BIRTH, LIFE, AND DEATH IN MICROELECTRONIC SYSTEMS

by

B. Widrow
W. H. Pierce
J. B. Angell

Technical Report No. 1552-2/1851-1
May 30, 1961

Prepared under
Air Force Contract AF33(616)-7726
and
Office of Naval Research Contract Nonr 225(24), NR 373 360
jointly supported by the U.S. Army Signal Corps,
the U.S. Air Force, and the U.S. Navy (Office of Naval ResearchL

Solid-State Electronics Laboratory
Stanford Electronics Laboratories
Stanford University
Stanford, California

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

I. INTRODUCTION

Many techniques for shrinking the size, weight, and power consumption of electronic components, circuit assemblages, and functional units have been proposed, demonstrated, and exploited in recent months.[1] In some cases, individual components retain their separate identities, and are interconnected by relatively standard, although sophisticated, wiring techniques. In certain of the more far-reaching approaches, a number of recognizable devices are combined into an integrated structure or array, or even into a complex structure in which the interconnecting medium between the devices contributes to the electrical properties of the structure.[2,3] Even more elegant, although speculative, techniques have been proposed in which large numbers of components are formed *en masse* in thin-film patterns on appropriate substrates by evaporative or ion-beam deposition or by electron-beam micromachining.[1,4] Thus, technological skills are leading us toward ever increasing density of components, with decreased cost per component.

One of the great hopes for microelectronics technology is that it will provide improved dependability of highly complex electronic systems. It is thought that this improvement will be the result of one or more of the following factors:

(1) Eliminating, or greatly reducing, the number of mechanically made electrical connections within a system,

(2) Increasing component uniformity during manufacture,

(3) Taking advantage of the ease of isolating physically small systems from damaging environments, and

(4) Exploiting the ease with which large numbers of moderately reliable components might be manufactured in systems whose reliability is ensured through a reasonable amount of redundancy.

[1] Superscript numerals refer to reference at end of report.

A part of this paper is concerned with the application of adaptive techniques to redundant systems in order to enhance the effectiveness with which redundancy improves dependability. How such techniques might be used to create systems which are trained by experience, rather than designed explicitly to perform given tasks or to tolerate low component yield at manufacture, is also considered. Finally, consideration is given to the characteristics of components out of which such adaptive systems could be built, and examples of devices showing the desired characteristics are discussed.

The *'birth'* of an electronic data-processing system is achieved when the system components or parts are so assembled that the desired system performance is obtained. With present-day design and assembly techniques, this achievement demands that the individual parts all function and be flawlessly interconnected. With appropriate redundance,[5] majority vote,[6] or weighted vote,[7] the need for initial perfection is considerably relaxed.

During *'life'*, the possibility of random errors in the signals of a data processing system can be greatly diminished through the use of paralleled (redundant) systems or parts, particularly if the system is continually adapted so as to place little confidence in those parts which are most inclined to make mistakes.*

At *'death'*, a system is incapable of further correct functioning. With present-day nonredundant design, the failure of any component would cause the death of the system, were it not for the external substitution of replacements. The use of redundancy, or redundancy plus adaptation, can greatly defer the death of a system in which individual parts or subsystems cannot be replaced.

*It is not the intent of this paper to propose that an adaptive vote-taker would necessarily provide an optimum political system.

Various schemes for effecting birth, postponing death, and providing dependable life of a large data handling system have been proposed recently.[8] For a system of relays, the use of redundant relay contacts in series, parallel, or lattice connections (the most suitable connections can be prescribed from statistical knowledge of the manner in which component failures occur) was described by Shannon and Moore.[5] The majority vote-taker of von Neumann can be used in a binary data-processing system to given correct signals at any point of a redundant system, provided the majority of inputs to the vote-taker are correct.[6] The adaptive vote-taker described here is a more elegant (often optimum) decision element for exploiting redundancy efficiently. The use of redundant systems (complete machines) was considered by Rosenheim and Ash, who compared the advantages of keeping one or more duplicate machines inactive but ready for operation with the advantages of running redundant machines independently and switching outputs when one fails.[9]

We are not yet prepared to prescribe quantitatively the level (components, subsystems, or complete systems) at which redundancy can be applied most efficiently. If one considers only the statistics of the problem, and ignores relative costs, it is probable that, with adaptive vote-takers, the size of individual subassemblies should be such that their reliability is comparable with that of the vote-taker; the various economic factors involved could appreciably alter this conclusion. Flehinger has shown that the reliability improvement depends more on the degree of redundancy than on the system level at which redundancy is applied when the individual parts are very reliable; when the parts are unreliable, redundancy must be applied at the level of relatively small subsystems.[10]

Improved system dependability is not the only promise of adaptive logic in microelectronics. Equally intriguing are the possibilities of systems whose function can be continually altered to optimize their performance on the basis of the statistics of past experience (for example, adaptive pattern recognizers) and of systems that are initially trained by experiences---rather than designed---to their desired function. The latter possibility is appealing because it implies not only that systems could be trained to ignore manufacturing defects, but that various system functions could be achieved using similar, or even the same, microelectronic fabricating facilities.

The heart of the system philosophies proposed in this report is an adaptive vote-taker, whose function is to determine whether or not the weighted sum of its input signals exceeds a given threshold. The vote-taker comprises variable-gain (weighting) elements plus a summing element and a threshold detector. The vote-weight assigned to each input must be stored in the vote-taker, thereby giving it memory. It is most desirable that permanent, analog quantities be remembered (stored), although transient or quantized memories might have economic advantages and be functionally adequate. Permanent, analog memory probably cannot be achieved electronically, except possibly via the persistent current stored in a superconducting ring; more likely, ionic or magnetic effects involving the translation or rotation of atoms will prove optimum for providing such memory. Certain electrochemical and magnetic phenomena described below have already been studied, and look promising; however, much remains to be done before the function of variable gain with memory can be achieved dependably and economically.

For microelectronic applications, the average power dissipation per element should be extremely small. This rule also applies to the variable-gain elements of an adaptive system, with the possible exception that during adaptation (which would typically occur infrequently during operation) higher power levels could be applied to the variable-gain elements. Fortunately, the redundancy and adaptation introduced to expidite birth and postpone death of an adaptive system also provide protection from random errors during operation. Consequently, the circuits in such a system can operate with a lower signal-to-noise ratio than those of a nonredundant system, so that the average power dissipation per component is correspondingly reduced.

II. IMPROVED DEPENDABILITY WITH REDUNDANCY PLUS ADAPTATION

When redundant digital circuits are used in an appropriate configuration, yield factors can be made arbitrarily close to 100 per cent, and error rates can be made arbitrarily close to zero, regardless of system complexity. In these configurations, restoring organs provide reliable output information from redundant but less reliable input information. Restoring organs were first proposed by von Neumann, who defined their function, indicated their placement in redundant systems, and demonstrated their universality in digital networks.[6] Let ρ be the number of circuits in a redundant network in which the same digit is independently computed; ρ will be called the redundancy of the circuit. A restoring organ is a circuit with ρ redundant inputs and ρ redundant outputs. The function of the restoring organ is to use the redundant information in the ρ inputs, each of which is the same digit, to make each of the ρ outputs more reliable.

The internal structure of a restoring organ need not be the complicated structure proposed by von Neumann. Simple restoring organs may be composed of decision elements, as shown in Fig. 1. Each decision element furnishes one of the outputs of the restoring organ. Every decision element uses information from each input, thereby making efficient use of the redundant information in the inputs. To construct a redundant circuit from a circuit without redundancy, insert ρ separate logical devices where one appears in the original. Then insert a restoring organ after each of the logical operations. The simple circuit of Fig. 2(a) is made redundant, as shown in Fig. 2(b), using the restoring organs illustrated in Fig. 1. The arrangement permits unreliable decision elements to be used, because an error in a decision element can introduce no more erroneous information in the circuit than an error in the circuit which follows the decision element.

