machine, upon the exact nature of the training sequence preceding the problem, and upon the time available for solution.

Similar remarks apply to human beings.

REFERENCES

1. R. J. Solomonoff: "An Inductive Inference Machine". IRE Convention Record, Section on Information Theory, pp. 56-62, 1957.
2. R. J. Solomonoff: "A Preliminary Report on a General Theory of Inductive Inference," Zator Technical Bulletin No. 138. AFSOR TN-50-1459, Zator Co., November 1960.
3. A. Newell, H. Simon, J. Shaw: "Report on a General Problem Solving Program." *Information Processing,* Butterworth Scientific Publications, London, 1960.
4. T. Kilburn, R. Grimsdale, F. Sumner: "Experiments in Machine Learning and Thinking." *Information Processing,* Butterworth Scientific Publications, London, 1960.
5. R. J. Solomonoff: "An Inductive Inference Code Employing Definitions." Zator Technical Bulletin No. 141, AFOSR 2214, Zator Co., 1961.

GENERALIZATION AND INFORMATION STORAGE IN NETWORKS OF ADALINE "NEURONS"

Bernard Widrow
Stanford University, Stanford, California
and
President, Memistor Corporation
Mountain View, California

A. Introduction

Every adaptive or learning machine whose structure evolves with experience has memory associated with the adapted parts. This is an integrative memory, a type of memory that does not contain detailed facts, but does contain a distillation of the experiences of the adaptive system. The trained system stores a means of reacting to stimuli, but does not store the training stimuli themselves. When a stimulus is presented to such a system, the system reacts to it instantly, and does not cull through its memory trying to match the present stimulus to a previous stimulus and thereby react as it did in a previous situation. Adaptive systems can be trained to produce specific output signals in reaction to specific input signals, and at the same time, to react reasonably to stimuli that are related in some fashion to the training stimuli yet not necessarily a part of the training signal set. Information is stored in the adaptive memory by an iterative training process. The storage "capacity" of such systems is statistical, and information is stored in a diffuse manner, everything stored everywhere, not specific facts in specific registers. A basic building block for many simple forms of adaptive memory systems is an adaptive threshold logic element, which is described below.

B. ADALINE, An Adaptive Logic Element

A basic building block of the memory systems to be considered is an adaptive threshold element, sometimes called an adaptive "neuron." For the past several years, we at Stanford University have called this element ADALINE (adaptive linear neuron). A functional diagram of this element is shown in Figure 1. It includes an adjustable threshold function and the

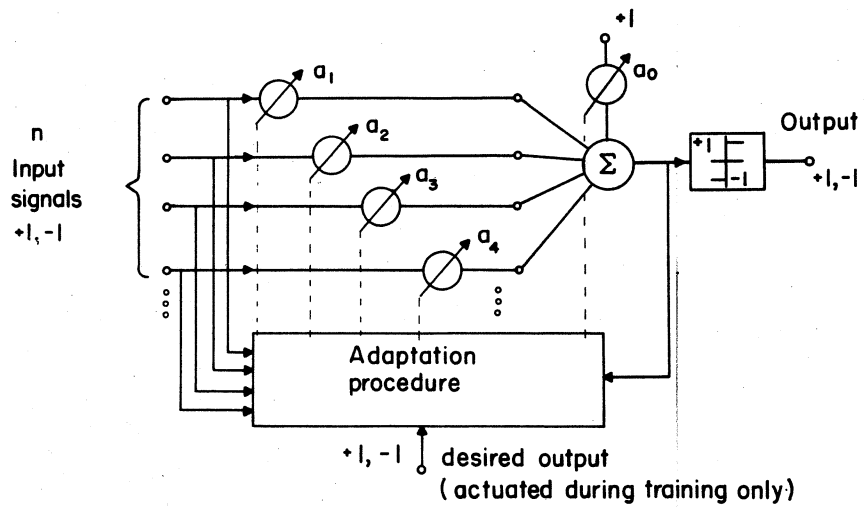


Figure 1. An Automatically-Adapted Threshold Element.

adaptation machinery which automatically adjusts the variable weights. It has been demonstrated experimentally and theoretically that this element can be trained to react specifically to a wide variety of binary input signals and that it can be trained to generalize in certain ways, i.e., to react as desired with high reliability to inputs that it has not been specifically trained on.

In Figure 1, the binary input signals on the input lines have values of +1 or -1, rather than the usual values of 1 or 0. Within the neuron shown, a linear combination of the input signals is formed. The weights are the gains a_1, a_2, ..., which could have both positive and negative values. The output signal is +1 if this weighted sum is greater than a certain threshold, and -1 otherwise. 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 would 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 $a_0, \ldots, a_n$. 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 input variables can be mapped into the single output variable. Only a subset of these, the linearly separable logic functions, can be realized by all possible choices of the gains. Although this subset is not all-inclusive, it is a useful subset, and it is "searchable," i.e., the

"best" function in many practical cases can be found iteratively without trying all functions within the subset. An iterative search procedure has been devised and is described below. This procedure is quite simple to implement, and can be analyzed by statistical methods that were originally developed for the analysis of adaptive sampled-data systems [1].

An adaptive pattern classification machine has been constructed for the purpose of illustrating adaptive behavior and artificial learning. A photograph of this machine, which is an adjustable threshold element (called "KNOBBY ADALINE"), is shown in Figure 2.

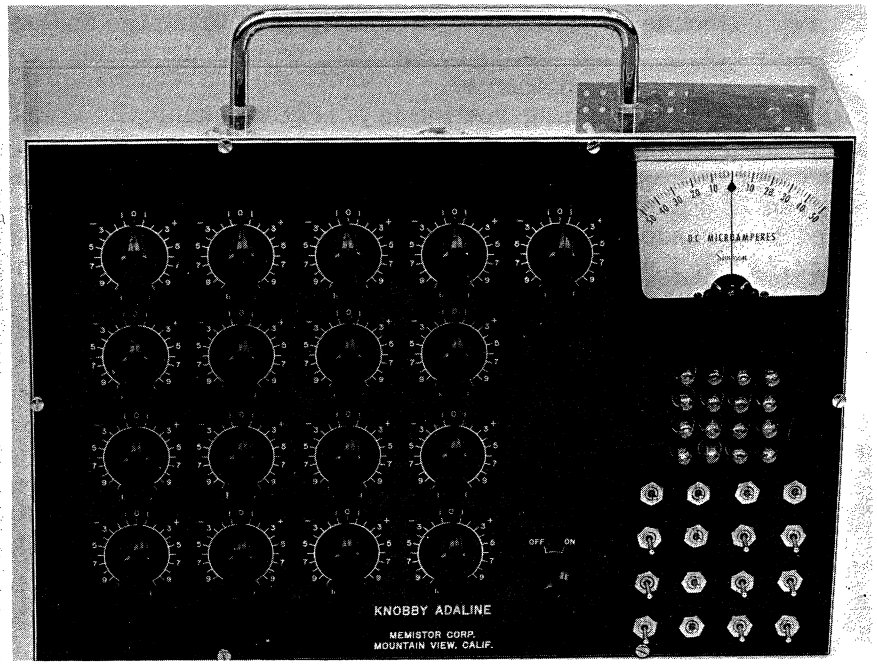


Figure 2. An Elementary Learning Machine.

