are called threshold units, and are capable of producing certain Boolean switching functions of the input variables. The thesis is concerned with cascades of these units, connected together by amplifying devices to form switching circuits. The amplifying devices are assumed to multiply the transmitted signals by integral weights. Statistical switches, whose closure probability at a given instant in time is adjustable, may replace, or supplement, the amplifying devices. Organization is defined as the adjustment of the thresholds, weights, and probabilities, to obtain a switching circuit which will produce the desired function, or functions, of the input variables.

The mathematical model described in Chapter II is deliberately generalized. In the description of this model an attempt has been made to develop a consistent symbology, since an appreciable amount of the research time in preparation of the thesis was devoted to learning the various notations of previous, more specialized, papers. It is felt that the contributions of the thesis are the development of the mathematical model, the methods applied to this model, and the system, referred to as the map-matrix method, applied to the analysis of the Boolean switching functions produced by threshold units.

The reader is assumed to have some knowledge of Boolean algebra and its switching circuit applications, including minterm canonical form expansions, truth tables, and Karnaugh map representations. Additional mathematical requirements are vector and matrix methods, and elementary combinatorial and probability theory. Some developments are based on elementary topology, but a knowledge of solid analytic geometry ahould

be sufficient for understanding these developments.

I wish to acknowledge my indebtedness to Captain Frank M. Brown for directing my interest in computers and logic into this channel, and for his help in enriching my rather meager knowledge of the subject. In addition, I wish to thank Mr. Cecil Gwinn of the Bionics and Computer Branch, Aeronautical Systems Division, for the use of many reports, and for the opportunity to observe two hardware realizations of self-organizing machines. Finally, I should like to express my appreciation to all those who have smoothed my path toward knowledge by reporting the results of their years of labor in a readable form in various technical journals and reports.

Robert B. Stuart

Abstract

The threshold unit is a simplified analog of the living neuron cell. Self-organizing machines have been constructed, using the threshold unit as a Boolean function generator. These machines may be represented by a mathematical model. The model may be analyzed by vector and matrix methods. Analysis of a simple Perceptron by these methods indicates that the ability of the machine to identify noisy patterns may depend on the existence of one threshold unit which assigns all noisy patterns to the same class as the corresponding correct patterns. A method based on matrix representation of Karnaugh maps may be used as a design tool for analysis of threshold units and functions.

Contents

Contents

THRESHOLDING IN SELF-ORGANIZING MACHINES

I. Introduction

A problem of major interest in the field of Bionics is the study and simulation of certain functions of the living brain. The brain represents the finest possible biological example of an existing, miniature, general-purpose, computing and control center. It receives inputs from a number of sensory receptors, processes the information contained in these inputs, stores some fraction of this information, and utilizes stored and transient information to control several remarkable servo systems. If the engineer were able to understand the mechanics of these various functions, and construct electro-mechanical analogs, much of the dangerous or dreary work which now requires a human observer or operator could be delegated to machines. For example, the exploration of space could be hastened by probes containing more sophisticated devices capable of sensing, reducing, and storing data, and forming statistical inferences.

One element of the brain is the neuron cell. The operation of these cells has been studied, and some relatively crude mathematical analogs constructed. The neuron is a nerve cell with many input lines and one output line. These lines are not completely analogous to the wires in an electrical circuit, but are sometimes represented as delay lines. The signal input lines are called dendrites, and terminate on the cell body at the synapses. Inputs which contribute to producing an output from the cell are termed excitatory, and those which tend to prevent an output, inhibitory. The cell forms a function of the signals appearing as inputs. If this function equals, or exceeds, a value called the threshold, a signal is produced on the output line. This output line is called the axon (Ref 4: 51).

There are a number of variables in the construction and operation of the living neuron cell. The input lines, or output line, may occasionally open in an unpredictable manner. The signal strengths may change, the delays in the lines may vary, and the threshold value may fluctuate (Ref 4: 52). A mathematical model capable of including the effects of these variations exists, in the form of an eighth-order, non-linear, differential equation. This equation may be solved, but contains the possibility that under a given set of conditions the cell may oscillate between the output and no-output conditions (Ref 5:493). A simplified model, called a threshold unit, in which the presence or absence of an output signal depends on the weighted algebraic sum of the binary input variables, will be used in this analysis. Time delays will be assumed equal to zero.

Estimates of the number of neurons in the human brain vary, but a commonly used figure is 10^{10}. If these neurons are to compute, classify, or control, they must be connected, but very little is known about the connective nets. The limit of current knowledge seems to be that certain regions of the brain receive inputs from certain senses, other regions control motor actions, and some regions may be removed with no apparent effect.

In Chapter II, a net based on the weighted-sum neuron model, or threshold unit, is described. Although a number of simplifying assumptions are made, this net could provide many problems. To restrain this analysis, an experimental problem concerned with the properties of a simplified net has been selected and solved in as general a manner as possible in Chapter III. The solution of the problem indicates that the noise-resisting ability of thresholded self-organizing machines may depend on the existence of at least one threshold unit meeting certain criteria. A method, called the map-matrix method, for the analysis of threshold functions, supports this theory. The method is introduced in Chapter II, and developed further in Appendix A.

Conclusions based on the analysis are presented in Chapter IV, along with recommendations for further research based on the model and methods developed.

II. Mathematical Model

The mathematical model for analysis of the properties of nets of simplified neuron analogs, or threshold units, will be developed in two parts; a description of the basic components, or building blocks, and a description of one stage of a general net, constructed from these building blocks. The symbology, notation, and mathematical methods, will be introduced. Examples will be used for illustration. Description of the self-organizing method for a specialized simplification of the net will be presented in Chapter III.

Logical connectives will be symbolized by:

$+$	Inclusive OR
$\oplus$	Exclusive OR
$\bar{x}$	NOT x
$\cdot$	AND

except in cases where AND may be indicated by juxtaposition without ambiguity. Vectors will be defined as rows, and indicated by a single underline. Matrices will be indicated by a double underline. Thus, $\underline{S}$ is a row vector, $\underline{S}^T$ is a column vector, and $\underline{\underline{S}}$ is a matrix, by definition. This non-standard notation is adopted to avoid confusion between

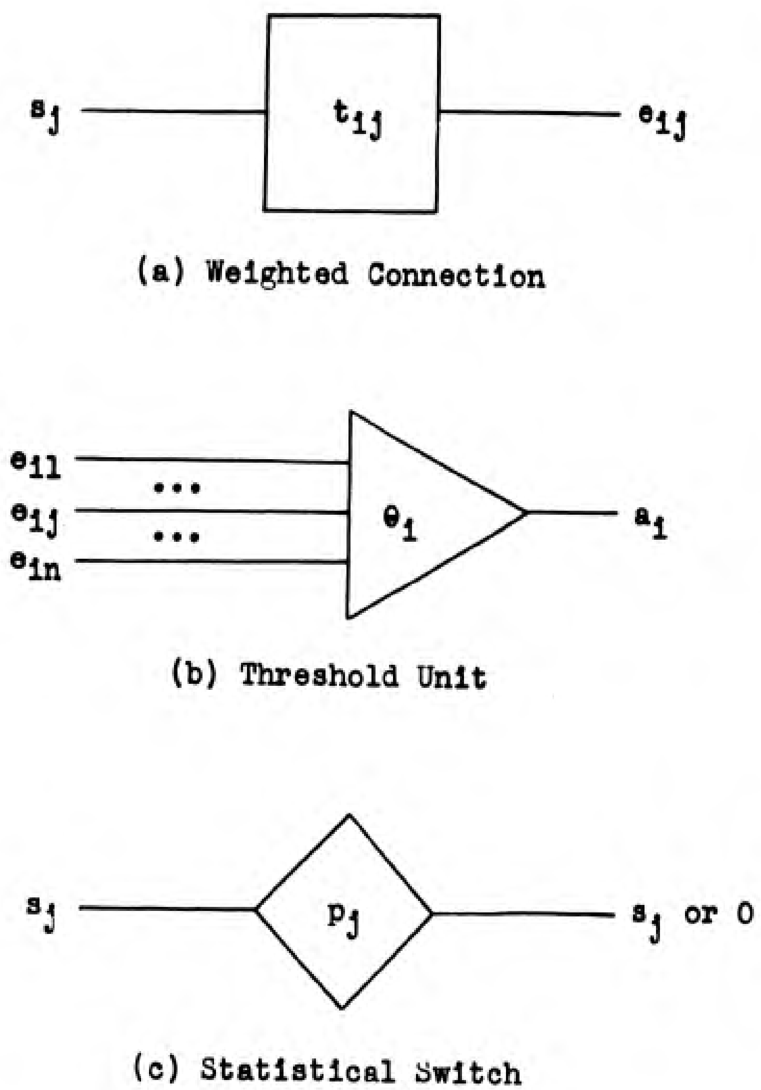


Fig. 1

Building Blocks

logical negation and vector notation. In general, vectors and matrices will be symbolized by capital letters, and their elements by corresponding small letters.

Building Blocks

The mathematical model is the result of a study of the Perceptron (Ref 6), the Artron (Ref 1), and the Adaline (Ref 8), three existing hardware realizations of theoretical models. It was found that the mathematical nature of these self-organizing machines could be represented by particular configurations of a generalized net constructed from stages formed of three basic components, or building blocks. These building blocks are the weighted connection, the threshold unit, and the statistical switch.

Weighted Connection. The weighted connection, shown in Fig. 1 (a), is the mathematical-schematic-representation of an amplifying unit with one input and one output. It multiplies the input signal, considered as a one-bit, binary, numerical variable, by an integer called the weight. The symbols, definitions, and restrictions are:

1. The input is a one-bit binary number corresponding to the state of a bi-stable device. The input binary variable is denoted by s_j, where j indicates the jth member of a numerically ordered set of n devices. The use of a symbol to represent a binary variable is adopted to allow direct

vector, truth table, and matrix methods.

2. If s_j, the binary variable, is equal to one, the corresponding bi-stable device is in one possible state, as specified. In the case of a lamp, this is the lighted state. If s_j is equal to zero, the device is in the other possible state, in the case of a lamp, unlighted.

3. The weighted connection multiplies the input by an integer called the weight. The weight is denoted in Fig. 1 (a) by t_{ij}, where j identifies the particular input variable, and i identifies the unit to which the weighted variable passes. The utility of this double subscript notation will be illustrated in the stage description.