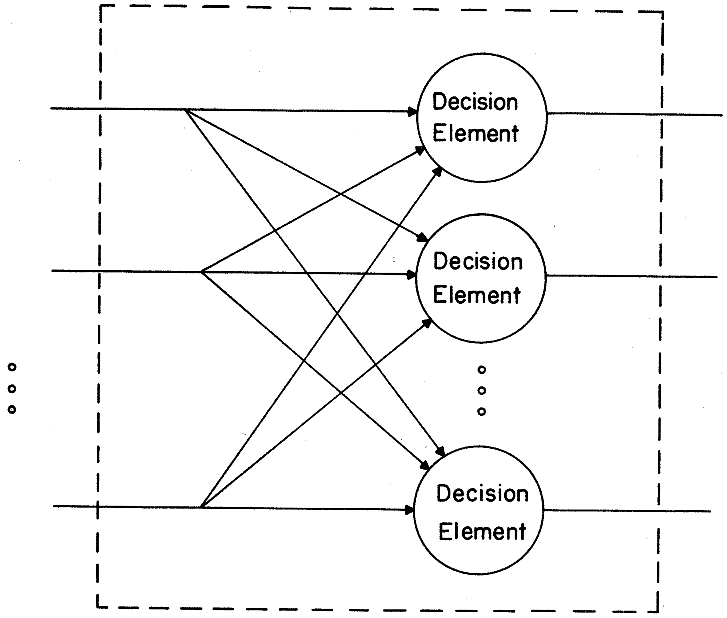
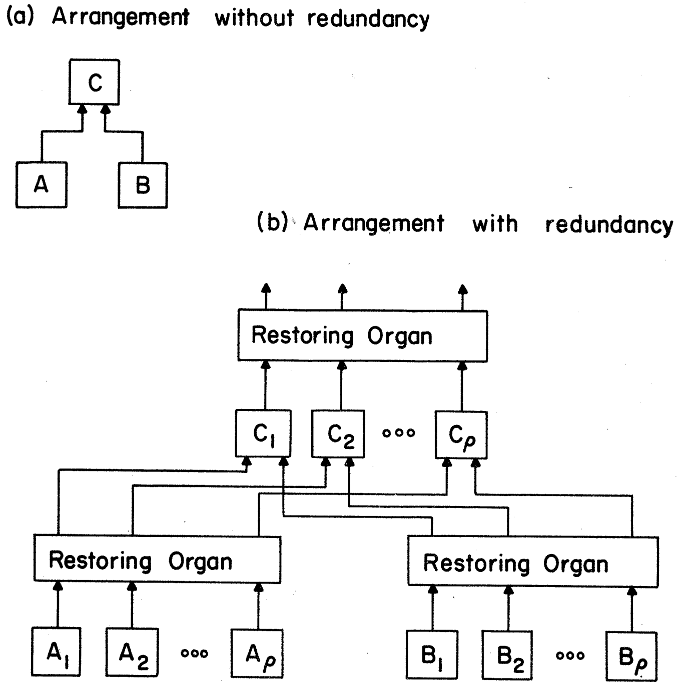


FIG. 1. A RESTORING ORGAN USING DECISION ELEMENTS. THE REDUNDANT INFORMATION ON THE INPUT LINES ON THE LEFT IS USED TO MAKE MORE RELIABLE INFORMATION ON THE OUTPUT LINES ON THE RIGHT.


(a) Arrangement without redundancy

(b) Arrangement with redundancy

FIG. 2. PLACEMENT OF RESTORING ORGANS IN A REDUNDANT CIRCUIT, ILLUSTRATED FOR A SIMPLE TREE CIRCUIT.

The simplest decision element for binary systems is a majority-rule circuit. When each of the inputs to a majority-rule decision element has error probability λ, the probability that a majority of independent inputs will be in error, λ_D, (assuming ρ odd) is the following sum of terms of the binomial distribution.

$$\lambda_D = \sum_{n=0}^{\frac{\rho-1}{2}} \binom{\rho}{\frac{p+1+2n}{2}} (1-\lambda)^{\frac{\rho-1-2n}{2}} \lambda^{\frac{\rho+1+2n}{2}} \qquad (1)$$

λ_D is bounded from below by the term for $n = 0$. λ_D can be bounded from above by an infinite geometric series in which the k^{th} term is $[\lambda/(1-\lambda)]^{k-1}$ times the term for $n = 0$. Therefore, for $\lambda < 0.5$,

$$\binom{\rho}{\frac{\rho+1}{2}} (1-\lambda)^{\frac{\rho-1}{2}} \lambda^{\frac{\rho+1}{2}} \leq \lambda_D \leq \frac{\binom{\rho}{\frac{\rho+1}{2}} (1-\lambda)^{\frac{\rho-1}{2}} \lambda^{\frac{\rho+1}{2}}}{1 - \frac{\lambda}{1-\lambda}} \qquad (2)$$

The logarithm of λ_D is plotted in Fig. 3 versus the redundancy, ρ, for several values of λ. In order to give an idea of the error probabilities involved, the mean time between errors has been plotted on the right side of the graph, assuming 10^5 calculations per second. When each input makes one error in 200, note that the mean time between errors in the output decision is a century with a redundancy of only 15.

If the inputs to a decision element are not all equally reliable, an improvement in system reliability may be obtained by distinguishing between inputs with different error probabilities. Pierce has shown that the binary number which is more likely to be correct may be obtained from independent inputs by a circuit which takes a weighted vote, such as shown in Fig. 4.[7] Let x_i be the binary digit, +1 or -1, which is on the i^{th} input. Each value of x_i is multiplied by a_i in the device shown

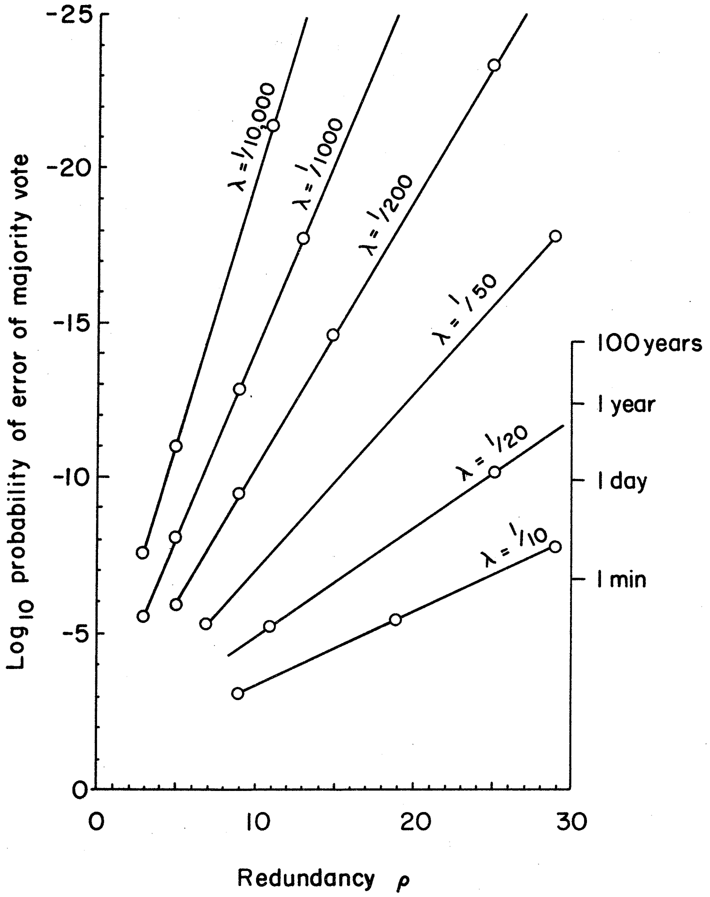


FIG. 3. LOGARITHM OF ERROR PROBABILITY OF MAJORITY VOTE VERSUS REDUNDANCY, FOR INPUTS WITH ERROR PROBABILITY λ, FOR ODD ρ. THE TIME SCALE GIVES MEAN INTERVAL BETWEEN ERRORS FOR 10^5 CALCULATIONS PER SECOND.