During a training phase, simple geometric patterns are fed to the machine by setting the toggle switches in the 4 × 4 input switch array. All gains, including the threshold level, are to be changed by the same absolute magnitude such that the analog error (the difference between the desired meter reading and the actual meter reading) is brought to zero. This is accomplished by changing each gain in the direction which will diminish the error 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. The weights associated with switches up

(+1 input signals) are incremented by rotation in the same direction as the desired meter needle rotation, the weights connected to switches in the down position are incremented opposite to the desired direction of rotation of the meter needle. The next pattern and its desired output is then 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 adaption), with small fluctuations about stable weights. Note that adaption is indicated even if the quantized neuron output is correct. If, for example, the desired response is +1, the neuron is adapted to bring the analog response closer to the desired response, even if the analog response is more positive than +1.

The iterative training routine is purely mechanical. Electronic automation of this procedure will be discussed below.

The results of a typical adaption on six noiseless patterns is given in Figure 3. During adaption, the patterns were selected in a random sequence, and were classified into 3 categories. Each T was to be mapped to +30 on the meter dial, each G to 0, and each F to -30. As a measure of performance, after each adaptation, all six patterns were read in (without adaptation) and six errors were read. The sum of their squares denoted by $\Sigma\varepsilon^2$ was computed and plotted. Figure 3 shows the learning curve for the case in which all gains were initially zero.

The error signal measured and used in adaption of the neuron of Figure 1 is the difference between the desired output and the sum before quantization. This error is indicated by ε in Figure 4. The actual neuron error, indicated by ε_n in Figure 4, is the difference between the neuron output and the desired output.

The objective of adaption is the following. 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, $\overline{\varepsilon_n^2}$. Individual neuron errors could only have the values of +2, 0, *and* -2 with a two-level quantizer. Minimization of $\overline{\varepsilon_n^2}$ is therefore equivalent to minimizing the average number of neuron errors.

The simple adaption procedure described in this paper minimizes $\overline{\varepsilon^2}$ rather than $\overline{\varepsilon_n^2}$. The measured error ε has zero mean (a consequence of the minimization of $\overline{\varepsilon^2}$) and will be assumed to be Gaussian-distributed. It can be shown that $\overline{\varepsilon^2}$ is very close to being a monotonic function of $\overline{\varepsilon_n^2}$, and that minimization of $\overline{\varepsilon^2}$ is essentially equivalent to minimization of $\overline{\varepsilon_n^2}$ and to minimization of the probability of neuron error. The ratio of

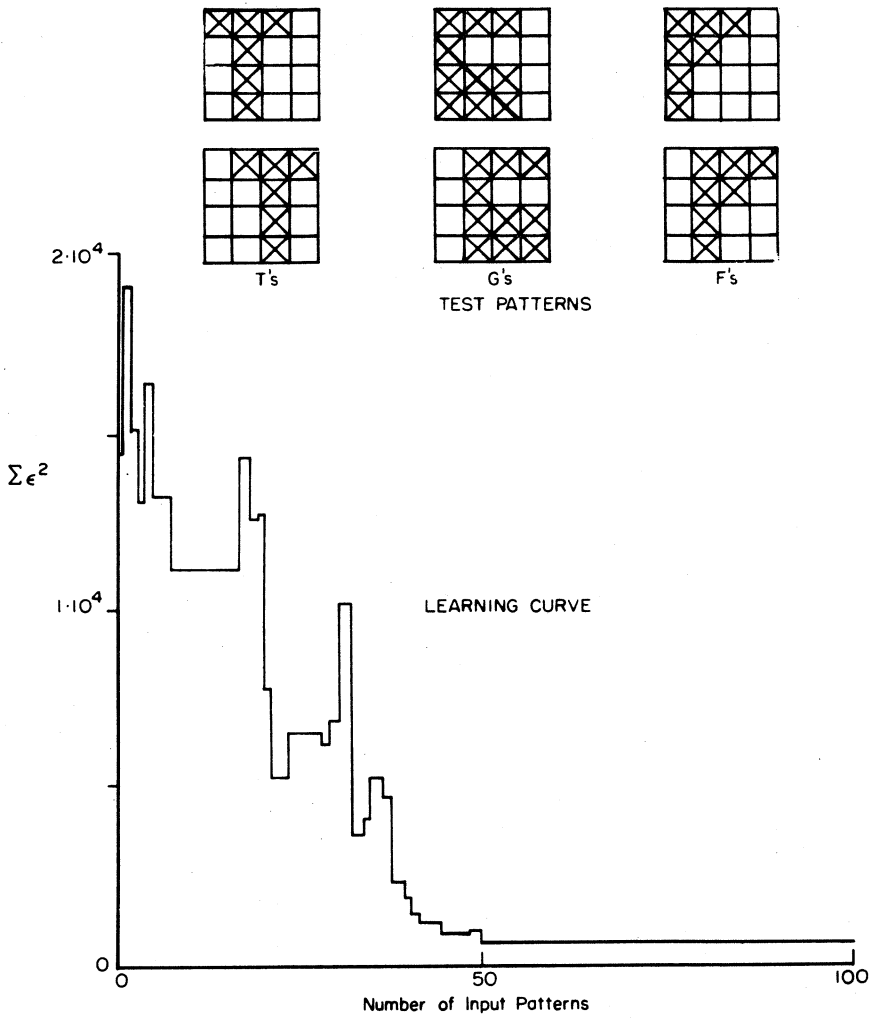


Figure 3. Measurement of Rate of Adaption.

these mean squares has been calculated and is plotted in Figure 4 as a function of the neuron error probability.