4. The output signal from the weighted connection, denoted by e_{ij}, is equal to $s_j t_{ij}$. If the binary variable s_j is equal to one,

$$e_{ij} = s_j t_{ij} = (1)t_{ij} = t_{ij} \qquad (1)$$

If the binary variable s_j is equal to zero,

$$e_{ij} = s_j t_{ij} = (0)t_{ij} = 0 \qquad (2)$$

<u>Threshold</u> <u>Units</u>. Fig. 1 (b) is a stylization of the simplified neuron cell analog, or threshold unit. This unit performs two functions; combination, and signal generation. It forms the algebraic sum of the weighted input signals. If this sum equals, or exceeds, a specified integer called

the threshold, the unit produces an output signal whose value is one. If the sum is less than the threshold, there is no output. The symbols, definitions, and restrictions are:

1. The input signals, denoted in Fig. 1 (b) by e_{ij}, are the integers formed by the weighted connections.

2. The threshold unit forms the algebraic sum of the integer input signals. This sum, given n possible inputs to threshold unit i, is

$$\sum_{j=1}^{n} e_{ij}$$

3. If, and only if, the sum of the integer inputs equals, or exceeds, a specified integer, denoted by θ_i, and called the threshold of the ith threshold unit, the output of the unit, denoted by a_i, is equal to one. If the sum is less than the threshold, the output of the unit is zero. The mathematical specification of the threshold unit may be written as an inequality, where the symbols represent the variables defined.

$$\sum_{j=1}^{n} e_{ij} \geq \theta_i \iff a_i = 1 \tag{3}$$

$$\sum_{j=1}^{n} e_{ij} < \theta_i \iff a_i = 0 \tag{4}$$

Statistical Switch. The statistical switch, shown in Fig. 1 (c), is a switch which opens and closes in an essentially random manner. It is assumed that the closure probability at a given instant in time may be set to any desired value. The function of the statistical switch, developed for use in the Artron, and named by its designers, is the simulation of certain probabilistic features of logical behaviour (Ref 1:13-16). Its effect is somewhat analogous to the loading of a coin, in that the switch effectively biases the probability of obtaining a given result on any one trial, where the trial is the application of a signal at the input to the switch, and the result may be transmission or blocking of the signal. If transmission of the signal is represented by heads on the coin, and blocking by tails, then the probability of obtaining a head on any one toss of the loaded coin is analogous to the probability that the switch will transmit the signal.

The mathematical model will be assumed capable of reaching the always open and always closed states. The closure probability, denoted in Fig. 1 (c) by p_j, is

$$0 \leq p_j \leq 1 \qquad (5)$$

In accordance with standard probability notation, the probability that the switch will be open at any given instant is

$$q_j = 1 - p_j \qquad (6)$$

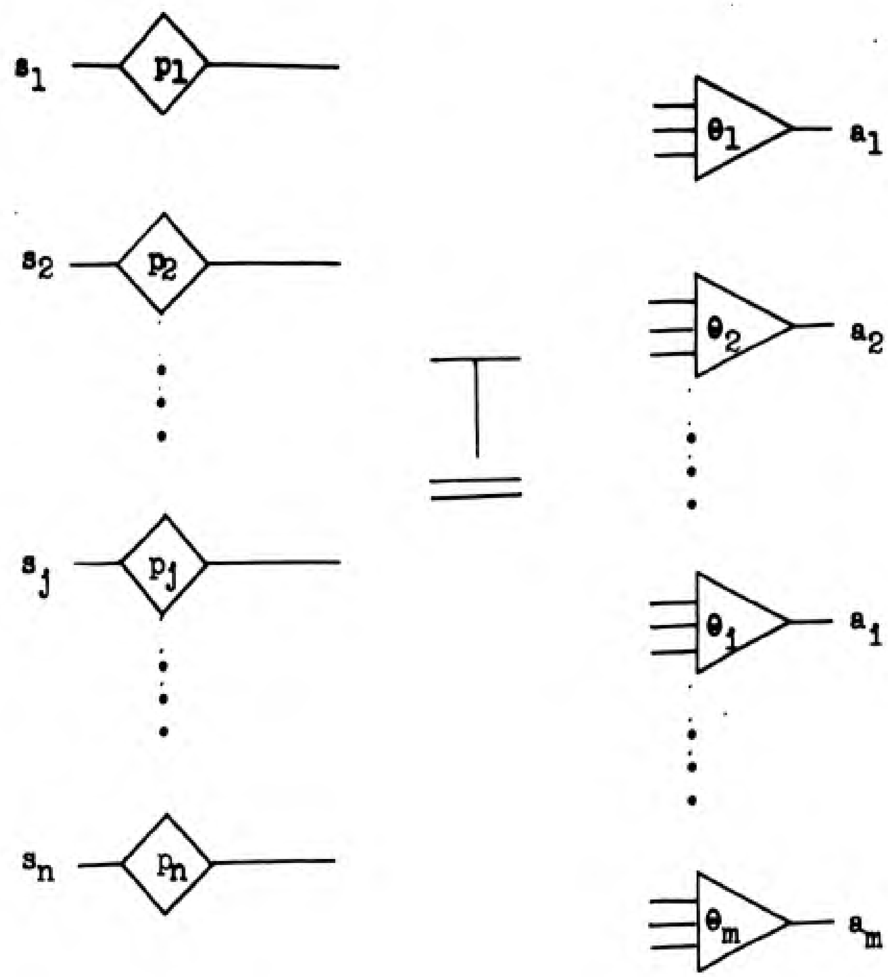


Fig. 2

Single Stage
of
Generalized Net

Stage Description

A single stage of the generalized net may be formed from the weighted connections, threshold units, and statistical switches, as illustrated in Fig. 2. In general, this stage is an n input, m output switching circuit. The inputs are the binary variables corresponding to an ordered set of n bi-stable devices; and the individual outputs are the m binary variables corresponding to m Boolean switching functions of the input variables. The stage description will proceed in a stepwise fashion, from input to output of the stage.

Input Configurations. The n inputs to the stage are one-bit, binary, numerical variables corresponding to n bi-stable devices, and are denoted in Fig. 2 by the small letter s, subscripted by the number assigned to the individual device and variable for identification. For, example, if a group of light bulbs, such as might appear as inputs to a pattern recognition device, is considered, the fact that bulb number one is lighted may be indicated by $s_1 = 1$. The unlighted state of the same bulb would then be identified by $s_1 = 0$. If the group consists of n devices, the state of the group at any one time may be mapped into an n element vector of 1 elements and 0 elements. The mapping may be accomplished by the identification of each device by a decimal number. The binary variable corresponding to the device is identified by the same decimal number, which appears as a subscript. The vector is constructed in row form, where the first element of the vector corresponds to the state of device one, the second element to the state of device two, and, in general, element j to the state of device j. The appearance of a 1 as element j of the vector indicates that bi-stable device j is in the state identified by $s_j = 1$. The appearance of a 0 as vector element j indicates that bi-stable device j is in the state identified by $s_j = 0$.

Example 1

Given: Two lights, identified numerically by position as

1 2

o o

each with two possible states, identified symbolically by

o lighted

● not lighted

and in binary variable notation by

o $s_j = 1$

● $s_j = 0$

<u>Example 1</u> (continued)

Then: The possible configurations, and binary element vectors, are

1	2	
●	●	configuration
0	0	vector elements
●	o	configuration
0	1	vector elements
o	●	configuration
1	0	vector elements
o	o	configuration
1	1	vector elements

Thus, the binary input to the stage may be considered as an n element vector, corresponding to one of the 2^n possible configurations of n bi-stable devices. This vector is defined as $\underline{S}$, and is called the input vector.

The vector $\underline{S}$ may be transformed into a square matrix $\underline{\underline{S}}$, where the binary elements of $\underline{S}$ appear along the principal diagonal of the matrix. If the element appearing in position j of the row vector $\underline{S}$ is defined as s_j, and the element appearing in row j, and column j, of the square matrix $\underline{\underline{S}}$, is defined as s_{jj}, then

$$s_{jj} = s_j \tag{7}$$

and the form of the matrix $\underline{\underline{S}}$ is, by definition

$$\underline{\underline{S}} = \begin{matrix} s_1 & 0 & 0 & \dots & 0 \\ 0 & s_2 & 0 & \dots & 0 \\ 0 & 0 & s_3 & \dots & 0 \\ \cdot & & & s_j & \cdot \\ 0 & 0 & 0 & \dots & s_n \end{matrix} \tag{8}$$

The $\underline{\underline{S}}$ matrix is useful in manipulations which will be discussed in connection with the threshold units.

Probabilistic Components. The statistical switches, which are the input devices of the stage, form components of the input $\underline{S}$ vectors, where the probability of any one particular component at any instant in time is determined by the closure probabilities of the individual switches. If the input to switch number one is a one; then the probability that a one will appear at the output of the switch at any particular instant in time is defined as p_1. For illustrative purposes, it is convenient to consider as an input an $\underline{S}$ vector composed entirely of ones. At a given instant the set of statistical switches may transmit all of the ones, some of the ones, or none of the ones. Thus, the set of statistical switches form a new set of vectors which are all possible components of $\underline{S}$.

Example 2

Given: An input vector $\underline{S}$, where

$\underline{S}$ = 1 1 1 1

and a group of four statistical switches, where element one of $\underline{S}$ passes through switch one, element two through switch two, element three through switch three, and element four through switch four.

Define: The probability that switch j will transmit element s_j as p_j, and the probability that switch j will block element s_j as q_j, where $q_j = 1 - p_j$.

Then: The possible components of $\underline{S}$ appearing as outputs from the four statistical switches, and their associated probability products, may be shown in tabular form.

Example 2 (continued)

Vector Components	Probability Products
0 0 0 0	$q_1q_2q_3q_4$
1 0 0 0	$p_1q_2q_3q_4$
0 1 0 0	$q_1p_2q_3q_4$
0 0 1 0	$q_1q_2p_3q_4$
0 0 0 1	$q_1q_2q_3p_4$
1 1 0 0	$p_1p_2q_3q_4$
0 1 1 0	$q_1p_2p_3q_4$
0 0 1 1	$q_1q_2p_3p_4$
1 0 0 1	$p_1q_2q_3p_4$
1 0 1 0	$p_1q_2p_3q_4$
0 1 0 1	$q_1p_2q_3p_4$
1 1 1 0	$p_1p_2p_3q_4$
1 1 0 1	$p_1p_2q_3p_4$
1 0 1 1	$p_1q_2p_3p_4$
0 1 1 1	$q_1p_2p_3p_4$
1 1 1 1	$p_1p_2p_3p_4$

It may be noted that the components represent all possible four-bit, binary words, and could also be mapped into truth value assignments for the four minterms of two variables.