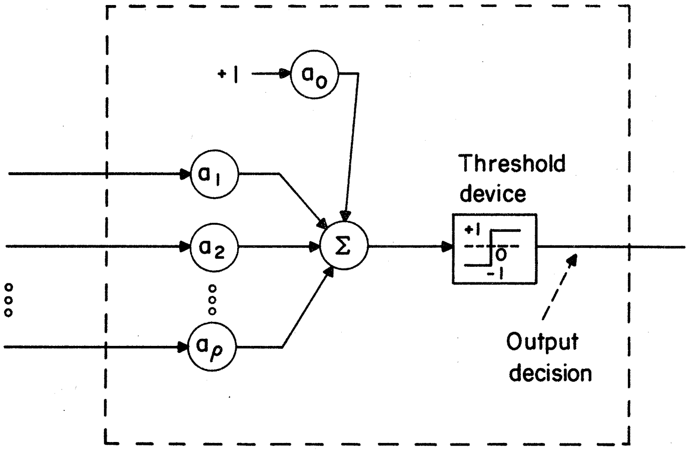


FIG. 4. A LINEARLY SEPARABLE DECISION ELEMENT.

as a triangle, giving the output $x_i\ a_i$. The values $x_i a_i$ are summed in the summing device, shown with a Σ, so that

$$\text{output of summing device} = a_o + \sum_{i=1}^{\rho} a_i x_i \tag{3}$$

If the output of the summing device is positive, the nonlinear device causes the value of output decision to be +1; if the output is negative, the nonlinear device causes the output decision to be -1. If the weighting factors, a_i, are

$$a_i = \log \frac{p(i^{th} \text{ input is correct})}{p(i^{th} \text{ input is incorrect})} \tag{4}$$

and the bias term, a_o, (which depends on relatively how often +1 is the correct answer) is[*]

$$a_o = \log \frac{\textit{a priori} \text{ probability of } +1}{\textit{a priori} \text{ probability of } -1} \tag{5}$$

then the output decision will be the binary digit more likely to be correct.[**]

[*] A. Bayes’ decision (assuming zero loss for a correct output decision) is made by adding the log of the ratio of relative losses for incorrect decisions.

[**] The proof is based upon the following equation for conditional probabilities:

$$\log \frac{p(x|A_1 \ldots A_n\ B_1 \ldots B_m)}{p(\overline{x}|A_1 \ldots A_n\ B_1 \ldots B_m)} = \log \frac{p_o(x)}{p_o(\overline{x})} +$$

$$\sum_{i=1}^{n} \log \frac{p(A_j \text{ is correct})}{p(A_j \text{ is incorrect})} - \sum_{j=1}^{m} \log \frac{p(B_j \text{ is correct})}{p(B_j \text{ is incorrect})}$$

where x is any Boolean proposition (here x = ‘+1 is correct’)

$\overline{x}$ is the complement of x (here $\overline{x}$ = ‘-1 is correct’)

$A_1 \ldots A_n$ are observations favorable to x

$B_1 \ldots B_n$ are observations favorable to $\overline{x}$.

$p_o(x)$ is the *a priori* probability of x

$p_o(\overline{x})$ is the a priori probability of $\overline{x}$.

Assuming independence of errors in the inputs, the equation follows from manipulation with conditional probabilities (Bayes’s Law) and the simple properties of logarithms.

The decision element of Fig. 4 can be made adaptive by circuits which estimate the probabilities in the expression for the optimum a_i, and then use this estimate to adjust the actual vote weights in the decision element. Defective inputs are automatically eliminated from the vote by a being given a vote weight of zero. In general, the more reliable inputs to a decision element are given greater votes.

The error probabilities of each input to a decision element can be estimated by counting errors between each input and the correct answer. The correct answer could initially be supplied for this purpose. However, if the output decision of the decision element is very reliable--- as it would be in a digital computer---then the output decision could be used as if it were the correct answer for the purpose of counting errors. Thus, a decision element with inputs which are initially very reliable could maintain a reliable output as the inputs fail one by one. Whenever an input failed, it would disagree with the output decision of the majority, and be thereby classified as defective and given a vote weight of zero.

Adaptation may often be exchanged for redundancy, and vice versa, without change of reliability. The rate of exchange will be discussed quantitatively in Appendix I.

The initial yield and the expected lifetime of a redundant system depend upon the complexity of the system, the amount of redundancy, and the type of adaptation used, if any. For instance,* consider a system with 100 different stages, each of which must work for the system to work. Assume that the probability that any stage works is 90 percent. Without redundancy, the probability of successful manufacture is 2.7×10^{-5}. When majority-rule decision elements are used, a redundancy of 3 gives a probability of successful manufacture of 0.058, while a redundancy of 9 gives

*The examples are special cases of the combinatorial formulas of Appendix I.

0.90. If adaptive decision elements which require only one good input are used,* a redundancy of 3 gives a probability of successful manufacture of 0.91, while a redundancy of 9 gives a probability of $1 - 10^{-7}$.

The lifetime of a system with many different stages is extended by the use of redundancy; the lifetime may be extended even beyond the median lifetime of each component if adaptation is also used. For example, assume that a system with 100 stages is made from stages which have a survival probability of e^{-t}. The survival probability of a system which uses majority-rule decision elements is shown as a function of time in Fig. 5 for $\rho = 3$ and $\rho = 9$; the system without redundancy is shown by the curve labeled $\rho = 1$. A system which uses perfectly adaptive decision elements, which require only one good input, has the survival probability shown in Fig. 6. Redundant systems have an initial period of high reliability, which makes them especially valuable in critical applications. The curves from Figs. 5 and 6 have been replotted in Fig. 7 on logarithmic scales in order to demonstrate this fact.

The compound problem of initial yield and lifetime can be treated simultaneously. Suppose the system with 100 stages has an initial yield probability of 95 per cent per stage, and a survival probability of $e^{-t/\tau}$ thereafter. The system can be analyzed, using Fig. 5, 6, and 7 by letting the time on these figures be a variable t', where $t' = \frac{t}{\tau} - \ell n\ 0.95$. Thus, at $t = 0$, Fig. 5 shows the probability that the nonredundant system has survived manufacture is below 0.2 (actually it is 0.006), while the probabilities that the majority-rule systems have survived manufacture are 0.55 for $\rho = 3$ and $1 - 0.003$ for $\rho = 9$. The adaptive systems have yields of $1 - 7x10^{-4}$ for $\rho = 3$ and $1 - 4x10^{-6}$ for $\rho = 9$. Given the fact that the systems survived manufacture, the median lifetime of each system is τ times the interval between $\ell n\ (1/0.95)$ and the value of time on the graphs

*This assumption is not necessarily impractical. External correct signals could be supplied temporarily in order to find the correct inputs to a decision element, even when they are in the minority.

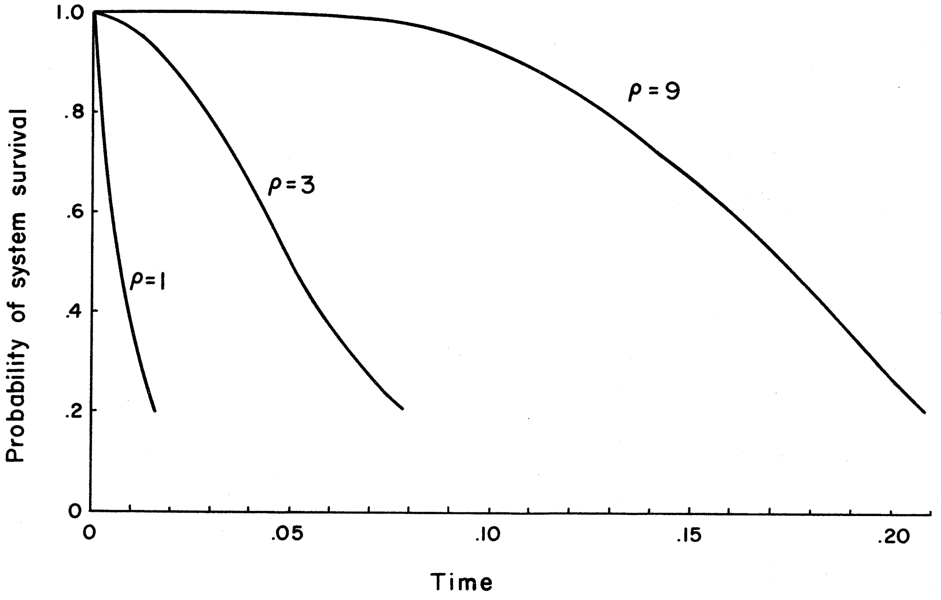


FIG. 5. SURVIVAL PROBABILITY AS A FUNCTION OF TIME FOR A SYSTEM OF 100 STAGES, WITH A REDUNDANCY OF ρ, USING MAJORITY-RULE DECISION ELEMENTS. EACH STAGE HAS SURVIVAL PROBABILITY e^{-t}.