Given any collection of input patterns and the associated desired outputs, the measured mean-square error $\overline{\varepsilon^2}$ must be a precisely parabolic function of the gain settings, $a_0, \ldots, a_n$. Let the kth pattern be indicated as the vector $S(k) = s_1(k), s_2(k), \ldots, s_n(k)$. The s's have values of +1 or -1, and represent the n input components numbered in a fixed manner. The kth error is

$$\varepsilon(k) = d(k) - a_0 - a_1 s_1(k) - a_2 s_2(k) - \ldots - a_n s_n(k) \qquad (1)$$

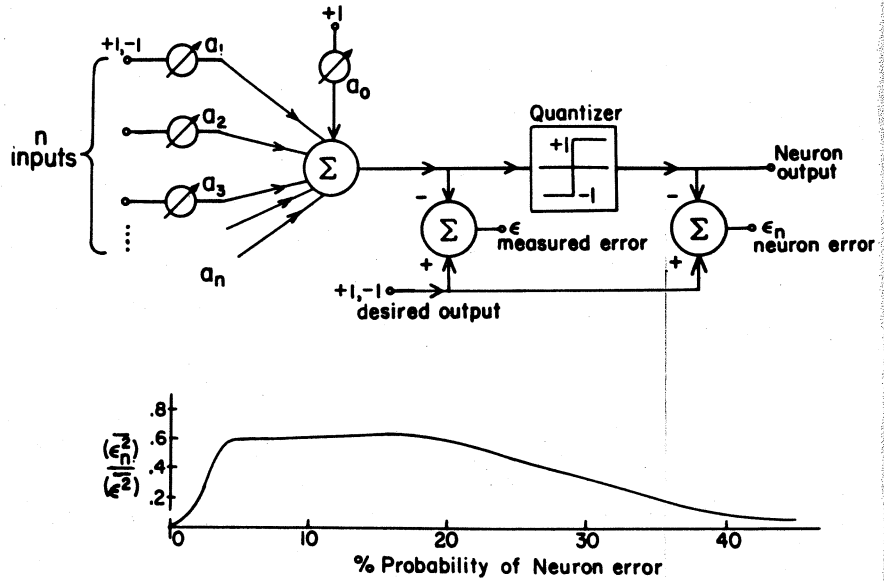


Figure 4. Relationship Between Error Rate and Mean Square Error.

For simplicity, let the neuron have only two input lines and a level control. The square of the error is accordingly

$$\varepsilon^2(k) = d^2(k) + a_0^2 + s_1^2(k)a_1^2 + s_2^2(k)a_2^2$$
$$- 2d(k)a_0 - 2d(k)s_1(k)a_1 - 2d(k)a_2 \qquad (2)$$
$$+ 2s_1(k)a_0a_1 + 2s_2(k)a_0a_2 + 2s_1(k)s_2(k)a_1a_2$$

The mean-square error averaged over k is

$$\overline{\varepsilon^2} = a_0^2 + \phi(s_1, s_1)a_1^2 + \phi(s_2, s_2)a_2^2 = \overline{da}_0$$
$$- 2\,\phi(d, s_1)a_1 - 2\,\phi(d, s_2)a_2 + 2\overline{s}_1 a_0a_1 + 2\overline{s}_2 a_0a_2 \qquad (3)$$
$$+ 2\,\phi(s_1, s_2)a_1a_2 + \phi(d, d)$$

The ϕ's are spatial correlations. $\phi(s_1, s_2) = \overline{s_1\, s_2}$, etc. Note that $\phi(s_j, s_j) = \overline{s_j\, s_j} = 1$.

Adjusting the a's to minimize $\overline{\varepsilon^2}$ 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 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 weight changes are made in the direction of the gradient. Transients consist of sums of geometric sequence components (there are as many natural "frequencies" as the number of weights). It can be shown that the method of steepest descent will be stable when the proportionality constant k between partial derivative and size of weight change is less than the reciprocal of the second partial derivative. It can also be shown that when k is small, transients can be approximately represented as being of the single time constant 1/2k.

The method of adaption that has been used requires an extremely small sample size per iteration cycle, namely one pattern. One-pattern-at-a-time adaption has the advantages that derivatives are very easy to measure and that no storage is required within the adaptive machinery except for the gain values (which contain the past experience of the neuron).

The square of the error for a single pattern (the mean-square error for a sample size of one) is given by Eq. (2). The partial derivatives are

$$\frac{\partial \varepsilon^2(k)}{\partial a_0} = [-2d(k) + 2a_0 + 2s_1(k)a_1 + 2s_2(k)a_2]$$
$$\frac{\partial \varepsilon^2(k)}{\partial a_1} = s_1(k)\,[-2d(k) + 2a_0 + 2s_1(k)a_1 + 2s_2(k)a_2] \qquad (4)$$
$$\frac{\partial \varepsilon^2(k)}{\partial a_2} = s_2(k)\,[-2d(k) + 2a_0 + 2s_1(k)a_1 + 2s_2(k)a_2]$$

Comparison of Eqs. (4) with Eq. (1) shows that the derivatives are simply related to the measured error, and suggest that the *derivatives could be measured without squaring and averaging and without actual differentiation.* The jth partial derivative is given by

$$\frac{\partial \varepsilon^2(k)}{\partial a_j} = -2s_j(k)\,\varepsilon(k) \qquad (5)$$

It follows that all partial derivatives have the same magnitude, and have signs determined by the error sign and the respective input signal signs. The adaption procedure described above for bringing $\varepsilon(k)$ to zero with each successive input pattern gives the constant k a value of $1/2(n+1)$. From the previous discussion we see that the time constant of the iterative process is therefore $\tau = (n+1)$ patterns. On the 4×4 ADALINE, there are $n = 16$ input line gains plus a level control. Therefore, the time constant

should be equal to the number of weights, i.e., 17 patterns (for verification, see the learning curve of Figure 3). The search procedure could be readily modified to speed up or slow down the adaption process by adjusting the fraction of the error corrected with each cycle of adaption.

The minimum mean-square error adaption process described above is a stable "performance feedback" process. Feedback is used to control system structure. A number of other adaption procedures have been investigated by W. C. Ridgway, III in his doctoral thesis [2]. For instance, adaption might be done only when the neuron digital output is incorrect, and then adaption would take place until the confidence level equalled some fixed quantity. He has shown that such a procedure is guaranteed to converge and will separate all possible sets of linearly separable patterns. The weights will stabilize automatically. This procedure has the advantage over the minimum mean-square error procedure in that the latter procedure will not be able to separate a small fraction of the patterns in some cases of large, complex, but linearly separable pattern sets. The minimum mean-square error procedure is the most useful in training generalizations into ADALINES.