Matrix Representation for Weighted Connections. Before considering the matrix representation of the weighted connections, it is convenient to assume that all statistical switches are temporarily set to the always closed position, that is, to assume all p_j equal to one. This defines a stage similar to the first stage of existing, self-organizing machines. Thus, the input to the weighted connections is the $\underline{S}$ vector, and not the entire set of all possible components of $\underline{S}$.

The weighted connections are represented in Fig. 2 by the $\underline{\underline{T}}$ matrix. This matrix, proposed by Captain Frank M. Brown in an unpublished memorandum, is a convenient form in which to arrange the individual t_{ij} previously defined in the description of the weighted connections. In general, given n inputs or $\underline{S}$ vector elements, and m threshold units, $\underline{\underline{T}}$ will be an m by n matrix. Each t_{ij} is the integer weight by which the input to the connection is multiplied. As previously defined, t_{ij} is the weight of the connection from the input identified by j to the threshold unit i. In the general case, the input may be s_j or zero, considered as variable with time as determined by the statistical switch with closure probability p_j, but the above assumption that p_j equals one restricts the inputs to constant values of s_j, such that the input to $\underline{\underline{T}}$ is not time dependent.

A constraint is placed on the rows of $\underline{\underline{T}}$ by the number of inputs allowed at each threshold unit. Thus, if the maximum number of inputs to threshold unit i is z, then row i of $\underline{\underline{T}}$ has, at most, z non-zero elements.

Example 3

Given: The partial network of statistical switches, weighted connections, and threshold units, as shown in Fig. 3, where

$$n = 2$$

$$m = 4$$

$$z = 2$$

and the weights are as shown. Let

$$p_1 = p_2 = 1$$

Then:

$\underline{\underline{T}}$ =

	s_1	s_2
unit 1	-1	-1
unit 2	-1	1
unit 3	1	-1
unit 4	1	1

In this example, z is equal to n, and no zero elements are required. However, in the case of physical threshold units, it is felt that production requirements would place some upper bound on z. In the case of magnetic core units, this might be the maximum number of wires which could be threaded through a single core.

The $\underline{\underline{T}}$ matrix may be constant or variable. In the Perceptron first stage, a constant matrix is constructed by the use of a random number table (Ref 7:1). In the Perceptron second stage, the elements are variables (Ref 6:4). In the Adaline, as described by its designers at a conference, the first stage elements are continuous variables.

<u>Combinatorial</u> <u>Units</u>. The combinatorial units for the stage diagrammed in Fig. 2 are threshold units. The inputs to the threshold units may be arranged in a matrix, defined as the $\underline{\underline{E}}$ matrix, and formed from $\underline{\underline{T}}$ and $\underline{\underline{S}}$ by matrix multiplication.

$$\underline{\underline{E}} = \underline{\underline{T}}\,\underline{\underline{S}} \tag{9}$$

Thus,

$$\underline{\underline{E}} = \begin{matrix} t_{11} & t_{12} & \cdots & t_{1j} & \cdots & t_{1n} \\ t_{21} & t_{22} & \cdots & t_{2j} & \cdots & t_{2n} \\ \cdot & & & & & \cdot \\ t_{i1} & & \cdots & t_{ij} & \cdots & t_{in} \\ \cdot & & & & & \cdot \\ t_{m1} & & \cdots & t_{mj} & \cdots & t_{mn} \end{matrix} \quad \begin{matrix} s_1 & 0 & \cdots & & 0 \\ 0 & s_2 & \cdots & & 0 \\ \cdot & & & & \cdot \\ 0 & \cdots & & s_j & \cdot\ 0 \\ \cdot & & & & \cdot \\ 0 & & \cdots & & s_n \end{matrix} \tag{10}$$

Multiplying:

$$\underline{\underline{E}} = \begin{array}{cccccc} s_1 t_{11} & s_2 t_{12} & \cdots & s_j t_{1j} & \cdots & s_n t_{1n} \\ s_1 t_{21} & s_2 t_{22} & & & & \\ \cdot\ \cdot & & & \cdot\ \cdot & & \\ s_1 t_{i1} & & \cdots & s_j t_{ij} & \cdots & s_n t_{in} \\ \cdot\ \cdot & & & \cdot\ \cdot & & \\ s_1 t_{m1} & & \cdots & s_j t_{mj} & \cdots & s_n t_{mn} \end{array} \qquad (11)$$

Hence:

$$e_{ij} = s_j t_{ij} \qquad (12)$$

This is the justification for the double subscript notation introduced previously in the description of the individual components, or building blocks. Each row, i, of the $\underline{\underline{E}}$ matrix has, as elements, the integers which are to be summed by the corresponding threshold unit i. The summations may also be written as a matrix multiplication if a summing vector is defined as an n element column vector of unity elements.

$$\underline{1}^T = \begin{array}{c} 1 \\ 1 \\ \cdot \\ 1_j \\ \cdot \\ 1_n \end{array} \qquad (13)$$

The vector formed by the multiplication of $\underline{\underline{E}}$ and $\underline{1}^T$ is defined as $\underline{\pi}^T$.

$$\underline{\pi}^T = \underline{\underline{E}}\,\underline{1}^T \tag{14}$$

Hence:

$$\underline{\pi}^T = \begin{matrix} e_{11} & e_{12} & \cdots & e_{1j} & \cdots & e_{1n} \\ e_{21} & e_{22} & & & & \\ \cdot & & & \cdot & & \cdot \\ e_{i1} & & \cdots & e_{ij} & \cdots & e_{in} \\ \cdot & & & \cdot & & \cdot \\ e_{m1} & & \cdots & & \cdots\cdots & e_{mn} \end{matrix} \quad \begin{matrix} 1 \\ 1 \\ \cdot \\ 1_j \\ \cdot \\ 1_n \end{matrix} \tag{15}$$

And, in general,

$$\pi_i = \sum_{j=1}^{n} e_{ij} \tag{16}$$

As defined previously, the output from threshold unit i, denoted by a_i, is one; if and only if, the algebraic sum of the weighted-binary-numerical input variables is equal to, or greater than the threshold, denoted by θ_i. Thus, a_i is a function of π_i as defined above, and θ_i.

$$\pi_i \geq \theta_i \iff a_i = 1 \tag{17}$$

$$\pi_i < \theta_i \iff a_i = 0 \tag{18}$$

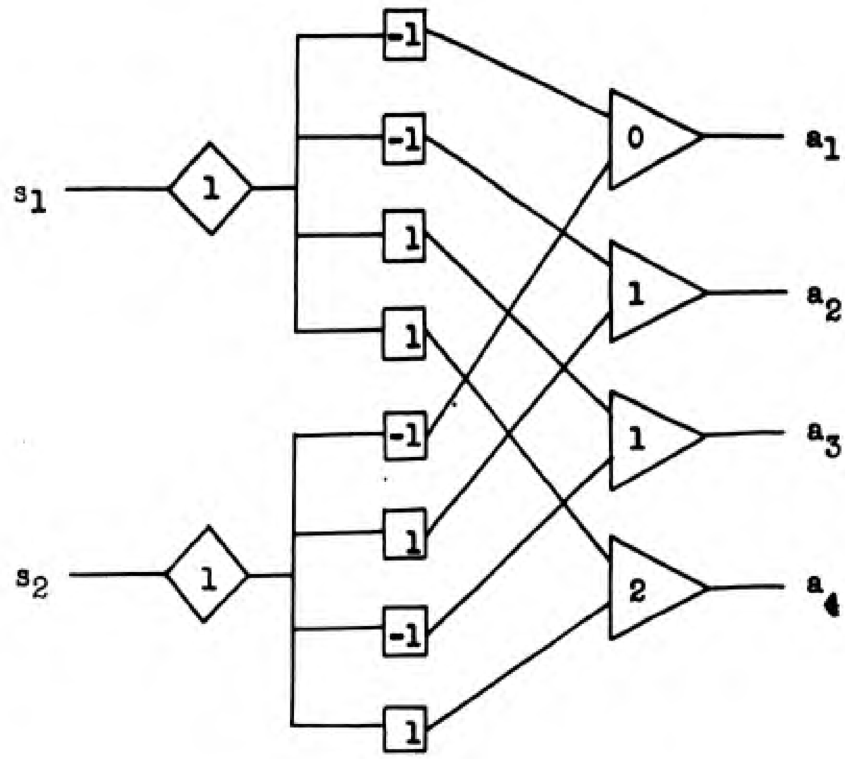


Fig. 3

Partial Network

Illustrative Example. The following example is designed to illustrate the previously defined matrices, vectors, and inequalities.

Example 4

Given: The $\underline{\underline{T}}$ matrix specified in Example 3, with the net shown in Fig. 3.

$$\underline{\underline{T}} = \begin{matrix} -1 & -1 \\ -1 & 1 \\ 1 & -1 \\ 1 & 1 \end{matrix}$$

Let $p_1 = p_2 = 1$

As shown, $\theta_1 = 0$

$\theta_2 = 1$

$\theta_3 = 1$

$\theta_4 = 2$

The possible input configurations may be represented in vector and matrix form.

		off	off
configuration		●	●
vector	$\underline{S}$ =	0	0
matrix	$\underline{\underline{S}}$ =	0	0
		0	0

Example 4 (continued)

			on	off
configuration			o	●
vector	$\underline{S}$	=	1	0
matrix	$\underline{\underline{S}}$	=	1	0
			0	0

.....

			off	on
configuration			●	o
vector	$\underline{S}$	=	0	1
matrix	$\underline{\underline{S}}$	=	0	0
			0	1

.....

			on	on
configuration			o	o
vector	$\underline{S}$	=	1	1
matrix	$\underline{\underline{S}}$	=	1	0
			0	1

.....

These represent all possible configurations of two bi-stable devices.

Example 4 (continued)

The corresponding E matrices, where $E = T\,S$, are

$S = 0 \quad 0$

$$E = \begin{matrix} -1 & -1 \\ -1 & 1 \\ 1 & -1 \\ 1 & 1 \end{matrix} \quad \begin{matrix} 0 & 0 \\ 0 & 0 \end{matrix} = \begin{matrix} 0 & 0 \\ 0 & 0 \\ 0 & 0 \\ 0 & 0 \end{matrix}$$

......

$S = 1 \quad 0$

$$E = \begin{matrix} -1 & -1 \\ -1 & 1 \\ 1 & -1 \\ 1 & 1 \end{matrix} \quad \begin{matrix} 1 & 0 \\ 0 & 0 \end{matrix} = \begin{matrix} -1 & 0 \\ -1 & 0 \\ 1 & 0 \\ 1 & 0 \end{matrix}$$

......