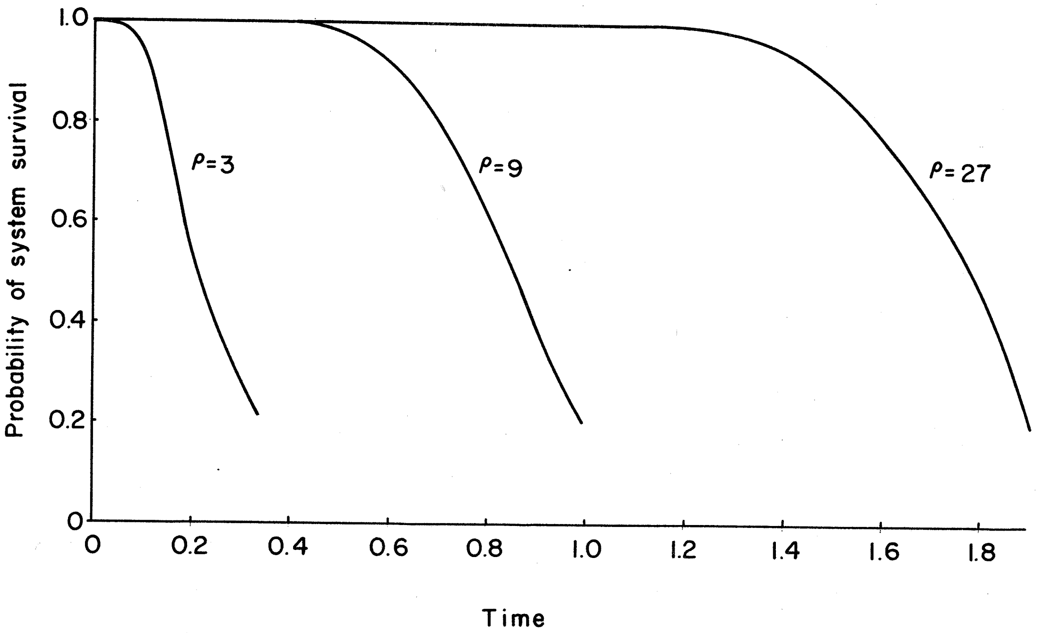


FIG. 6. SURVIVAL PROBABILITY AS A FUNCTION OF TIME FOR A SYSTEM OF 100 STAGES, WITH A REDUNDANCY OF ρ, USING PERFECTLY ADAPTIVE DECISION ELEMENTS. EACH STAGE HAS SURVIVAL PROBABILITY e^{-t}.

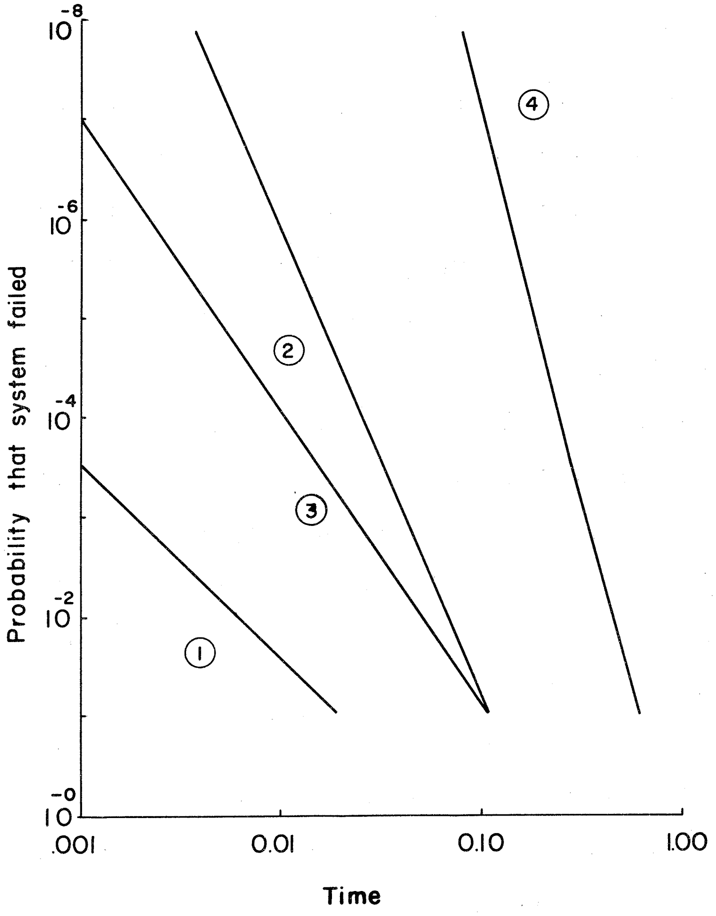


FIG. 7. LOGARITHMIC PLOTS WHICH SHOW THE HIGH RELIABILITY OF REDUNDANT, AND REDUNDANT ADAPTIVE, SYSTEMS DURING THE INITIAL PART OF LIFETIME. ALL CURVES ARE FOR A SYSTEM OF 100 STAGES WITH A REDUNDANCY OF ρ, CONNECTED BY DECISION ELEMENTS; ASSUMING EACH STAGE HAS SURVIVAL PROBABILITY e^{-t}. CURVE 1--ρ=3, MAJORITY-RULE DECISION ELEMENTS; CURVE 2--ρ=9, MAJORITY-RULE DECISION ELEMENTS; CURVE 3--ρ=3, PERFECTLY ADAPTIVE DECISION ELEMENTS; CURVE 4--ρ=9, PERFECTLY ADAPTIVE DECISION ELEMENTS.

for which the survival probability is 0.5 times the probability of surviving manufacture. Thus the majority-rule systems which survived manufacture would have a median lifetime of 0.015 τ for ρ = 3, and 0.12 τ for ρ = 9. The adaptive systems would have a median lifetime of 0.16 τ for ρ = 3, 0.80 τ for ρ = 9, and 1.72 τ for ρ = 27.

III. ADAPTIVE LOGIC

In the previous section, it was seen that adaptive decision elements, also called vote-takers, dispersed throughout microelectronic systems, are like automatic repairmen constantly on duty in their respective locales, always ready to delete parts that become defective. This type of self-repair makes optimal use of the remaining functioning components, and is especially applicable to systems of fixed logical structure. A new type of logic, adaptive logic, is being devised that promises to play a significant role in the future development of computers. This type of logic is not designed in detail in the usual way. Instead, it can learn to function by being trained by the designer, or it can spontaneously learn from its environment. In a sense, such systems are inherently reliable. They can adapt to their own internal failures. Systems containing adaptive vote-takers are bridges between conventional fixed-logic systems and systems adaptive 'from the ground up'.

A self-contained automatically-adapted logical element called the ADALINE 'neuron'[11,12] has been developed for pattern recognition systems and as a basic element for adaptive logical circuits. This element would serve directly as an adaptive vote-taker, and such an application is discussed in detail below. A schematic of ADALINE is shown in Fig. 8. (Note the similarity to the decision element of Fig. 4.) It represents a flexible threshold-logic circuit having input lines, a single output line, and an input line, called the 'desired output', which is actuated during training only.

The binary input signals to ADALINE have values of +1 or -1, rather than the usual values of 1 or 0. Within the neuron, a linear combination of the input signals, each of which is multiplied by a certain weighting factor, is formed. The weights are the gains $a_1, a_2, \dots a_n$, which can have both positive and negative values. The output signal is +1 if the weighted sum is greater than a certain threshold, and -1 otherwise. The

The threshold level is determined by the setting of a_0, whose input is permanently connected to a +1 source. Varying a_0 varies a constant added to the linear combination of input signals.

For fixed gain settings, each of the 2^n possible input combinations could cause either a +1 or a -1 output. Thus, all possible inputs are classified into two categories. The input-output relationship is determined by choice of the gains a_0, $a_1, \ldots a_n$. In the adaptive neuron, these gains are set during the training procedure.

In general, there are 2^{2^n} different input-output relationships, or truth functions, by which the n binary input variables can be mapped into a single binary output variable. Only a subset of these relationships, *the linearly separated truth functions,*[13] can be realized by a single neuron of the form shown in Fig. 8.* Although this realizable subset is not all-inclusive, it is a very useful subset, and it is 'searchable', in that optimum gain settings for a given truth function can usually be found by a convergent iterative process.

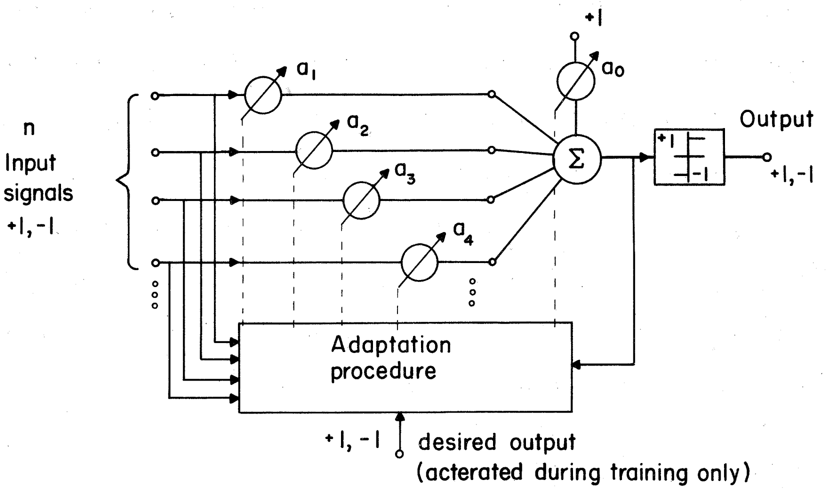


FIG. 8. BLOCK DIAGRAM OF THE ADAPTIVE ADALINE NEURON.