An important question is, how many patterns or stimuli can the single adaptive neuron be trained to react to correctly at a time. This is a statistical question. If within a large group of patterns those which are to give the + response are similar to each other and dissimilar to those to give the - response, the neuron has little trouble adapting to making the desired distinctions. On the other hand, if two patterns that differ by one bit are to give opposite responses, the critical weight must have a large value and be of appropriate sign. If two other similar patterns are to be inserted to give opposite responses, and the same weight is the critical one but here the necessary sign requirement for this weight is opposite to the previous requirement, clearly the set of only 4 patterns will not be linearly separable. A series of experiments was devised by J. S. Koford where patterns containing unbiased random bits and random desired responses were applied to ADALINES with varying numbers of inputs. *It was found that the average number of random patterns that can be absorbed by an ADALINE is equal to twice the number of weights.* This is one basic measure of memory capacity.

C. Comparison of a Neuron Memory With a Magnetic Core Storage

A large magnetic core storage installed in the TX-2 computer at the M.I.T. Lincoln Laboratory contains 36 memory

planes. Each plane is a 256 × 256 array of cores. The entire system is capable of storing close to 65,000 36-bit binary words, about two and a half million bits of capacity. It is instructive to compare this memory system with a bank of 36 ADALINES, whose 4 × 4 inputs are connected in parallel, as illustrated in Figure 5.

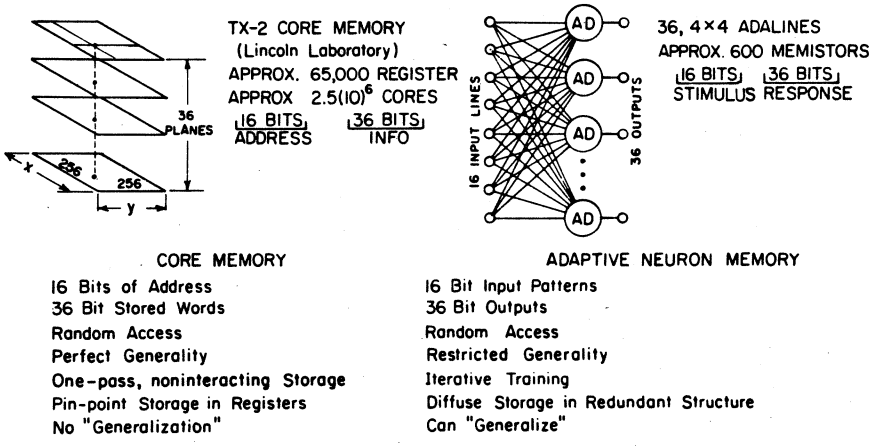


Figure 5. Comparison of a Magnetic-Core Memory With an Adaptive Neuron Memory.

There are two phases to the operation of the core and the neuron memory systems. For the core storage, there is the recording phase, where a 36-bit word is inserted into a given register, and can be reproduced during the second phase of operation, the sensing or reading phase, such that the 36-bit stored word is like a response to the 16-bit address stimulus. For the neuron memory, there is the training phase, where 36-bit responses are trained-into the bank of ADALINES. During the sensing phase, 16-bit patterns are presented, and responses to them are formed almost simultaneously. The address is analogous to pattern, and the information stored at a given address in the core stored in analogous to the response to a pattern in the neuron memory. Both memories have the random-access feature.

Recording information in the core storage does not disturb previously recorded information in other addresses. Recording information in the neurons does disturb previously recorded data, and it is in iterative process which must be repeated until all desired training responses are "socked in," if they are separable.

The neuron memory is highly restricted in generality. Each neuron can only be trained to classify linearly separable

patterns. The core memory has perfect generality, in that all possible combinations of the 36 output response bits can be inserted into each of the 2^{16} storage registers. The core memory has no ability to generalize, however. If the contents of a selected register are to contain relevant information, it is necessary that the specific information be inserted into that register. It matters not that the pattern of bits representing this register might be related in some way to those of other registers in which relevant information has been stored. The neuron memory on the other hand can be trained to respond specifically to certain inputs, and to react similarly to other inputs that are related in some generalized fashion to the training patterns. Certain kinds of generalizations that can be trained into the single ADALINE will be described below. Structures containing more than one ADALINE that have greater storage capacity will also be described. It is a general result that the greater the storage capacity of a given structure, the greater the generality and, on the other hand, the greater is the amount of training required to produce generalization learning.

D. MADALINE, A Parallel Network of ADALINES

Storage capacity in excess of that of a single ADALINE can be readily achieved by use of parallel multi-neuron networks. Several neurons can be used to assist each other in solving problems by automatic load-sharing.

The configuration in Figure 6 shows 5 ADALINES having parallel-connected inputs. Their 5 outputs are connected to a majority-rule element whose output is the system output. One procedure for training this network is the following. A pattern is inserted, and if the response of the majority element is the desired response, no adaption takes place. If, on the other hand, the desired response is +1, and 3 of the 5 ADALINES read -1 for the given input pattern, one of the latter three must be adapted to the +1 state. The one that is adapted is the one whose confidence level (the sum before quantization) is closest to zero, i.e., the one whose analog response is closest to the desired response. If more of the ADALINES were originally in the -1 state, enough of them would be adapted to the +1 state to make the majority decision be +1. The ones adapted would have had confidence levels closest to zero. This adaption procedure is symmetric with respect to adaption when the desired response is -1. Differences in initial conditions and the results of subsequent adaption cause the various neurons to take "responsibility" for certain parts of the training problem. The basic principle of load-sharing is summarized as follows: *assign responsibility to the neuron or neurons that can most easily assume it.*