$S = 0 \quad 1$

$$E = \begin{matrix} -1 & -1 \\ -1 & 1 \\ 1 & -1 \\ 1 & 1 \end{matrix} \quad \begin{matrix} 0 & 0 \\ 0 & 1 \end{matrix} = \begin{matrix} 0 & -1 \\ 0 & 1 \\ 0 & -1 \\ 0 & 1 \end{matrix}$$

......

Example 4 (continued)

$\underline{S}$ = 1 1

$$\underline{\underline{E}} = \begin{matrix} -1 & -1 \\ -1 & 1 \\ 1 & -1 \\ 1 & 1 \end{matrix} \quad \begin{matrix} 1 & 0 \\ 0 & 1 \end{matrix} = \begin{matrix} -1 & -1 \\ -1 & 1 \\ 1 & -1 \\ 1 & 1 \end{matrix}$$

......

The corresponding $\underline{\pi}$ vectors, where $\underline{\pi}^T = \underline{\underline{E}}\ \underline{1}^T$, are

$\underline{S}$ =	0 0	1 0	0 1	1 1
π_1 =	0	-1	-1	-2
π_2 =	0	-1	1	0
π_3 =	0	1	-1	0
π_4 =	0	1	1	2

......

The corresponding a_i, with the θ_i as specified,

$$\theta_1 = 0$$
$$\theta_2 = 1$$
$$\theta_3 = 1$$
$$\theta_4 = 2 .$$

Example 4 (continued)

may be presented in tabular form.

$\underline{S}$ =	0 0	1 0	0 1	1 1
a_1 =	1	0	0	0
a_2 =	0	0	1	0
a_3 =	0	1	0	0
a_4 =	0	0	0	1

......

Then: The four threshold units are Boolean switching function generators, where

$$a_1 = \bar{s}_1\bar{s}_2$$
$$a_2 = \bar{s}_1 s_2$$
$$a_3 = s_1\bar{s}_2$$
$$a_4 = s_1 s_2$$

Geometrical and Graphical Methods. A geometrical method exists for the analysis of threshold units and their possible switching functions. This method is derived from the n-cube used in coding theory to define Hamming distance (Ref 2:147-160). A set of n orthogonal axes is defined in the n-space. A unit cube is constructed in the space, where the origin of coordinates represents the (0, 0, ... 0_n) state, and is one vertex of the n-cube. The remaining vertices represent the remainder of the 2^n possible states of n bi-stable devices, or n element binary vectors. Any pair of inequalities of the forms (3), and (4), defines an n - 1 dimension hyperplane in the n-space, which separates the vertices into two groups. The vertices in one group correspond to states for which a, as previously defined, is equal to one. The vertices in the second group correspond to states for which a is equal to zero. Thus, a truth table may be constructed, defining a Boolean switching function.

The n-cube, and dividing hyperplane, become difficult to represent graphically when n is greater than three. It is felt that a second method, developed in this analysis, may prove more useful for thresholded-circuit design. This method has been named the Karnaugh map-matrix method, and will be referred to as the map-matrix method. The method will be described in a stepwise fashion.

1. A Karnaugh map (Ref 3:593-599) for n variables is drawn for each variable.

2. In the map for each variable a one is entered in those areas corresponding to minterms which may have truth value one if the corresponding variable has truth value one.

3. A zero is entered in the other areas of the map for the individual variable.

4. The map for each variable is written directly as a matrix, and multiplied by the specified weight. The matrices corresponding to the weighted variables are then added.

5. Each element of the matrix formed by the addition is compared to the specified threshold. Those elements which equal, or exceed, the threshold, correspond to minterms of truth value one in the Karnaugh map of the function generated.

The matrices of one and zero elements formed from the corresponding Karnaugh maps of the functions are defined as $\underline{\underline{K}}\ (s_1, s_2, \ldots s_j, \ldots s_n)$

Example 5

Given: The Karnaugh map for two variables

	s_1	$\bar{s}_1$
s_2	$s_1 s_2$	$\bar{s}_1 s_2$
$\bar{s}_2$	$s_1 \bar{s}_2$	$\bar{s}_1 \bar{s}_2$

Then:

$$\underline{\underline{K}}\ (s_1) = \begin{matrix} 1 & 0 \\ 1 & 0 \end{matrix}$$

$$\underline{\underline{K}}\ (s_2) = \begin{matrix} 1 & 1 \\ 0 & 0 \end{matrix}$$

Let $t_{11} = w_1$, and $t_{12} = w_2$.

$$w_1 \underline{\underline{K}}\ (s_1) + w_2 \underline{\underline{K}}\ (s_2) = \begin{matrix} w_1 + w_2 & w_2 \\ w_1 & 0 \end{matrix}$$

Example 5 (continued)

Let $w_1 = 1$, and $w_2 = 1$.

$$w_1\underline{\underline{K}}(s_1) + w_2\underline{\underline{K}}(s_2) = \begin{matrix} 2 & 1 \\ 1 & 0 \end{matrix}$$

$\therefore$

If $\theta_1 = 0$

$$\underline{\underline{K}}(s_1, s_2) = \begin{matrix} 1 & 1 \\ 1 & 1 \end{matrix}$$

and

$$a_1 = I$$

$\therefore$

If $\theta_1 = 1$

$$\underline{\underline{K}}(s_1, s_2) = \begin{matrix} 1 & 1 \\ 1 & 0 \end{matrix}$$

and

$$a_1 = s_1 + s_2 \qquad \text{(OR)}$$

$\therefore$

If $\theta_1 = 2$

$$\underline{\underline{K}}(s_1, s_2) = \begin{matrix} 1 & 0 \\ 0 & 0 \end{matrix}$$

and

$$a_1 = s_1 s_2 \qquad \text{(AND)}$$

<u>Example 5</u> (continued)

Let the desired function be

$a_1 = s_1 \oplus s_2$

Then,

$$\underline{\underline{K}}\,(s_1, s_2) = \begin{matrix} 0 & 1 \\ 1 & 0 \end{matrix}$$

which requires that

$$w_1 > w_1 + w_2 \qquad (1)$$
$$w_1 > 0 \qquad (2)$$
$$w_2 > w_1 + w_2 \qquad (3)$$
$$w_2 > 0 \qquad (4)$$

but, adding (1) and (3)

$$w_1 + w_2 > 2(w_1 + w_2)$$

Since,

$$w_1 > 0$$

and

$$w_2 > 0$$

The inequalities are inconsistent, and $s_1 \oplus s_2$ is not a threshold function.

Multiple Stages

Stages of the type described may be cascaded to form networks. The following notation has been adopted.

1. The symbols for the variables in the first, or input stage, are as defined.

2. Symbols for the variables in suceeding stages will be identified by a subscript corresponding to the number of the stage. This symbol will appear at the lower left of the variable.

3. The output a_i from stage number one may be written in column vector form. This vector is defined as $\underline{A}^T$.

4. The outputs from stage one are the inputs to stage two. Thus, in the notation described,

$$ {}_2\underline{S} = \underline{A} \tag{19} $$

and, in general,

$$ {}_{k+1}\underline{S} = {}_k\underline{A} \tag{20} $$

where

$$ {}_{k+1}s_j = {}_k a_i \tag{21} $$

The function of the statistical switches may be illustrated using this notation. Let the input to the second stage be the output of the stage used in Example 4. Thus, the inputs are the minterm switching functions discussed in this example. Using the notation described,

$$_2s_1 = a_1 = \bar{s}_1\bar{s}_2$$
$$_2s_2 = a_2 = \bar{s}_1 s_2$$
$$_2s_3 = a_3 = s_1\bar{s}_2$$
$$_2s_4 = a_4 = s_1 s_2$$

Let the second stage consist of one threshold unit, where

$$_2\underline{\underline{T}} = 1 \quad 1 \quad 1 \quad 1$$

and

$$_2\theta_1 = 1$$

These conditions specify an OR gate. Then the switching function generated by the two stages is

$$_2a_1 = (f_1)\bar{s}_1\bar{s}_2 + (f_2)\bar{s}_1 s_2 + (f_3)s_1\bar{s}_2 + (f_4)s_1 s_2$$

where the f_i are one, or zero, according to the component of $\underline{A}$ produced by the configuration of the statistical switches. The probability for any particular component may be found as the probability product in Example 2. Thus, the two stages generate all sixteen switching functions of two binary inputs. This illustrates the use of the model to represent the Artron (Ref 1), mathematically.