*As an example of a truth function which cannot be realized, no combination of gains a_0, a_1, and a_2 in a two-input neuron could give a +1 output with inputs -1, -1 and +1, +1 while giving a -1 output with inputs +1, -1 and -1, +1. Indeed, as n becomes large, the fraction of all possible truth functions which a single neuron can realize becomes exceedingly small.

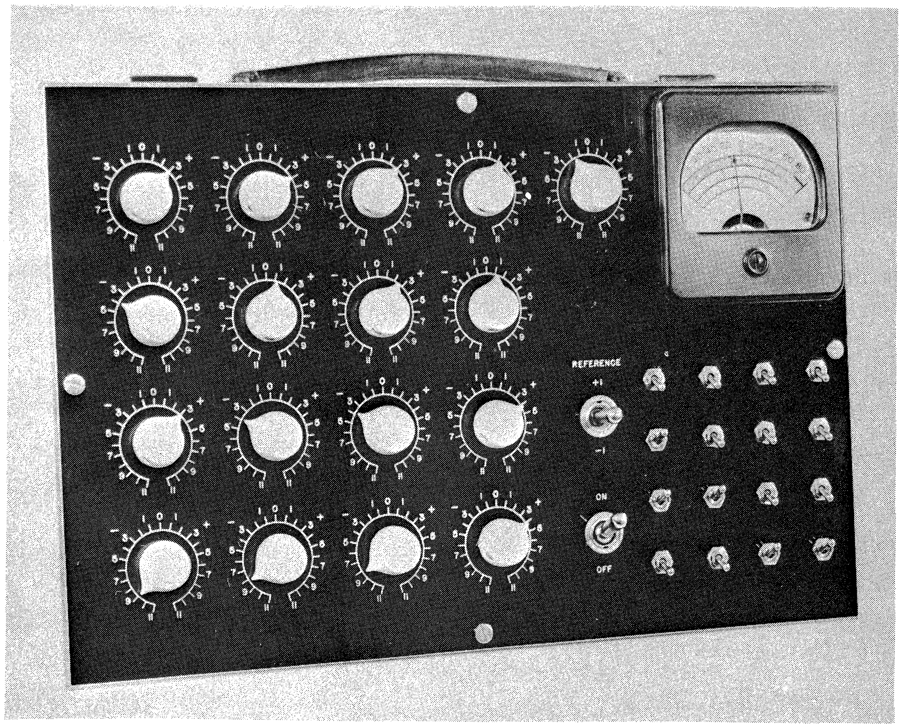


FIG. 9. A MANUALLY-ADAPTED ADALINE NEURON.

Application of this neuron in adaptive pattern classifiers was first made by Mattson.[14,15] He has shown that complete generality in choice of switching function could be achieved by combining these neurons. He devised an iterative digital computer routine for finding the best set of a's for the classification of noisy geometric patterns. An iterative procedure having similar objectives has been devised by B. Widrow and M. E. Hoff and is described next. This procedure is simple to implement, and can be analyzed by statistical methods that have been developed for the analysis of adaptive sampled-data systems.[16]

A. AN ADAPTIVE PATTERN CLASSIFIER

An adaptive pattern-classification machine has been constructed for the purpose of studying and illustrating adaptive behavior and artificial learning. It represents a single manually-adapted ADALINE neuron. A photograph of this machine is shown in Fig. 9.

During training, crude geometric patterns are fed to the machine by setting the toggle switches in the 4 x 4 input switch array. Setting another toggle switch (the reference switch) tells the machine whether the desired output for the particular input pattern is +1 or -1. The system learns something from each pattern and accordingly experiences a design change. The machine's total 'experience' is stored in the values of the weights $a_0 \ldots a_{16}$. The machine can be trained on undistorted noise-free patterns by repeating them over and over until the iterative search process converges, or it can be trained on a sequence of noisy patterns on a one-pass basis such that the iterative process converges statistically. Combinations of these methods can be accommodated simultaneously. After training, the machine can be used to classify the original patterns, and noisy or distorted versions of these patterns.

Details of the iterative searching routine used to train the manually adapted ADALINE are given in Appendix II. The iterative routine described is purely mechanical, and requires only adherence to a fixed set of rules. Electronic automation of this procedure, to get the completely self-adaptive ADALINE neuron of Fig. 8, will be discussed below.

B. STATISTICAL THEORY FOR THE ADAPTIVE NEURON ELEMENT

The statistical theory which led to the highly successful iterative searching routine, described in Appendix II, used to train ADALINE is derived in detail in references 11 and 12. Appendix III summarizes the results of this theory, which shows that the training procedure described in Appendix II converges toward thosegain settings, $a_0 \ldots a_n$, which minimize the mean of the square of the neuron errors, ϵ_n of Fig. 10, for all the patterns on which the neuron has been trained.

It is also possible to predict how much training a neuron needs before it will have reached its optimum state for handling a given set of input patterns. One can even show, statistically, how much worse than optimum the neuron is after any number of training experiences. To this end, it

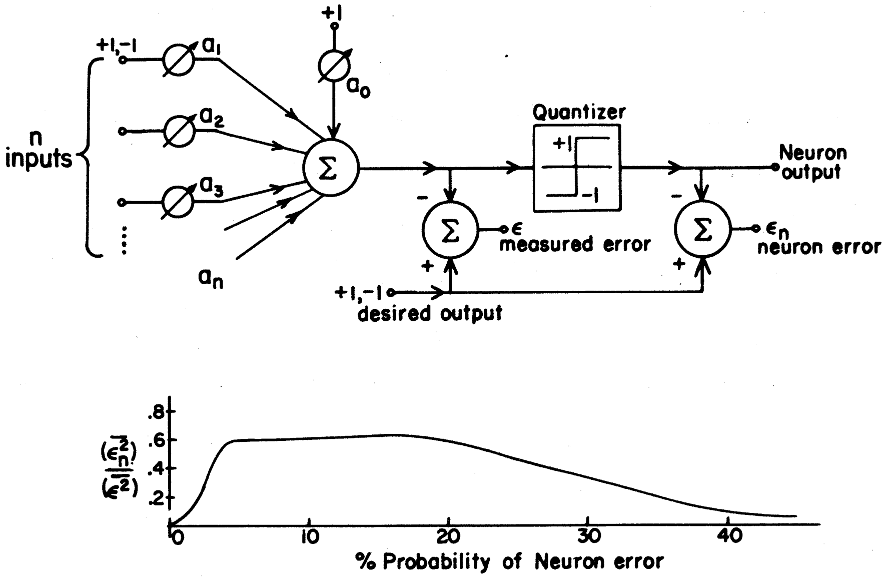
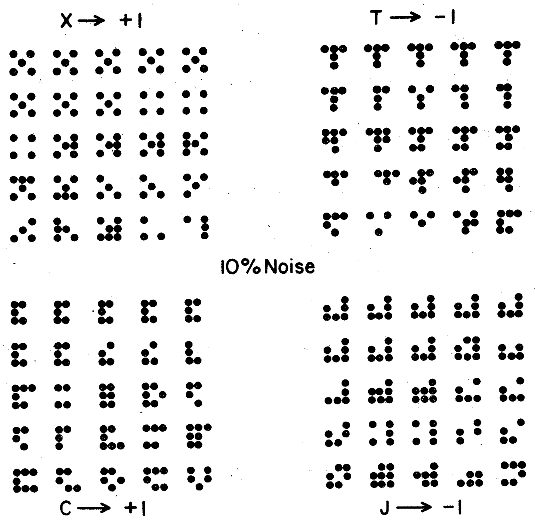


FIG. 10. RELATIONS BETWEEN ACTUAL NEURON ERRORS AND MEASURED ERRORS.

is useful to define a dimensionless parameter M, the 'misadjustment', as the ratio of the excess error probability to the minimum error probability. M = 0 implies a perfectly adjusted neuron, and M = 1 implies a neuron that makes twice as many errors as the optimum neuron. M is a measure of how an adaptive system performs, on the average, after adapting transients have died out, compared to a fixed system, whose design is optimized, based on perfect statistical knowledge. Misadjustment formulas developed for adaptive sampled-data systems[16] may be applied to the adaptive neuron.