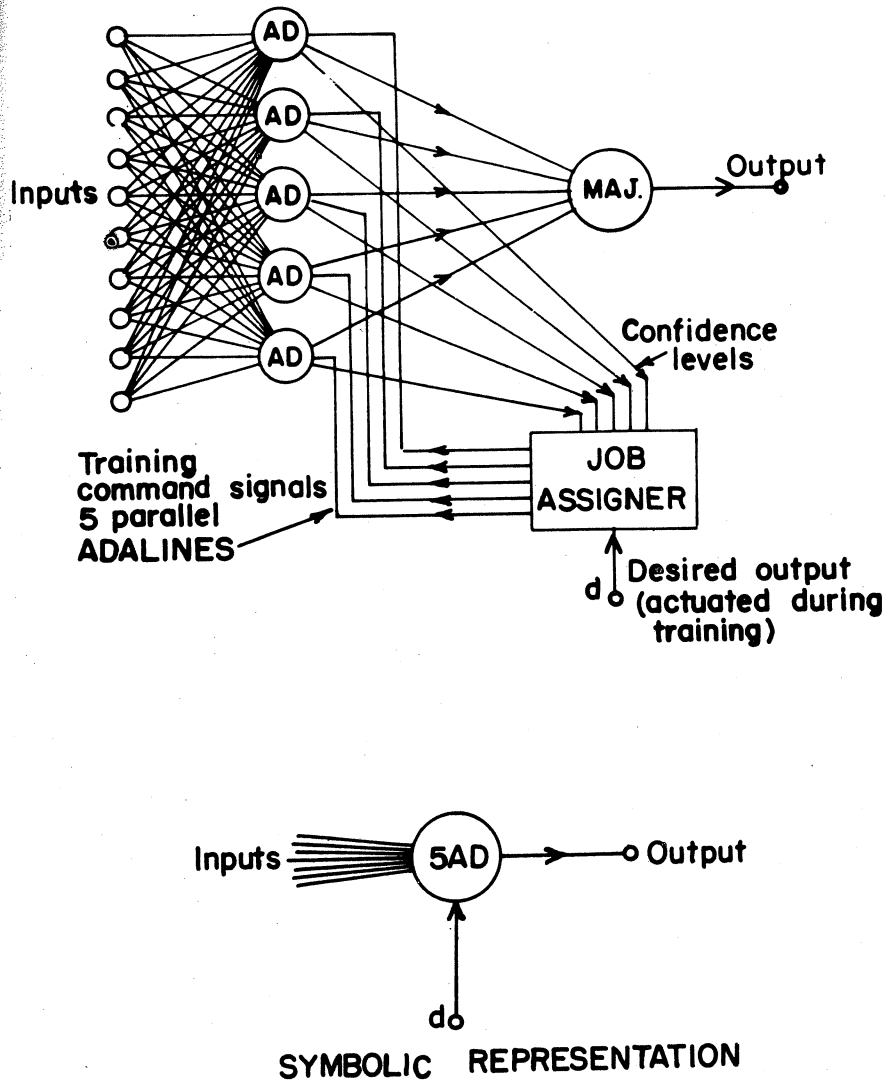


Figure 6. Configuration of MADALINE.

In Figure 6, the "job assigner," a purely mechanical process, assigns responsibility during the training process by transferring the desired-response adapt commands to the selected ADALINES. The job assigner utilizes confidence level information.

The adaptive system of Figure 6 was suggested by common sense, was tested by simulation, and was found to work very elegantly. It was subsequently proven by Ridgway in his doctoral thesis that if a set of weights exists that will solve the

training problem, then this system will converge on a solution. The essence of the proof lies in showing that the probability of a given neuron taking responsibility for adaption to a given input stimulus-desired response is greatest if that neuron had taken such responsibility during the previous adapt cycle when the stimulus was most recently inserted. The division of responsibility stabilizes at the same time that the responses of the individual neurons to their share of the "load" stabilizes. In the case that the training problem is not perfectly separable by this system, it can be shown that the adaptation process tends to minimize error probability.

In a sense, the MADALINE (multiple ADALINES) structure of Figure 6 is 2-layer. The first layer is of adaptive logic elements, the second layer is of fixed logic. There are a variety of fixed logic schemes that could be used on the second layer. Convergent adaption procedures have been devised by M. E. Hoff, Jr., which will be described in his doctoral thesis, that can be used with all possible fixed-logic second layers. A simple fixed-logic element is an "OR" element. If any of the ADALINES produce the +1 output, the OR element gives a system output of +1. During training, if the desired output for a given input pattern is +1, only the one neuron whose confidence level is closest to zero need be adapted if any adaptation is to be done, i.e., if all neurons give -1 outputs. If the desired output is -1, all neurons must give -1 outputs, and any giving +1 outputs must be adapted. Ridgway has proven that this system is convergent.

The memory capacities of MADALINE structures utilizing both the majority element and the OR element have been measured by Koford. Although the logic functions that can be realized with these output elements are different, both types of elements yield structures with the same statistical storage capacity. The average number of patterns that can be adapted to by a MADALINE equals the capacity per ADALINE multiplied by the number of ADALINES. The memory capacity is therefore equal to twice the number of weights.

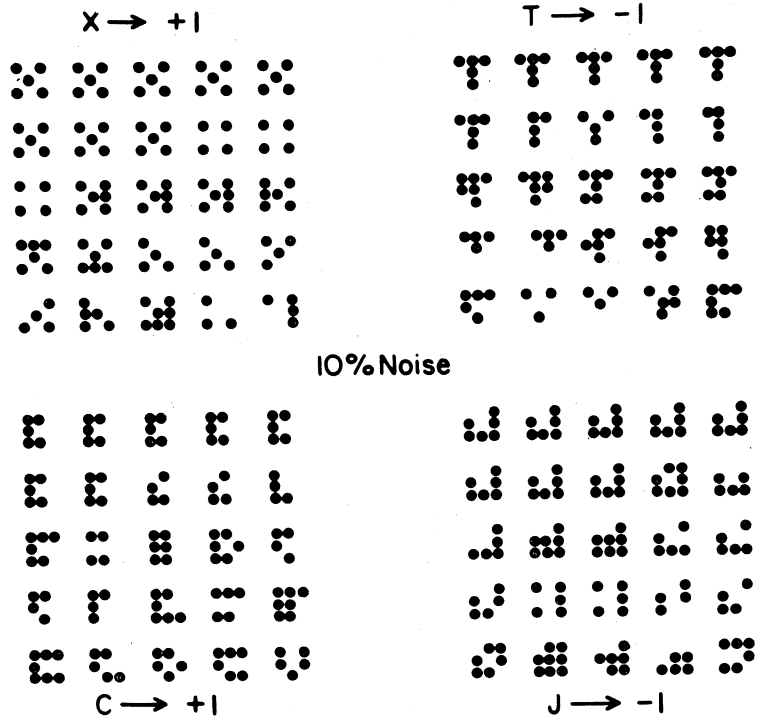
E. Generalization Experiments With ADALINES and Simple Networks of ADALINES

With suitable pattern-response examples and the proper training procedures, generalizations can be trained into ADALINES. The kinds of generalizations to be considered here are concerned with the training of ADALINES to react to patterns and to be statistically insensitive to noise, translation, rotation, and size. ADALINES can be forced to react consistently on a training set of patterns for all possible positions for example,

and then they will react consistently in all positions with high reliability on new patterns never seen before which are quite unrelated to the training set.