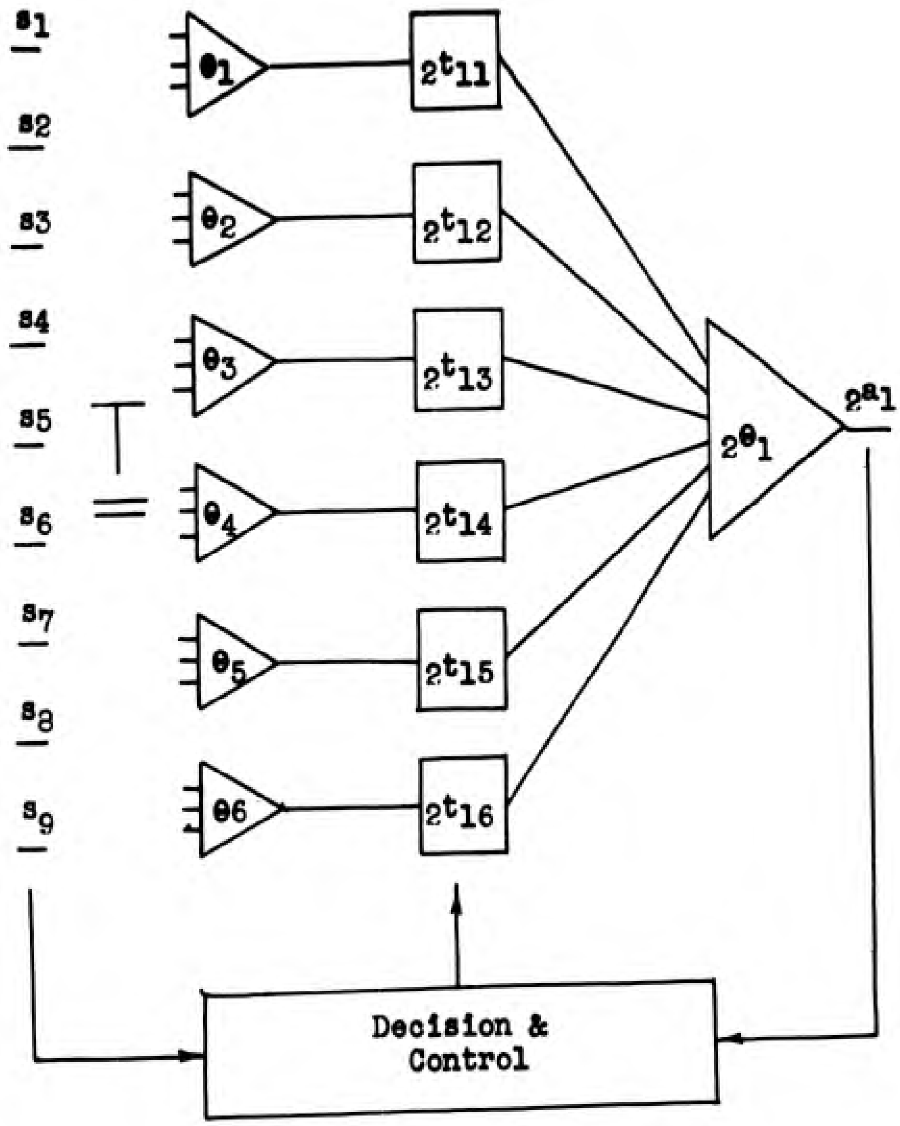


Fig. 4

Simple Perceptron

III. Problem on a Simple Perceptron

A simple Perceptron, as defined by the machine's designers (Ref 6:5), may be represented by two stages of the type described in Chapter II. Fig. 4 is a block diagram of the machine assumed for this problem, with six threshold units in the first stage, and a single unit in the second stage. It may be noted that the statistical switches do not appear in either stage. This is equivalent to assuming that all p_j, for both stages, are equal to one. The probabilistic features of the simple Perceptron occur in the construction of the first stage $\underline{\underline{T}}$ matrix, and are not time dependent. This matrix is constructed by a random process for selecting the elements. The second $_2\underline{\underline{T}}$ matrix, a single row in this case, contains the adjustable weights, whose values are set during the organization. This organization is effected by a sequential algorithm, defined as the alpha-error-correction procedure (Ref 6:8-9). The problem will be presented in two parts, the organization of the machine to correctly identify two possible patterns of a nine-light matrix, and an investigation of the ability of the organized machine to correctly identify noisy patterns.

Organization

The input patterns are two of the 2^9 possible patterns of a nine light matrix. The lights are arranged in a three-by-three square, and are identified numerically by position. Using small circles to represent the individual lamps, the identification is

1	2	3
o	o	o
4	5	6
o	o	o
7	8	9
o	o	o

If an open small circle, o , is used to indicate a lighted lamp, and a blacked in small circle, o , indicates an unlighted lamp; all possible patterns may be presented symbolically.

<u>Correct Input Patterns</u>. The two patterns selected are identified by the symbols most closely resembling their visual geometric interpretations, and are

●	o	●		o	●	o
o	o	o		●	o	●
●	o	●		o	●	o
	+				x	

called plus, and x. These patterns were chosen to have eight dissimilar elements, and one similar element. This choice was made to insure that a noisy version of one pattern would not be similar to a noisy version of the other pattern; and to test the effect of the similar element on the organization, and the noisy pattern recognition.

It is assumed that there exists a scanning device which identifies the lighted lamps by one, and the unlighted lamps by zero, and produces input vectors to the net, where element one of the input vector corresponds to the state of lamp number one, element two to the state of lamp number two, and, in general, vector element j to the state of lamp j.

The input vector produced by the plus pattern is defined as $\underline{S}_{+}$, and the vector produced by the x pattern as $\underline{S}_{x}$. Thus,

$$\underline{S}_{+} = 0,1\ 0\ 1\ 1\ 1\ 0\ 1\ 0$$
$$\underline{S}_{x} = 1\ 0\ 1\ 0\ 1\ 0\ 1\ 0\ 1$$

Probabilistic $\underline{\underline{T}}$ Matrix. In the actual Perceptron, the first stage weighted connections, represented in this problem by the first $\underline{\underline{T}}$ matrix, were made by consulting a random number table (Ref 7:1). In this problem, the random selection was simulated by restricting the t_{ij} to the values 1, 0, and -1, and rolling a die to determine the value of each element. Two sides of the die were labeled with each value, and after a thousand rolls it was assumed that the die was essentially honest, giving equal probability to each weight on any one roll. One of the $\underline{\underline{T}}$ matrices constructed in this manner is selected to illustrate certain properties of the model. This matrix is

$$\underline{\underline{T}} = \begin{matrix} 1 & 0 & -1 & -1 & 1 & 1 & 1 & 0 & 1 \\ -1 & 0 & 1 & -1 & -1 & -1 & 1 & 0 & 0 \\ -1 & 0 & 1 & 0 & -1 & 0 & 1 & 0 & -1 \\ -1 & 1 & -1 & 1 & 0 & 1 & -1 & 0 & 1 \\ -1 & -1 & 0 & 1 & -1 & 0 & -1 & -1 & 0 \\ 0 & 1 & 0 & 0 & 1 & -1 & -1 & -1 & 0 \end{matrix}$$

If the $\underline{\pi}$ vector produced by $\underline{S}_+$ is defined as $\underline{\pi}_+$, and that produced by $\underline{S}_x$ as $\underline{\pi}_x$, then by the methods of Chapter II,

$$\underline{\pi}_+^T = \begin{matrix} 1 \\ -3 \\ -1 \\ 3 \\ -2 \\ 0 \end{matrix} \qquad \underline{\pi}_x^T = \begin{matrix} 3 \\ 0 \\ -1 \\ -2 \\ -3 \\ 0 \end{matrix}$$

Threshold Values. Since the input vectors each have five elements equal to one, the elements of $\underline{\pi}$ may vary, in the general case, from five to minus five. The first stage threshold was selected as zero for all units, since this is the median value. This selection determines the outputs from the first stage threshold units. If the output a_i are written in column vector form, where the topmost element is the output of unit one, the next element is the output from the second threshold unit, and, in general, element i is the output of threshold unit i; and these vectors defined as $\underline{A}^T$, and subscripted by the pattern which produces them, then

$$\text{Given:} \quad \theta_i = 0, \qquad \underline{A}_+^T = \begin{matrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{matrix} \qquad \underline{A}_x^T = \begin{matrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{matrix}$$

It may be noted that threshold units one, three, five, and six produce the same outputs for both patterns, and that units two and four produce different outputs. The organizing ability of the machine depends on this differentiation, regardless of any possible ability of the second stage.

__The Organization Algorithm.__ At this point, the desired output, $_2a$, must be specified for each pattern. In the experimental problem the output for $\underline{S}_+$ was specified as one, and the output for $\underline{S}_x$ as zero. Thus,

$$\underline{S}_+ \implies {}_2a_1 = 1$$

$$\underline{S}_x \implies {}_2a_1 = 0$$

The machine will identify any other input pattern in one of the two classes, but these patterns are not presented during the training period.

To complete the conditions, the threshold of the second stage was specified as one, and the initial value of the second stage weighted connections as all zero. Thus,

$${}_2\theta_1 = 1$$

$${}_2\underline{\underline{T}} = 0 \quad 0 \quad 0 \quad 0 \quad 0 \quad 0$$

It was assumed that, when the input patterns were presented, the Decision and Control Unit, shown schematically in Fig. 4, compared the actual values of $_2a_1$ to the desired value of $_2a_1$, determined by $\underline{S}$, and adjusted the values of $_2\underline{\underline{T}}$. This process is sequential. In the general case, a pattern is presented, the output is checked against the desired output, and the elements of $\underline{\underline{T}}$ adjusted before the next pattern is presented. The algorithm may be written as an equation, where the secondary subscripts refer to successive states of ${}_2\underline{\underline{T}}$.

$$ {}_2t_{1j,t+1} = {}_2t_{1j,t} + (d)({}_2s_j) \qquad (22) $$

where

$$ {}_2s_j = a_1 \qquad (21) $$

and

$$ d = {}_2a_1 \text{ desired} - {}_2a_1 \text{ actual} \qquad (23) $$

Various systems for presenting the patterns were tested. It has been shown that the order of presentation does not affect the convergence of a simple Perceptron, so long as each pattern recurs in finite time (Ref 6:18-48), and it was found in the experiment that convergence occured in all cases where a solution existed, regardless of the order of presentation.

Convergence Sequence. The use of the organization algorithm will be illustrated by its use in the case of the previously stated $\underline{\underline{T}}$. The patterns will be presented alternately, beginning with $\underline{S}_+$. This was one of the possible orders of presentation used in the experiment.

Let the first input pattern be $\underline{S}_+$. The $\underline{A}_+^T$ and the corresponding ${}_2\underline{\underline{S}}_+$ are,

$$\underline{A}_+^T = \begin{matrix} 1 \\ 0 \\ 0 \\ 1 \\ 0 \\ 1 \end{matrix} \qquad {}_2\underline{\underline{S}}_+ = \begin{matrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{matrix}$$

Since all elements of ${}_2\underline{\underline{T}}$ are initially zero,

$${}_2\pi_1 = {}_2\underline{\underline{T}} \; {}_2\underline{\underline{S}} \; \underline{1}^T = 0$$

Hence, since ${}_2\theta_1 = 1$,

$${}_2a_1 = 0$$

As previously specified, the desired output for the input $\underline{S}_+$ is one. Hence,

$$d = {}_2a_1 \text{ desired} - {}_2a_1 \text{ actual} = 1 - 0 = 1$$

and the elements of ${}_2\underline{\underline{T}}$ must be changed according to the algorithm (22).

The algorithm is presented for reference.

$$ {}_2t_{1j,t+1} = {}_2t_{1j,t} + (d)({}_2s_j) \qquad (22) $$

Thus, before presenting a pattern at the second time interval,

$$ {}_2t_{11,t+1} = 0 + (1)(1) = 1 $$
$$ {}_2t_{12,t+1} = 0 + (1)(0) = 0 $$
$$ {}_2t_{13,t+1} = 0 + (1)(0) = 0 $$
$$ {}_2t_{14,t+1} = 0 + (1)(1) = 1 $$
$$ {}_2t_{15,t+1} = 0 + (1)(0) = 0 $$
$$ {}_2t_{16,t+1} = 0 + (1)(1) = 1 $$

And, when the second pattern is presented,

$$ {}_2\underline{\underline{T}} = 1 \quad 0 \quad 0 \quad 1 \quad 0 \quad 1 $$

It may be noted that only those elements of ${}_2\underline{\underline{T}}$ which received an input of one from the first stage have been changed.

The second input pattern is now presented, producing the input $\underline{S}_x$. This pattern produces an output from the first stage of threshold units, $\underline{A}_x$, which may be tranformed into a second stage input matrix, ${}_2\underline{\underline{S}}_x$, as in the previous case, where the input was $\underline{S}_+$.