Simulation tests have shown that the misadjustment formulas are highly accurate over a very wide range of pattern and noise characteristics. A description of a typical experiment and its results is given in Fig. 11. Noisy 3 x 3 patterns were generated by randomly injecting errors in ten per cent of the positions of the four 'pure' patterns, X, T, C, J. The best system, arrived at by slow precise adaptation on the full body of 100 noisy patterns, was able to classify these patterns as desired, except for twelve errors. The gains were then set to zero and



Best neuron makes 12 errors out of 100

EXPERIMENT #	PATTERNS ADAPTED ON	NUMBER OF ERRORS	MISADJUSTMENT
1	95, 79, 07, 60 73, 61, 08, 02, 72, 26	25	$M = \frac{25-12}{12} = 108\%$
2	70, 69, 52, 55, 32 97, 30, 38, 87, 01	19	$M = \frac{19-12}{12} = 58\%$
3	65, 12, 84, 83, 34 38, 71, 66, 13, 80	20	$M = \frac{20-12}{12} = 67\%$
4	07, 42, 85, 88, 63 35, 37, 92, 79, 22	28	$M = \frac{28-12}{12} = 133\%$

FIG. 11. EXPERIMENTAL ADAPTATION ON NOISY 3 x 3 BINARY PATTERNS.

ten patterns were chosen at random. The best system for these patterns was arrived at and tested on the full body of 100 patterns. Twenty-five classification errors out of 100 were made. The misadjustment was 108%. The experiment was repeated three more times, and the misadjustments that resulted, in order, were 58%, 67% and 133%. Since N = 10 patterns and n = 9 input lines, the expected misadjustment was, using the following formula for the theoretical misadjustment,

$$M = \frac{m + 1}{N} = 100 \text{ per cent}$$

An average taken over the four experiments gives a measured misadjustment of 91.5%, a close agreement.

The adaptive neuron can thus adapt to the job of pattern classification after seeing a very few patterns. A misadjustment of 20% is reasonable in many applications. To achieve this, all one has to do is supply the adaptive classifier with about five times as many patterns as there are input lines, regardless of how noisy the patterns are and how difficult the 'pure' patterns are to separate. Although the misadjustment formulas have been derived for the specific classifier consisting of a single adaptive neuron, it is suspected that the following 'rule of thumb' will apply well to a variety of adaptive classifiers: *the number of patterns required to train an adaptive classifier is equal to several times the number of bits per pattern.*

C. NETWORKS OF ADAPTIVE ADALINE NEURONS

Pure patterns and noisy versions of them that are linearly separable are readily classified by the single neuron. Nonlinearly separable pure patterns and their noisy equivalents can also be separated (as in the experiment of Fig. 11) by a single neuron, but absolute performance can be improved and the generality of the classification scheme can be greatly increased by using more than one neuron.

Two ADALINES were combined by using the following adaptation procedure: if the desired output for a given input pattern applied to both machines was -1, then both machines were adapted in the usual manner to ensure this; if the desired output was +1, the machine with the smallest measured error ϵ was assigned to adapt to give a +1 output while the other machine remained unchanged. If either or both machines gave outputs of +1, the pattern was classified as +1. If both machines gave -1 outputs, the pattern was classified as -1.

This procedure assigns specific 'responsibility' to the neuron that can most easily assume it. If, at the beginning of adaptation, a given neuron takes responsibility for producing a +1 with a certain input, pattern, it will invariably take this responsibility each time the pattern is applied during training. Notice that it is not necessary for the teacher to assign responsibility. This is done by a purely mechanical 'job assigner'. The combination does this automatically and requires only input patterns and the associated desired outputs, like the single neuron.

Various classification problems could be solved simultaneously by multiplexing neurons or combinations of neurons. One neuron might be trained to decide whether the man in a given picture does or does not have a green tie, while another neuron or combination could be trained to decide whether or not the man has a checkered shirt. Each neuron or combination has its own output line, and each is fed the appropriate desired output signal during training. The input signals are common to all neurons. In this manner, it is possible to form adaptive classifiers that can separate, with great accuracy, large quantities of complicated patterns into many output categories. Each neuron becomes a 'specialist' in classifying certain types of patterns.

D. ADALINE AS AN ADAPTIVE VOTE-TAKER

Vote-taking is actually a form of pattern recognition. The array of output signals arising at each calculation cycle from a set of voters comprises a spatial pattern which the vote-taker must classify (which the adaptive vote-taker must *learn* to classify) and deliver an output decision. The ADALINE neuron, utilizing the above-described adaptation procedure, has been applied directly to the job of adaptive vote-taker. Its performance closely approximates the ideal (whose structure is based on sureness information measurements), and is simple to implement physically. The training of the adaptive vote-taker is a continuous process. The 'correct' decision is injected at the 'desired output' point (Fig. 8). The changes in weight values per computation cycle are made to be exceedingly small. In a practical situation, the time constant of the adaptation process would be of the order of magnitude of the average interval between component failures.

The 'correct' decision signal could be supplied externally to permit adapting on check programs. An alternative method would derive this signal from the output decision of vote-taker itself. In Fig. 8, the 'desired output' point would be connected to the neuron output in a 'bootstrap' feedback arrangement. This alternative is the more attractive, since it does not require external signals to be supplied to vote-takers dispersed throughout a system, and since adaptation is possible during normal productive system operation. The bootstrap arrangement introduces a stability problem, however. Long chains of random errors could cause the vote-taker to so adapt as to consistently produce incorrect results. This can be prevented by setting the vote weights initially to produce correct results, and by making the adaptation process a very slow one. In system design, the chief problem is to choose a time constant of adaptation long enough to prevent instability and, at the same time, short enough to weed out components as they become defective.

E. REALIZATION OF AUTOMATIC ADAPTIVE NEURONS BY CHEMICAL 'MEMISTORS'

The structure and the adaptation procedure of the ADALINE neuron are sufficiently simple that an electronic fully-automatic neuron is being developed. The objective is a self-contained device, like the one sketched in Fig. 8, that has many signal input lines, a 'desired output' input line (which is actuated during training only), an output line, and a power supply. The device itself should be suitable for mass production, should contain few parts, and should be reliable.

To have such an adaptive neuron, it is necessary to be able to store the gain values, analog quantities which could be positive or negative, in such a manner that these values can be changed electronically.

A new circuit element called the memistor (a resistor with memory) has been devised by B. Widrow and M. E. Hoff for the realization of automatically adapted ADALINE neurons.[17] A memistor provides a single variable gain. Each neuron therefore employs a number of memistors equal to the number of input lines plus one.

A memistor consists of a conductive substrate with insulated connecting leads, and a metallic anode, all in an electrolytic plating bath. The conductance of the element is reversibly controlled by electroplating.

Like the transistor, the memistor is a 3-terminal element. The conductance between two of the terminals is controlled by the *time integral* of the current in the third, rather than by its instantaneous value as in the transistor. Reproducible elements have been made which are continuously variable (thousands of possible analog storage levels), and which typically vary in resistance from 100 ohms to 1 ohm, and cover this range in about 10 seconds with several milliamperes of plating current. Adaptation is accomplished by direct current, while sensing the neuron logical structure is accomplished nondestructively by passing alternating currents through the array of memistor cells.

A circuit for a memistor ADALINE is shown in Fig. 12. Notice the schematic symbol for the 3-terminal memistor. This circuit presumes that the neuron input signals are applied by means of switches, and that the over-all direction and extent of adaptation are controlled manually. The direction in which each memistor should be adapted (plated or stripped) is determined by the algebraic product of the error signal multiplied by the particular input signal. This product, and hence the direction of adaptation, is effected by the joint action of the adaptation control switch and a gang of each pattern switch, as shown in Fig. 12.

In the circuit of Fig. 12, the effect of positive and negative gain values is obtained by balancing the memistor against a fixed resistor in a bridge arrangement. The sensing of the gain is done by applying an a-c voltage to the memistor, and another a-c voltage with a 180-degree phase difference to the fixed resistor. The currents are proportional to the conductances and are summed. An individual gain is zero when the memistor conductance equals that of its reference, and an ideal value of reference conductance is the average of the conductance extremes of the memistor. None of the element values or memistor characteristics are critical, because of the inherent feedback in the adaptation process. These neurons have been built and have adapted (with somewhat reduced efficiency) even with 30 per cent of their memistors improperly manufactured and defective.