(1) Generalization with respect to noise

In Figure 7, a set of patterns is shown that was used in an experiment on generalization for insensitivity to noise. A single 3×3 ADALINE was first trained on 100 noisy X, T, C, J patterns. This problem was able to be solved with a minimum



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	$M = \frac{28-12}{12} = 133\%$

Figure 7. Training with Noisy Patterns.

error rate of 12%. The weights were returned to zero and 10 patterns of the hundred were selected at random and trained into the ADALINE. Then the response of the ADALINE was tested on the full group of 100 patterns. On the average, a set of such experiments involving training with very small sample size produced an error rate of 23 per cent. The theoretical error rate [3] for training with a number of patterns equal to the number of weights of the neuron is twice the minimum error rate or 24 per cent, which checks nicely with the experimental result. If the number of training patterns was increased to 20, the error rate would have been 18 per cent. The number of patterns required to train an ADALINE to discriminate noisy patterns equals several times the number of bits per pattern, roughly the statistical capacity of the neuron.

(2) Generalization with respect to rotation of patterns

Insensitivity to rotation by 90° is a characteristic that can be perfectly trained-into an ADALINE. An experiment was made as depicted in Figure 8 by using the 4 × 4 KNOBBY ADALINE shown in Figure 2. C's rotated in all four positions were trained-in to give the +1 response, while T's were trained-in to give the -1 response in all four rotations. The initial weights were set to zero, and during training, the minimum mean-square error adaption procedure with an adaptive time constant of 32 patterns was utilized. The process converged with the desired responses trained in precisely, and the set of weights shown in Figure 8 resulted. Without further training, new patterns totally unrelated to the training patterns were inserted, and it was observed that not only were the decisions made by the ADALINE perfectly consistent for each pattern over the four rotations, but the four meter readings (confidence levels or analog outputs) for each pattern were identical. The reason for this is simple. Rotation of the weights by 90° yields an identical set of weights. Let the a-matrix represent the set of weights (not including the threshold weight). The threshold weight remains the same for all rotations.

$$[a] = [a]^t = \left[[a]^t\right]^t = \left[\left[[a]^t\right]^t\right]^t \qquad (6)$$

Other training patterns and other numbers of training patterns were used in this experiment, and in each case, after convergence, the same symmetry expressed in Eq. (6) resulted automatically. Adaptation with a time constant, long compared to the number of training patterns, allows the neuron to retain

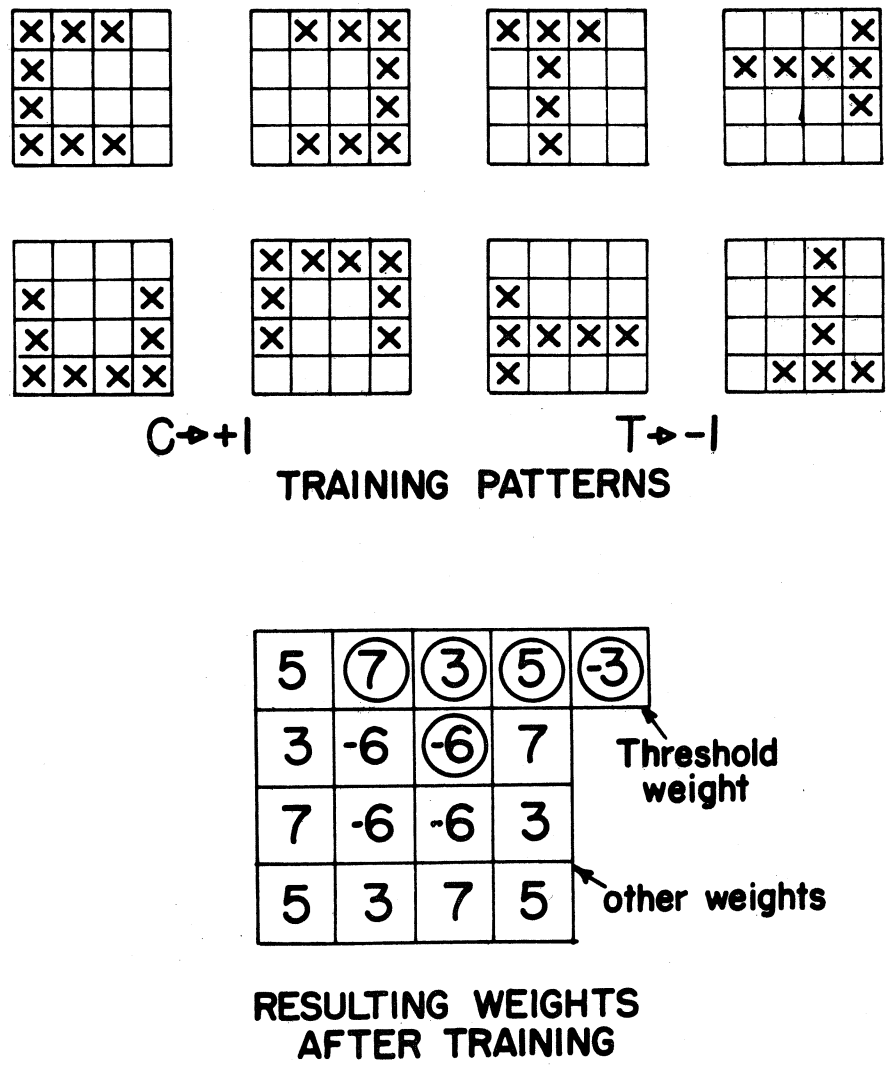


Figure 8. Training-in Insensitivity to Rotation of Patterns by 90°.

responses to all the training patterns essentially equally. Minimization of mean-square error forces the response voltage to each training pattern in all four rotations to be consistent even when it might not be possible for this voltage to be precisely +1 or -1. This forces the symmetry of Eq. (6).

An interesting question is, how many specific responses on the average can be trained-in and yet have the neuron trained to be insensitive to 90° rotation for all patterns. The 4 × 4 neuron has a capacity of 32 patterns. Eight basic patterns on the average

can be trained in, since each basic pattern must be inserted in all four rotations. Another point of view on this question was suggested by Hoff. The four encircled weights and the threshold shown in Figure 8, once chosen, set the rest of the weights when the constraint of Eq. (6) is followed. There are 4 "degrees of freedom" plus the threshold freedom. The number of basic patterns that can be discriminated therefore corresponds to the capacity of a 4-input neuron which is 8 patterns.