Thus,

$$\underline{A}_x^T = \begin{matrix} 1 \\ 1 \\ 0 \\ 0 \\ 0 \\ 1 \end{matrix} \qquad {}_2\underline{\underline{S}}_x = \begin{matrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{matrix}$$

The sum, ${}_2\Pi_1$, is

$${}_2\Pi_1 = {}_2\underline{\underline{T}}\ {}_2\underline{\underline{S}}\ \underline{1}^T = 1\ 0\ 0\ 1\ 0\ 1 \quad \begin{matrix} 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{matrix} \quad \begin{matrix} 1 \\ 1 \\ 1 \\ 1 \\ 1 \\ 1 \end{matrix}$$

$${}_2\Pi_1 = 2$$

Since the sum is greater than the second stage threshold ${}_2\theta_1$,

$${}_2a_1 = 1$$

Hence, since the desired output for the x pattern is 0,

$$d = 0 - 1 = -1$$

and the elements of ${}_2\underline{\underline{T}}$ must be adjusted before the third pattern presentation.

The new values of $_2t_{1j}$, as specified by (22), are

$$_2t_{11,t+2} = 1 + (-1)(1) = 0$$
$$_2t_{12,t+2} = 0 + (-1)(1) = -1$$
$$_2t_{13,t+2} = 0 + (-1)(0) = 0$$
$$_2t_{14,t+2} = 1 + (-1)(0) = 1$$
$$_2t_{15,t+2} = 0 + (-1)(0) = 0$$
$$_2t_{16,t+2} = 1 + (-1)(1) = 0$$

And, when the third pattern is presented,

$$_2\underline{\underline{T}} = 0 \quad -1 \quad 0 \quad 1 \quad 0 \quad 0$$

The third pattern is the plus pattern, as specified by the alternation. As before, this pattern produces $\underline{S}_+$, $\underline{A}_+$, and $_2\underline{\underline{S}}_+$. By the previous methods,

$$_2\underline{\pi} = 1$$
$$_2a_1 = 1$$
$$d = 0$$

and the values of $_2\underline{\underline{T}}$ are unchanged. In similar fashion, when the x pattern is presented,

$$_2\underline{\pi} = -1$$
$$_2a_1 = 0$$
$$d = 0$$

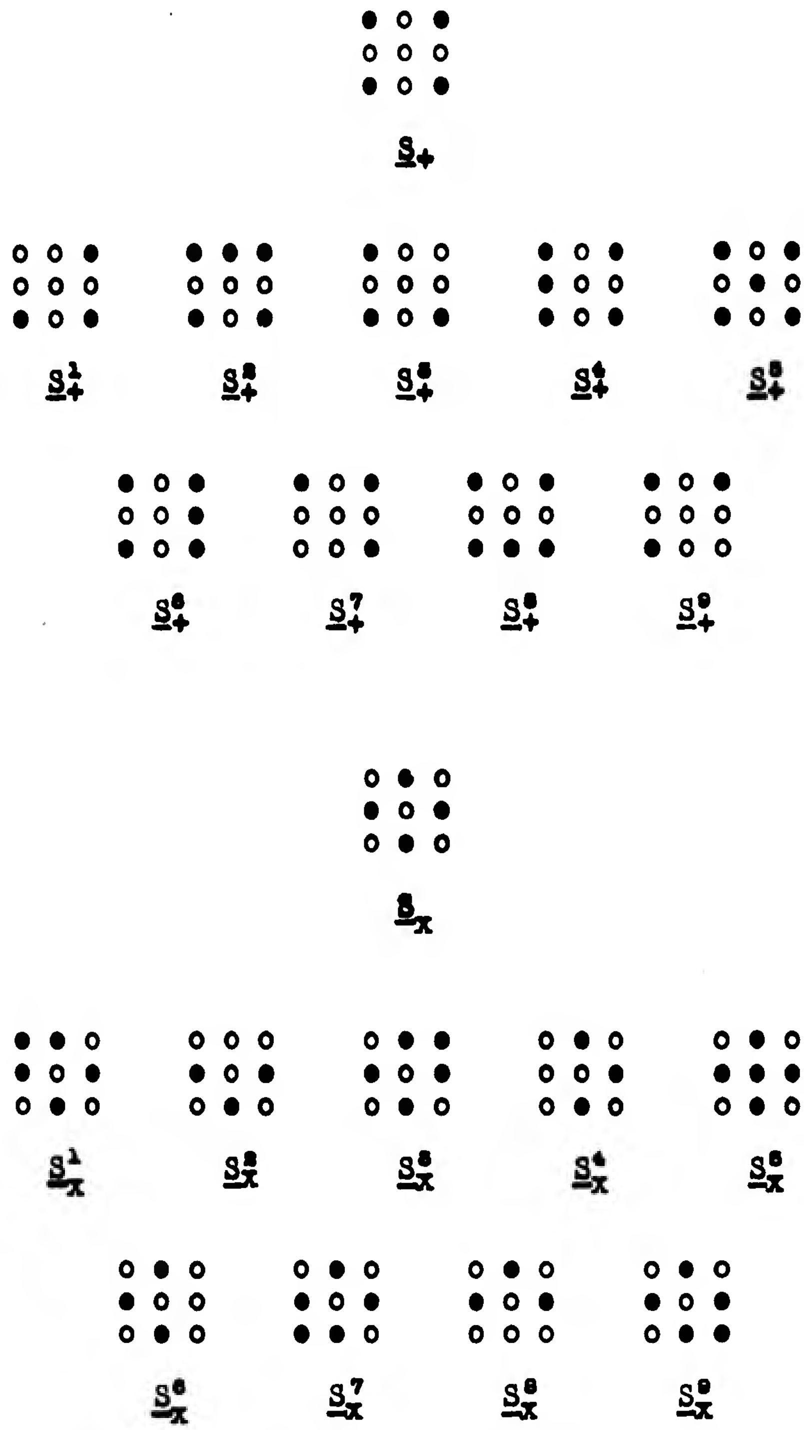


Fig. 5

Input Patterns

and the value of ${}_2\underline{\underline{T}}$ is again unchanged. Further presentations of the plus and x patterns cannot change the value of ${}_2\underline{\underline{T}}$, since, in both cases, d = 0. Therefore, the machine has converged to a solution, where the solution may be considered as the value of ${}_2\underline{\underline{T}}$.

$$ {}_2\underline{\underline{T}} = 0 \quad -1 \quad 0 \quad 1 \quad 0 \quad 0 $$

Noisy Pattern Identification

Fifteen Perceptron analogs, of the type described, were constructed and organized. All converged to a solution as defined for the two training patterns; plus, and x. Some successfully identified the nine noisy versions of each pattern created by one incorrect element. The correct patterns, their noisy versions, and the notation adopted are shown in Fig. 5. The superscript decimal on $\underline{S}$ indicates the position of the error, both in the pattern and its input vector. This notation is carried through the various $\underline{A}$, ${}_2\underline{\underline{S}}$, and ${}_2\underline{\pi}$.

Since the small integers used are suitable for mental arithmetic, and the problem required some 1800 multiplication-comparison steps, a system was developed whereby the $\underline{A}^T$ vectors produced by the various patterns could be listed in tabular form.

Each first stage $\underline{\underline{T}}$ matrix, as determined by rolling

i \ S	1	2	3	4	5	6	7	8	9
1	1	0	-1	-1	1	1	1	0	1
2	-1	0	1	-1	-1	-1	1	0	0
3	-1	0	1	0	-1	0	1	0	-1
4	-1	1	-1	1	0	1	-1	0	1
5	-1	-1	0	1	-1	0	-1	-1	0
6	0	1	0	0	1	-1	-1	-1	0

S_1^+	1	1	0	1	1	1	0	1	0

$_2\underline{\underline{T}}$
0
-1
0
1
0
0

A_+^T	A_+^1	A_+^2	A_+^3	A_+^4	A_+^5	A_+^6	A_+^7	A_+^8	A_+^9
1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0
0	0	0	1	0	1	0	0	0	0
1	1	1	1	1	1	1	1	1	1
0	0	0	0	0	0	0	0	0	0
1	1	0	0	1	0	1	1	1	1

A_x^T	A_x^1	A_x^2	A_x^3	A_x^4	A_x^5	A_x^6	A_x^7	A_x^8	A_x^9
1	1	1	1	1	1	1	1	1	1
1	1	1	0	0	1	0	0	1	1
0	1	0	0	0	1	0	0	0	1
0	0	0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0	0	0
1	1	1	1	1	0	0	1	0	1

Fig. 6

Machine 3

$\underline{\underline{T}}$ Matrix, $_2\underline{\underline{T}}$

Sample Noisy Input Card

$\underline{A}^T$ Vector Tables

the die, was written on quarter-inch, squared paper. Tables for the value of each a_1, for each pattern, were written beside the $\underline{\underline{T}}$ matrix, with a column for each $\underline{A}^T$ vector. A set of twenty, three-by-five, lined, file cards were prepared, one for each correct pattern, and one for each noisy pattern. The patterns were represented on the cards by their $\underline{S}$ vectors, which were written along the narrow side of the card, such that the nine elements of $\underline{S}$ would fall directly in line with the nine elements of a row of $\underline{\underline{T}}$. Thus, the π_1 for each pattern could be formed mentally by sliding the card down the $\underline{\underline{T}}$ matrix, compared to the zero threshold of the first stage units; and the corresponding a_1 written in the $\underline{A}^T$ vector tables.

Fig. 6 is a reproduction of the completed tables for the machine determined by the previously chosen, illustrative $\underline{\underline{T}}$ matrix. The organized value for $_2\underline{\underline{T}}$ is written in transpose form between $\underline{\underline{T}}$ and the $\underline{A}^T$ vector tables. Thus, the value for $_2\pi$ may also be formed mentally, compared to the second stage threshold value of one, and $_2a_1$ determined. A portion of one of the file cards is also illustrated in Fig. 6, in this case $\underline{S}^1_+$, the card for the $\underline{S}_+$ vector with lamp number one incorrectly lighted.

It may be noted that the machine determined by this particular $\underline{\underline{T}}$ matrix succeeds in correctly identifying all of the noisy patterns, since the $_2\pi$ for all of the noisy plus patterns is one, and the $_2\pi$ for all of the noisy x patterns is minus one, or zero.

Of the fifteen machines constructed, five succeeded in correctly identifying all of the noisy patterns. An attempt was made to find the conditions leading to this result.