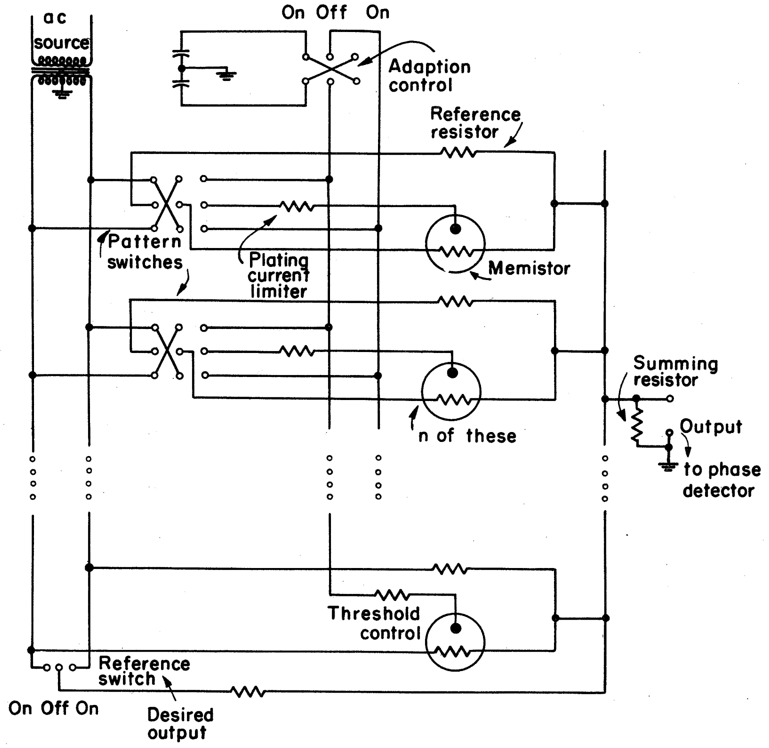


FIG. 12. CIRCUIT OF A MEMISTOR ADALINE.

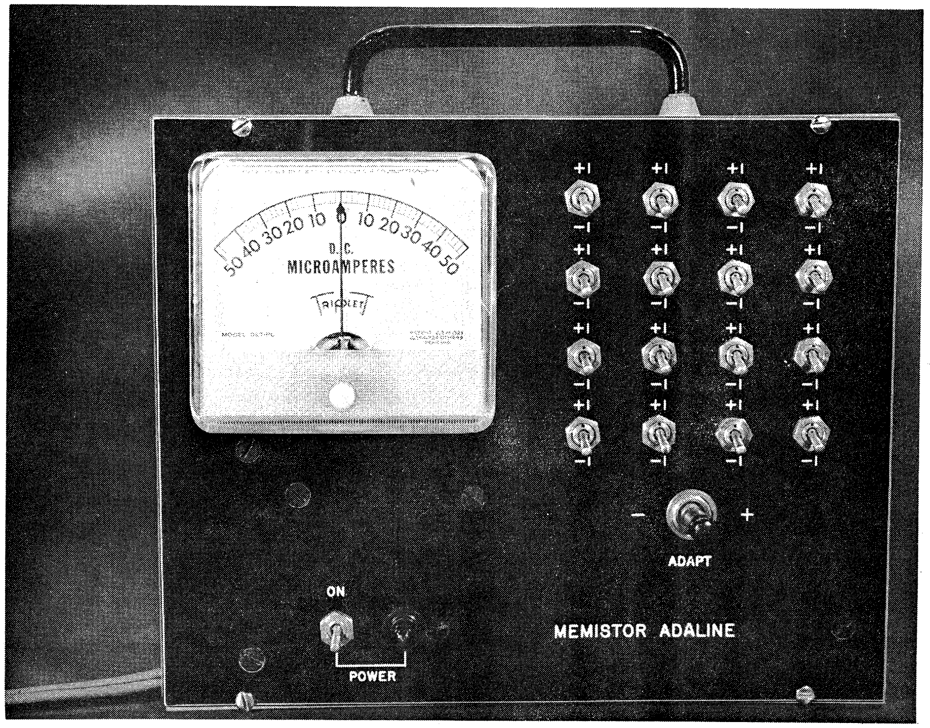


FIG. 13. AN ADAPTIVE MEMISTOR NEURON.

The first working memistors were made of ordinary pencil leads immersed in test tubes containing copper sulphate-sulphuric acid plating baths. Present elements are made by grinding down small $\frac{1}{10}$ watt carbon resistors so that a flat graphite surface is obtained with the resistors' connections exposed. Light coats of rhodium provide smooth substrates for plating, and protect the copper lead connections. The connections are insulated, and the substrates are sealed, with their individual copper plating baths, in lucite cells. These elements are small and rugged, cheap, simple, and non-critical in manufacture. Improvements are being sought (by using different baths and different plating metals, different geometries and different substrate materials) in lifetime, and in electrical characteristics such as stability, relaxation, smoothness, and speed of plating.

The first successful neuron using the lucite cells is pictured in Fig. 13. Patterns are fed to it in the usual manner, and it is trained to deliver the desired response to each pattern by holding the adapt control in the direction of desired needle motion, until the needle reads the desired response, then released. This 4 x 4 ADALINE has no knobs on its front panel, being equipped instead with 17-dimensional 'power steering'.

In addition to the electrochemical memistors described above, magnetic elements have shown promise for the creation of variable gain with memory. Analog storage in saturated magnetic cores has already been demonstrated.[18] A variable small-signal transformer of the form shown in Fig. 14 also shows promise; in this structure the coupling between the perpendicular input and output windings is controlled by the difference in small-signal permeabilities of legs A and B.

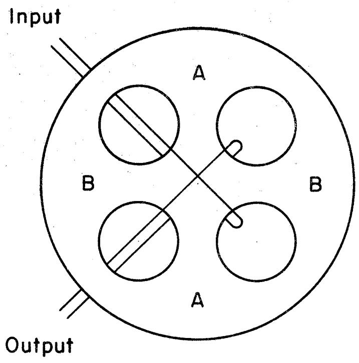


FIG. 14. AN ELECTRONICALLY VARIABLE TRANSFORMER.

IV. CONCLUSION

The application of the technology of microelectronics will be enhanced greatly by the use of redundancy and adaptation in prescribing systems which can adapt around their own internal flaws and which can be trained to their intended function. Study of various phenomena and device configurations which might provide the needed variable gain with memory should accompany the advances now being made in microelectronic techniques for fabricating active and passive components and subassemblies.

APPENDIX I: ERROR PROBABILITIES OF ADAPTIVE DECISION ELEMENTS

The probability that the decision element of Fig. 4 will make an error is based on the probabilities that the output of the summing element will be negative when the correct answer is +1, and positive when the correct answer is -1. Let v be the output of the summing element times the correct answer, so that v > 0 implies a correct output and v < 0 implies an incorrect one. If errors in the inputs are independent, v is just the sum of ρ random variables, namely the sum of each input times the correct answer times the vote weight. Therefore, the probability density of v is the convolution of the probability densities of the terms in the sum.

Let λ_D be the error probability of the output decision. λ_D may be found exactly from the probability density of v, or it may be approximated by the inequality

$$\lambda_D < \prod_{i=1}^{\rho} \left\{ 2\sqrt{\lambda_i(1-\lambda_i)} \cosh\left[\frac{a_i - \ln\left(\frac{1-\lambda_i}{\lambda_i}\right)}{2}\right]\right\} \tag{A-1}$$

where Π denotes the product of ρ terms of the form shown.

The closeness of the bound can be evaluated for the majority-rule decision element with equally reliable inputs. When the formula for majority-rule [Eq. (2) of text] is evaluated using Stirling's formula, the ratio of the bound on λ_D given by (A-1) to the actual value of λ_D is approximately $\sqrt{(\pi\rho)/2}$. Because the bound on λ_D goes geometrically in ρ, the bound is quite close. (Example: Find ρ when each input to a majority-rule decision element makes one error in 50, and λ_D must be less than 10^{-14}. The exact formula gives $\rho = 23$; the bound gives $\rho = 26$.)