The same training procedure could be used to train-in a direct sensitivity to rotation, rather than an insensitivity. The experiment was remade, with a C mapped +1 and a T mapped -1, however with a rotated C mapped -1, and a rotated T mapped +1, etc., the following set of weights resulted.

-11	+9	-2	+11	0
+2	0	0	-9	
-9	0	0	+2	
+11	-2	+9	-11	

threshold weight

The symmetry in the weights can be described by

$$[a] = -[a]^t = \left[[a]^t\right]^t = -\left[\left[[a]^t\right]^t\right]^t \qquad (7)$$

Rotation of any input pattern by 90° causes a sign reversal in confidence level, and therefore an opposite decision.

(3) Generalization with respect to left-right translation

Perfect solutions to the problem of training an ADALINE to be insensitive to left-right pattern translation exist. A solution requires the columns of the a-matrix to be identical. On a 4×4 input array, there is a choice of 4 independent weights, each choice setting a row of weight values. It follows that the statistical discrimination capacity subject to the constraint of insensitivity to left-right translation is that of a 4-input ADALINE or 8 basic patterns. The total capacity of the 4×4 ADALINE is 32 patterns, and this corresponds to the four positional possibilities for each of the 8 basic patterns. Patterns containing +1's in only a single column can have four positions. Patterns having +1's in more than one column can be placed in four positions by considering the input pattern space to be continuous and folded over a cylinder having a vertical axis.

In Figure 9, a group of 4 basic patterns and their desired responses is shown. These were trained-into an ADALINE by slow adaption with the minimum mean-square error procedure. In this case, patterns were not folded over, but were inserted intact, only as they naturally appear. The number of positions for each training pattern is indicated. The resulting set of weights is shown. A very strong tendency for the columns to be identical is evident. With this set of weights, roughly 95% of all new patterns will be mapped consistently in all four positions, some +1, some -1.

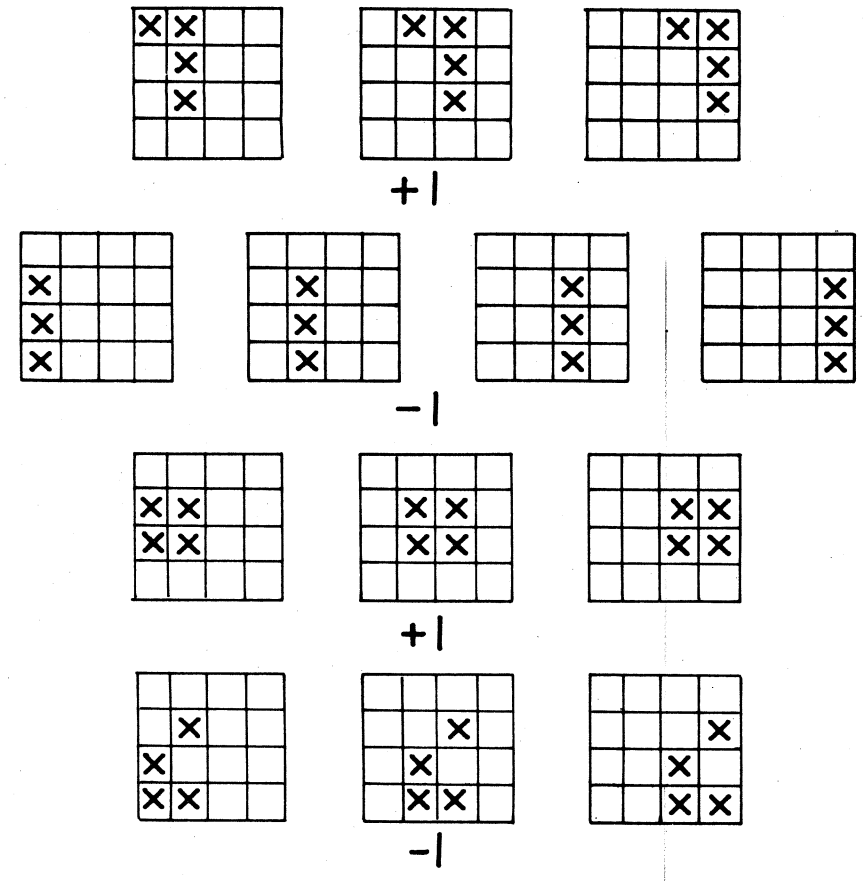
By symmetry, the same training procedures apply to training for insensitivity to up-down motion. If both left-right and up-down insensitivity is desired, the only perfect solution is the relatively trivial one, all weights in the a-matrix being equal. Discrimination is based on pattern "area," the number of +1 pattern bits. More sophisticated discrimination based on pattern features other than area have been made by using two ADALINES. This will be discussed below.

If it were desired that motion by one unit left or right cause a sign reversal, this could be trained-in using a similar procedure. The columns would be alike except for sign change in each element from that of a neighboring column. The capacity would again be four basic patterns. If both up-down and left-right motion should cause a sign change, all elements in the a-matrix should be equal in magnitude, but alternate in sign. Discrimination would be based on area and position.

An experiment was made to train a KNOBBY ADALINE to give a sign reversal for left-right motion and at the same time, to give a sign reversal for rotation by 90°. The pattern T on a 3×2 grid was trained in to produce +1 in the vertical left position, -1 in the vertical right position, etc. The following set of weights resulted. Notice the symmetries and sign alternations.

8	-5	5	-8	-2
5	-1	1	5	
5	1	-1	-5	
-8	5	-5	8	

It was found that approximately 85% of new patterns would consistently produce sign alternation for all possible left-right, up-down, and rotation by 90° motions. The remaining 20% of patterns would be consistent in most situations, perhaps be incorrect in only 2 out of 16 cases. Symmetrical patterns would be perfectly consistent.