A Sufficient Condition. It may be noted that element four of the $\underline{A}^T$ vectors in Fig. 6 is one for all plus patterns, and zero for all x patterns. This condition is sufficient for proper identification of noisy patterns as defined, since if $_2t_{14}$ is set to one, and all other $_2t_{1j}$ to zero; then $_2\pi$ will be one for all plus patterns, and zero for all x patterns. The occurrence of the minus one element in the $_2\underline{\underline{T}}$ illustrated is not necessary to the noisy pattern identification. If it is assumed that all three possible values of the elements of the first stage $\underline{\underline{T}}$ have equal probability of occurrence, then the probability of obtaining the sufficient condition may be computed.

Since the value of any a_i depends on a single row of $\underline{\underline{T}}$, and a constant threshold, zero, it is only necessary to examine the possible elements of a row of $\underline{\underline{T}}$. In general, this row may be any of the 3^9 permutations of a nine element vector formed from one, zero, and minus one. The permutations which meet the sufficient condition may be determined by consideration of the mathematical properties of threshold units.

The rows of $\underline{\underline{T}}$ which will meet the condition are those in which the weighted sum, π_i, cannot be driven across the threshold value, zero, by a single error in $\underline{S}$, the input vector. In equation form, since a single error can change the sum by, at most, one; where π_{i+} indicates an element of $\underline{\pi}_+$, as defined, and π_{ix} an element of $\underline{\pi}_x$,

$$\pi_{i+} \geq 1$$
$$\pi_{ix} \leq -2$$

It may be noted that row four of the illustrative $\underline{\underline{T}}$ matrix produces values for π_4 which meet these criteria.

Since π_i is the dot, or inner product, of $\underline{S}$ and row i of $\underline{\underline{T}}$, and the elements of $\underline{S}$ are defined for the plus and x input patterns, π_i depends on the values of the elements of row i of $\underline{\underline{T}}$. An attempt was made to find all permutations of the elements of a single row of $\underline{\underline{T}}$ which would satisfy the equations for π_{i+} and π_{ix}, hereafter called the criteria. Since the one elements of $\underline{S}_+$ occur in positions two, four, five, six, and eight; and the one elements of $\underline{S}_x$ occur in positions one, three, five, seven, and nine, π_{i+} and π_{ix} are not completely independent. Element t_{i5} will affect both sums. There are three possible cases, where t_{i5} is equal to zero, one, and minus one.

If t_{i5} is equal to zero, an error in $\underline{S}$ in position five cannot affect π_1 in any case. The remainder of the possible values of row i of $\underline{\underline{T}}$ are independent. The criteria are satisfied when the sum of elements t_{12}, t_{14}, t_{16}, and t_{18} is equal to or greater than one. The combinations of four ones, zeros, and minus ones, may be listed and the permutations computed.

$$t_{12} + t_{14} + t_{16} + t_{18} \geq 1$$

combinations			permutations	
1	0	-1		
1	3	0	4	
2	2	0	6	
2	1	1	12	
3	1	0	4	
3	0	1	4	
4	0	0	1	
			31	total

In similar fashion, since the sum of t_{11}, t_{13}, t_{17}, and t_{19}, must be less than, or equal to, minus two,

$$t_{11} + t_{13} + t_{17} + t_{19} \leq -2$$

combinations			permutations	
1	0	-1		
0	2	2	6	
0	1	3	4	
0	0	4	1	
1	0	3	4	
			15	total

Since the positions are independent, when t_{15} is equal to zero, there are

$$(31)(15) = 465$$

permutations which meet the criteria.

When t_{15} is equal to one, the sum over both pattern inputs is increased by one. An error in position five of $\underline{S}$ can only decrease the sum by one. As defined, only one error per noisy pattern is allowed, so an error in the fifth position, or element five, of $\underline{S}$, excludes any other error. Thus, since the sum over positions t_{12}, t_{14}, t_{15}, t_{16}, and t_{18}, need only equal or exceed one, and t_{15} is identically one in this case, the sum over t_{12}, t_{14}, t_{16}, and t_{18}, need only equal or exceed zero. The combinations and permutations among the four positions may be computed in a tabular listing, as previously.

$t_{15} = 1$

$t_{12} + t_{14} + t_{16} + t_{18} \geq 0$

combinations			permutations
1	0	-1	
0	4	0	1
1	3	0	4
1	2	1	12
2	2	0	6
2	1	1	12
2	0	2	6
3	0	1	4
3	1	0	4
4	0	0	1
			50 total

By similar reasoning, the sum over positions t_{11}, t_{13}, t_{17}, and t_{19}, must be less than, or equal to, minus three.

$t_{11} + t_{13} + t_{17} + t_{19} \leq -3$

combinations			permutations
1	0	-1	
0	1	3	4
0	0	4	1
			5 total

Since the assumption of a value for t_{15} makes the remainder of the positions independent, there are

$$(50)(5) = 250$$

permutations of the values which satisfy the criteria when t_{15} is equal to one.

In the case where t_{15} is equal to minus one, the sum over the independent even elements must be greater than, or equal to, two, giving

$$t_{12} + t_{14} + t_{16} + t_{18} \geq 2$$

combinations			permutations	
1	0	-1		
2	2	0	6	
3	1	0	4	
3	0	1	4	
4	0	0	1	
			15	total

and the sum over the independent odd elements, t_{11}, t_{13}, t_{17}, and t_{19}, must be equal to, or less than, minus one, giving,

$$t_{11} + t_{13} + t_{17} + t_{19} \leq -1$$

combinations			permutations
1	0	-1	
0	0	4	1
0	1	3	4
0	2	2	6
0	3	1	4
1	0	3	4
1	1	2	12
			31 total

Thus, there are

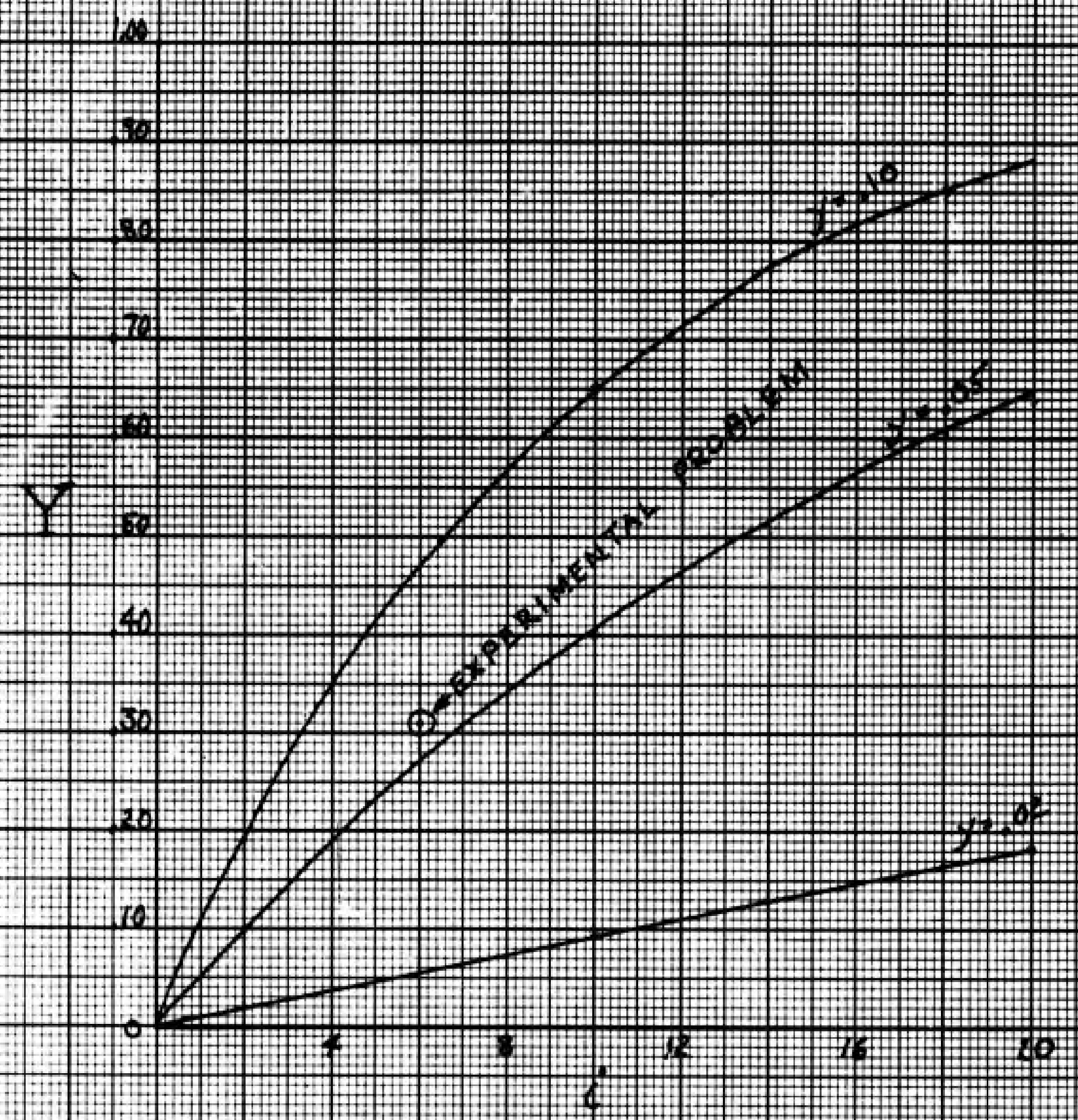
$$(15)(31) = 465$$

permutations which satisfy the criteria when t_{15} is equal to minus one.

Therefore, since all possible cases have been considered, there are

$$465 + 250 + 465 = 1180$$

possible permutations of the one, zero, and minus one, elements of one row of $\underline{\underline{T}}$ which will satisfy the conditions for noisy pattern identification, as defined.



$$Y = 1 - (1 - y)^{i}$$

Y PROBABILITY $\frac{\text{AT LEAST ONE SUCCESS}}{i \text{ TRIALS}}$

y PROBABILITY $\frac{\text{SUCCESS}}{\text{ONE TRIAL}}$

FIG. 7

If the elements of $\underline{\underline{T}}$ are selected by a random process, the probability that any row i, of $\underline{\underline{T}}$, will meet the conditions for noisy pattern identification, as defined, is

$$1180 \; / \; 3^9 \;=\; .060$$

to three significant figures.

Machine Success. Since a successful machine, where success is defined as the ability to identify noisy patterns with a single incorrect element, depends on the presence of one row of $\underline{\underline{T}}$ which meets the conditions, a formula from probability theory may be applied to determine the probability of machine success. If the probability of success on one trial is defined as y, and the probability of at least one success in i trials is defined as Y, then

$$Y \;=\; 1 \;-\; (1 - y)^i \tag{24}$$

In this case the value of y is .060, and i is equal to 6, giving

$$Y \;=\; 1 \;-\; (\,.940)^6 \;=\; .310$$

Values for (24) are plotted in Fig. 7 for small y, and values of i up to twenty.