The bound on λ_D clearly demonstrates the advantage of adaptation, for the cosh term has its minimum, for each i, when

$$a_i = \ln \frac{1-\lambda_i}{\lambda_i} \tag{A-2}$$

This is just the optimum vote weight discussed in the text. If the optimum a_i are used, adding an input with error probability, λ_j multiplies the bound on λ_D by $2\sqrt{\lambda_j(1-\lambda_j)}$. Thus, for small λ_i and good adaptation, the bound on λ_D goes roughly as 2^ρ times the product of the square roots of the λ_i. If an input has an error probability of one half, the bound on λ_D will be increased unless that input is given a vote weight of zero. If poor or no adaptation is used, then there must be an increase in the redundancy for given λ_D over that with perfect adaptation.

The importance of redundancy in a large digital network can be demonstrated by a simple combinational analysis. Assume a redundant digital network using restoring organs in the manner shown in Fig. 2. Let

N = total number of different stages in the network,

ρ = the redundancy (number of inputs to each decision element),

m = number of inputs to a decision element which must perform correctly in order for the decision element to perform correctly,

λ = the probability that the output of one logical stage performs correctly.

Note that the value of m depends upon the adaptation. Without adaptation (majority-rule) m is greater than $\rho/2$. With perfect adaptation, m could conceivably be as low as 1.

The probability that one of the stages will not have at least m correctly performing inputs is

$$\sum_{h=0}^{m-1} \binom{\rho}{h} (1-\lambda)^h \lambda^{\rho-h}$$

Therefore, the probability that the system with N stages performs correctly is

$$\text{p(system performs correctly)} = \left[1 - \sum_{h=0}^{m-1} \binom{\rho}{h} (1-\lambda)^h \lambda^{\rho-h}\right]^N \tag{A-3}$$

The above formula was used to find the initial yields for the examples in the text. It was also used to find the survival probability plotted in Figs. 5 and 6, by setting $\lambda = e^{-t/\tau}$ for $\tau = 1$. The median lifetime, T, is found by equating the left side of the equation to 0.5 and λ in the right side to $e^{-T/\tau}$. When $m = 1$, the median lifetime can be found explicitly:

$$T = -\tau \ln 1 - \left[(1 - 2^{-1/N})^{1/\rho}\right] \qquad \text{(A-4)}$$

For $N = 100$, $\rho = 1$ gives $T = 0.0069\ \tau$, $\rho = 3$ gives $T = 0.21\ \tau$, and $\rho = 9$ gives $T = 0.86\ \tau$. Thus, a redundancy of 3, with adaptation, can extend the median lifetime by a factor of 30.

APPENDIX II: TRAINING THE MANUAL ADALINE NEURON

This appendix is a description of the iterative searching routine used to train the manually adapted ADALINE shown in Fig. 9. A pattern is fed to the machine, and the reference switch is set to correspond to the desired output. The error is then read (by switching the reference switch, the error voltage appears on the meter, rather than the neuron output voltage). All gains including the level are to be changed by the same absolute magnitude, so that the error is brought to zero. This is accomplished by changing each gain (which could be positive or negative) in the direction which will diminish the error magnitude by 1/17. The 17 gains may be changed in any sequence, and after all changes are made, the error for the present input pattern is zero. Returning the reference to the neutral position, the meter reads exactly the desired output. The next pattern, and its desired output, are presented and the error is read. The same adjustment routine is followed and the error is brought to zero. If the first pattern were reapplied at this point, the error would be small but not necessarily zero. More patterns are inserted in like manner. Convergence is indicated by small errors (before adaptation), with small fluctuations about a stable root-mean-square value.

This adaptation procedure may be readily modified to get slower (and smoother) adaptation by correcting only a fraction of the error with the insertion of each pattern.

APPENDIX III: STATISTICAL THEORY FOR ADAPTIVE NEURONS

The error signal measured and used in adaptation of the neuron of Fig. 9 is the difference between the desired output and the weighted sum before quantization. This error is indicated by ϵ in Fig. 10. The actual neuron error, indicated by ϵ_n in Fig. 10, is the difference between the neuron output and the desired output.

The objective of adaptation could be stated in the following manner. Given a collection of input patterns and the associated desired outputs, find the best set of weights $A_0, A_1, \ldots a_n$ to minimize the mean square of the neuron error, $\overline{\epsilon_n^2}$. Individual neuron errors could only have the values of +2, 0, *and* -2 with a two-level quantizer. Minimization of $\overline{\epsilon_n^2}$ is therefore equivalent to minimizing the average number of neuron errors.

The simple adaptation procedure described in this paper minimizes $\overline{\epsilon^2}$, rather than $\overline{\epsilon_n^2}$. The measured error ϵ will be assumed to be Gaussian-distributed with zero mean. Using certain geometric arguments, it can be shown that under these conditions, $\overline{\epsilon_n^2}$ is a monotonic function of $\overline{\epsilon^2}$ and that minimization of $\overline{\epsilon^2}$ is equivalent to the minimization of $\overline{\epsilon_n^2}$ and thus to the minimization of the probability of neuron error. The ratio of these mean squares has been calculated and is plotted in Fig. 10 as a function of the neuron error probability. This plot is a good approximation even when the error probability density differs considerably from the above assumptions.

Given any collection of input patterns and the associated desired outputs, the measured mean square error $\overline{\epsilon^2}$ can be shown to be a precisely parabolic function of the gain settings, $a_0, \ldots a_n$. Therefore, adjusting the a's to minimize $\overline{\epsilon^2}$ is equivalent to searching a parabolic stochastic surface (having as many dimensions as there are a's) for a minimum. How well this surface can be searched will be limited by a sample size, i.e., by the number of patterns 'seen' in the searching process.

The method of searching that has proven most useful is the method of steepest descent. Vector adjustment changes are made in the direction of the surface gradient. The procedure described for bringing each error to zero implements the method of steepest descent with each successive input pattern.

REFERENCE

1. M. M. Perugini, and Nilo Lindgren, 'Microminiaturization', Electronics, vol. 33, pp. 78-108; November 25, 1960.
2. I. A. Lesk, et al., 'A categorization of the solid-state device aspects of microsystems electronics', Proc. IRE, vol. 48, pp. 1833-1841; November, 1960.
3. J. T. Wallmark, 'Design considerations for integrated electronic devices', Proc. IRE, vol. 48, pp. 293-300; March, 1960.
4. K. R. Shoulders, 'On microelectronic components, interconnections, and system fabrication', 1960 Western Joint Computer Conference, May, 1960.
5. C. E. Shannon and E. F. Moore, 'Reliable circuits using less reliable relays', J. Franklin Inst., vol. 262, pp. 191-208 and 281-297; September and October, 1956.
6. J. von Neumann, 'Probabilistic logics and the synthesis of reliable organisms from unreliable components', Automata Studies, Princeton University Press; 1956.
7. W. H. Pierce, 'A proposed system of redundancy to improve the reliability of digital computers', Technical Report 1522-1, Stanford Electronics Laboratories, Stanford, California; July 29, 1960.
8. J. J. Suran ---- this issue.
9. D. E. Rosenheim, and R. B. Ash, 'Increasing reliability by the use of redundant machines', IRE Trans. (Electronic Computers), vol. EC-8, pp. 125-130; June, 1959.
10. B. J. Flehinger, 'Reliability improvement through redundancy at various system levels', IBM J. Res. and Dev., vol. 2, pp. 148-158; April 1958.
11. B. Widrow and M. E. Hoff, 'Adaptive switching circuits', 1960 WESCON Convention Record, part IV, pp. 96-104; August 23, 1960.
12. B. Widrow and M. E. Hoff, 'Adaptive switching circuits', Technical Report No. 1553-1, Stanford Electronics Laboratories, Stanford University, Stanford, California; June, 1960.
13. R. McNaughton, 'Unate truth functions', Technical Report No. 4, Applied Math. and Statistics Lab., Stanford University; October 21, 1957.
14. R. L. Mattson, 'The design and analysis of an adaptive system for statistical classification', S.M. Thesis, Electrical Engineering Dept., Mass. Inst. of Tech.; May 22, 1959.
15. R. L. Mattson, 'A self-organizing logical system', 1959 E.J.C.C. Convention Record, Inst. of Radio Engineers, 1959.
16. B. Widrow, 'Adaptive sampled-data-systems---a statistical theory of adaption', 1959 WESCON Convention Record, part 4.
17. B. Widrow, 'An adaptive ADALINE neuron using chemical memistors', Technical Report No. 1553-2, Stanford Electronics Laboratory, Stanford, California; October 17, 1960.
18. A. E. Brain, 'The simulation of neural elements by electronical networks based on multi-aperture magnetic cores', Proc. IRE, vol. 49, pp. 49-52; January, 1961.