Results. Of the ninety rows constructed in the fifteen $\underline{\underline{T}}$ matrices, six met the conditions. Of the fifteen machines, five succeeded in correctly identifying all noisy patterns. The percentage of successes in the experimental portion of the problem may be compared with the probabilities computed. It must be noted that one $\underline{\underline{T}}$ matrix constructed by the die rolling process contained two rows which met the conditions. The probability of this particular occurrence is approximately .04.

	actual percentage	computed probability
row of $\underline{\underline{T}}$	6.67	.060
machine success	33.3	.310

The agreement between percentage and probability is somewhat better than might be expected for a sample of this size. It is felt that this problem illustrates that the noise-resisting properties of thresholded nets depend on the existence of at least one threshold unit meeting conditions specified by the pattern recognition task, and that; although the probability of occurrence of such an element is small on any one trial, multiple first stage units increase the probability of success over a number of trials.

IV. Conclusions and Recommendations

The conclusions from the analysis are of a general, theoretical nature. More specialized studies of the mathematical nature of thresholded, probabilistic, self-organizing machines are recommended. The purpose of this analysis was not the selection of one particular machine, or self-organizing method, but the development of a general mathematical model, and methods based on the model. It is felt that the model described could form a comparison base for analysis of many self-organizing systems.

The experimental problem illustrates one application of the model and methods. It also indicates a tendency of the threshold unit to assign equal truth values to adjacent minterms of the Karnaugh map, where the central minterm is assigned truth value one, or zero, by the training process. This tendency gives a thresholded pattern recognition device some ability to resist noise. The theory of this tendency is developed further in Appendix A.

Conclusions

The conclusions, based on the study, are:

1. The mathematical model is adequate for representation of the Artron (Ref 1), and the Perceptron (Ref 6). If the

connection weights are allowed to vary continuously, the model is sufficient to represent the Adaline (Ref 8).

2. The vectors, matrices, and methods, as defined, allow a step-by-step analysis of the function, or functions, generated, resulting in a truth table determination of these functions.

3. If it is desired that a single output unit generate several functions of the input variables, according to some probability distribution, the use of the statistical switch is indicated.

4. The experimental problem indicates that the noise-resisting properties of a simple Perceptron may depend on the existence of a single threshold unit possessing the ability to correctly identify noisy patterns. It has not been shown that the existence of such a unit is necessary, but that this existence is sufficient.

5. The close agreement of the experimental percentages and the permutation calculations supports the single unit hypothesis.

6. If it is possible to estimate the probability of obtaining a successful single unit, the total number of units required for a given machine success probability may be found from equation 23. Probability tables contain this function.

Recommendations

The following specialized studies, based on this analysis, are recommended.

1. It is felt that a model in which each connection has both a weight and a closure probability may better simulate the neuron cell. A study based on this model is recommended.

2. It is felt that the map-matrix method is more suited to switching circuit design than other methods. An extension of Appendix A as a design tool is recommended.

3. Complex pattern recognition tasks may be analyzed for probabilistic results in the manner illustrated in Chapter III. A computer program for the permutation analysis may be possible. However, it is felt that a thresholded machine, similar to the simple Perceptron, could provide more direct results. A statistical noise study, using an Adaline (Ref 8), with random weight selection, is recommended.

Bibliography

1. Carne, E. B., et al. A Self-Organizing Binary Logical Network. Published Technical Paper. Falls Church, Virginia: Melpar, Inc., July 1961.

2. Hamming, R. W. "Error Detecting and Error Correcting Codes." The Bell System Technical Journal, 26:147-160 (April 1950)

3. Karnaugh, M. "Synthesis of Combinational Logic Circuits." Communications and Electronics, 9:593-599 (November 1953)

4. McCulloch, W. S. "Background." in Bionics Symposium: Living Prototypes--The Key to New Technology. Wright Air Development Division Technical Report 60-600. Wright-Patterson Air Force Base: Wright Air Development Division, 1960, pp. 51-54.

5. -----. "Discussion." in Bionics Symposium: Living Prototypes--The Key to New Technology. Wright Air Development Division Technical Report 60-600. Wright-Patterson Air Force Base: Wright Air Development Division, 1960, pp. 491-493.

6. Rosenblatt, F. On The Convergence of Reinforcement Procedures in Simple Perceptrons. Cornell Aeronautical Laboratory Report No. VG-1196-G-4. Buffalo, New York: Cornell Aeronautical Laboratory, Inc., February 1960.

7. The Perceptron. Published Speech. Cornell Aeronautical Laboratory, Inc., Physics Division, December 1960.

8. Widrow, B., et al. "Birth, Life, and Death in Micro-Electronic Systems." Institute of Radio Engineers Transactions of the Professional Group on Military Electronics, MIL-5:191-201 (July 1961)

Appendix A

The Karnaugh Map-Matrix Method

The Karnaugh map-matrix method is a system for the analysis of threshold functions, and the design of thresholded Boolean function generators. It is based on the direct transfer of truth values from map areas to matrix elements, where the truth values are represented by numerical ones and zeros. The matrices are defined as follows.

Let a Karnaugh map for n variables be drawn. Any single variable may be represented by entering ones and zeros in the map, where a one is entered in minterm areas which may have truth value one if the variable has truth value one, and a zero in minterm areas which must have truth value zero if the variable has truth value one. The ones and zeros may be written directly as a matrix, where these numbers retain their positional relations. Thus, n matrices may be constructed for the n variables. These matrices are defined as $\underline{\underline{K}}\,(s_j)$, where s_j is the jth variable, and $\underline{\underline{K}}$ indicates the described matrix formation operation.

Each matrix is then multiplied by the weight, or amplification factor, assigned to the corresponding variable. Thus, given that the jth variable is multiplied by the weight

t_{ij} during transmission to threshold unit i, the weighted matrix is $t_{ij}\ \underline{\underline{K}}\ (s_j)$.

The weighted matrices may be added to form a single matrix, $\sum_{j=1}^{n} t_{ij}\ \underline{\underline{K}}\ (s_j)$. The elements of this matrix are compared to the threshold of unit i, defined as θ_i. The solution matrix, defined as $\underline{\underline{K}}\ (s_1, s_2, \ldots s_j, \ldots s_n)$, is formed by entering ones and zeros as determined by the comparison. If an element of $\sum_{j=1}^{n} t_{ij}\ \underline{\underline{K}}\ (s_j)$ is greater than, or equal to, the threshold, θ_i, the corresponding element of $\underline{\underline{K}}\ (s_1, s_2, \ldots s_j, \ldots s_n)$ is one. If the element of the weighted sum matrix is less than the threshold, the corresponding element of $\underline{\underline{K}}\ (s_1, s_2, \ldots s_j, \ldots s_n)$ is zero. The ones and zeros of $\underline{\underline{K}}\ (s_1, s_2, \ldots s_j, \ldots s_n)$ may be transferred to a Karnaugh map for the n variables, maintaining their positional relationship. The map then determines the function generated by the threshold unit.

This method is sufficient to determine the function generated by a given threshold and set of weights. A second procedure, based on the map-matrix, is recommended for those design problems in which the function is specified, and the weights and threshold must be determined.

Any element of $\sum_{j=1}^{n} t_{ij}\ \underline{\underline{K}}\ (s_j)$ may be maximized with respect to the other elements by consideration of the corresponding minterm area of the Karnaugh map. The weight one is assigned to those variables which are un-negated in the minterm. The weight minus one is assigned to the negated variables. Thus, if there exist x un-negated variables in the minterm, the value of the corresponding element of $\sum_{j=1}^{n} t_{ij} \underline{K} (s_j)$ will be x. Elements corresponding to adjacent minterms will have value x - 1, and, in general, elements corresponding to minterms at distance h will have value x - h. If the threshold of the unit, θ, is set at x, then the unit will generate the minterm function.

This procedure may be extended to maximize elements corresponding to adjacent minterms of the Karnaugh map, and has been systemized into a design algorithm. The algorithm has been tested on threshold functions of four variables, and will be illustrated by an example.

Let the output of the threshold unit be a, and the desired function be:

$$a = \bar{s}_1(s_2s_3 + s_2s_4) + s_2s_3s_4$$

Expand the function in a minterm canonical form.

$$a = s_1s_2s_3s_4 + \bar{s}_1s_2s_3s_4 + \bar{s}_1s_2\bar{s}_3s_4 + \bar{s}_1s_2s_3\bar{s}_4$$

Construct a table with a single column for each variable, and assign those values, one, and minus one, which would maximize the elements of $\sum_{j=1}^{n} t_{ij} \underline{K} (s_j)$ corresponding to each individual minterm of the expansion.

minterm	s_1	s_2	s_3	s_4
$s_1s_2s_3s_4$	1	1	1	1
$\bar{s}_1s_2s_3s_4$	-1	1	1	1
$\bar{s}_1s_2\bar{s}_3s_4$	-1	1	-1	1
$\bar{s}_1s_2s_3\bar{s}_4$	-1	1	1	-1

The algebraic sum of each column determines the weight to be assigned to each variable. Thus,

$$\begin{array}{lcr} t_{11} & = & -2 \\ t_{12} & = & 4 \\ t_{13} & = & 2 \\ t_{14} & = & 2 \end{array} \quad \div\ 2 = \quad \begin{array}{cr} = & -1 \\ = & 2 \\ = & 1 \\ = & 1 \end{array}$$

The division illustrates one method for the reduction of the column sums to minimum integral values. While it reduces the weight values, it has no effect on the maximization of the desired elements. A second method is the reduction of the sums by inspection, retaining only the relative magnitude of the assigned weights.

A general mathematical treatment of the map-matrix method is being developed, but is incomplete. In all cases tested, for four variables, the method correctly specifies the weights.

Vita

Robert Bradley Stuart was born on [redacted] [redacted], the son of Bradley Robert Stuart and Sue [redacted] Stuart. He was graduated from the United States Naval Academy in June 1955, awarded the degree of Bachelor of Science, and commissioned as Second Lieutenant in the United States Air Force. He received the rating of Pilot in June 1957. His military assignment prior to his coming to the Air Force Institute of Technology was as Pilot, 22nd Bomb Wing, Strategic Air Command.

Permanent address: [redacted]

This thesis was typed by Capt. and Mrs. Robert B. Stuart

UNCLASSIFIED

UNCLASSIFIED