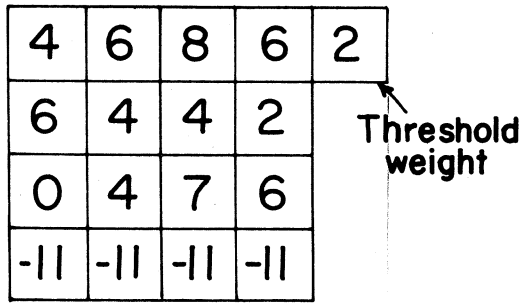
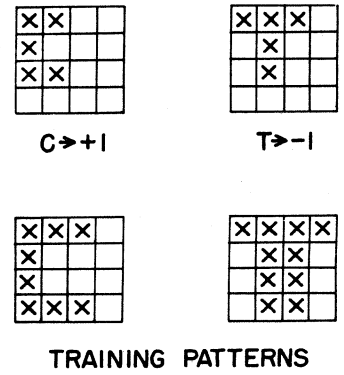


Figure 9. Training-in Insensitivity to Left-Right Pattern Translation.

(4) Generalization with respect to pattern size

An ADALINE can be trained to be highly insensitive to pattern size. The training procedure again requires slow minimum mean-square error adaption. In Figure 10, a set of "small" and "large" patterns is shown that comprised examples for the training experiment. On a 3 × 3 array in the upper left hand corner of a 4 × 4, a T and a C were inserted as shown. In the full 4 × 4 array, expanded versions of these patterns were trained-in to give corresponding responses. After training, it was found that new patterns gave widely fluctuating analog responses. For about 90% of new pattern inputs, the same binary response resulted for the small as for the large versions, and the corresponding confidence levels were extremely close.



To be perfectly insensitive to size, the weights of an ADALINE must be such that an element of area of the small pattern "sees" the same total weight (input patterns are thought of as continuous two-dimensional functions and weights are thought of as continuous distribution functions) as the corresponding area element of the large pattern that it maps into. The only perfect solution for the single adaptive neuron where the small and large fields overlap is the trivial one, all weights zero. The set of weights shown in Figure 10 did not give perfect consistency, yet, since there was not complete overlap between the small and the large pattern fields, good performance was possible.

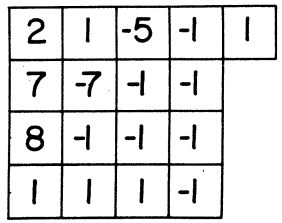


Figure 10. Training-Insensitivity to Pattern Size.

(5) Generalization experiments with two parallel ADALINES

The same kinds of insensitivities can be trained into two or more ADALINES. Larger numbers of specific discriminations can be trained in, and more complex generalizations, such as rotation, translation and size, can be trained in simultaneously.

An experiment was made where T was trained in to give +1 response in a left vertical position on a 4 × 4 array. A J in this

position was trained to give a -1 response. The signs of the responses were reversed by motion one unit to the right and by rotation by 90°. These patterns were inserted in all positions and rotations. The two ADALINE outputs were combined in an "OR" element. Where a given neuron had responsibility, the neuron was adapted slowly with the minimum mean-square error procedure. The resulting set of weights were the following.

7	-5	1	-3	-9	10	6	-1	-11	-10
-1	-3	5	7		2	1	5	-2	
-4	5	-1	5		8	8	-4	-4	
-7	5	5	11		-8	-2	-4	8	

To a good approximation, rotation of a pattern by 90° caused the roles of the two ADALINES to interchange, with a sign reversal. Rotation of the first set of weights by 90° and changing signs gives almost the same weights as the second set. Each neuron need only learn to respond insensitively to left-right or up-down motion. The division of responsibility with respect to rotation occurred naturally and automatically as a result of the training process. Other divisions of responsibility are possible with different initial conditions and training pattern sequences. About 90% of all new patterns were mapped consistently with the above set of weights.

A wide variety of insensitivity (or sensitivity) training experiments have been made with single ADALINES and multi-ADALINES, and the results reported in this section are representative of what should be expected in performance. In many of these experiments, the relationships among the weights that result are simple and could have been determined beforehand. However, it should be realized that by using a single training routine, a wide variety of learning and generalization processes can be induced merely by designing appropriate sets of training patterns. All of these generalization experiments were done on KNOBBY ADALINES, with initial weight settings of zero. In general, more training patterns would be required where it is not practical to set all weights to zero. The total number of patterns required per ADALINE would be at least equal to the number of weights. The objective is to make the responses to patterns not specifically trained-in to depend only on the training experience and not on the initial conditions.

F. A Convergent Adaption Procedure For Multi-Layered Networks

In Figure 11, a network consisting of two layers of adaptive ADALINES is shown. A variety of basic patterns are to be

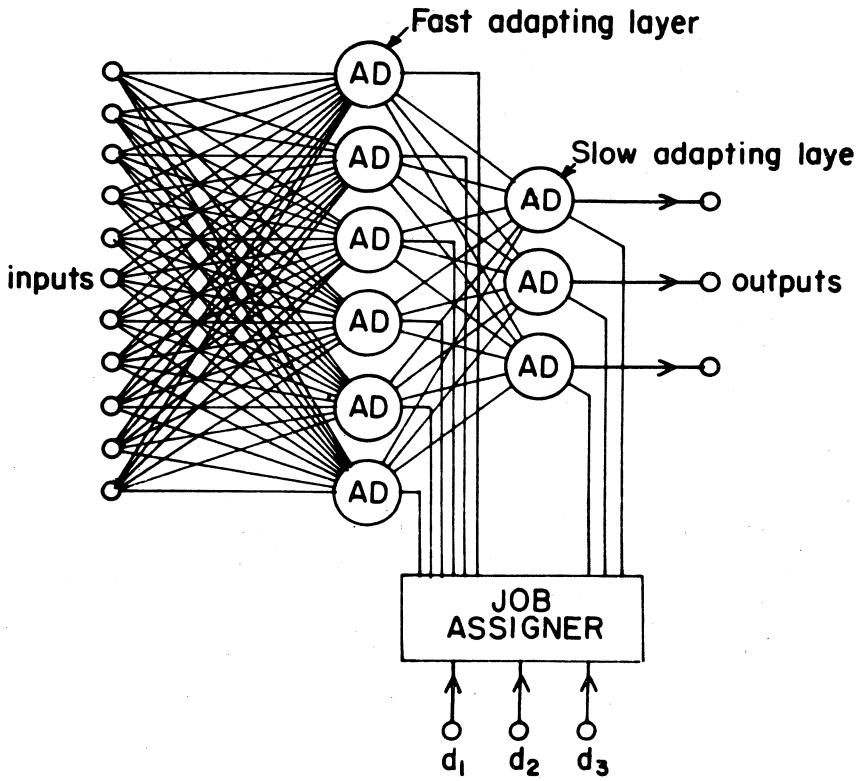


Figure 11. A Two-Layer Network of Adaptive ADALINES.

classified into one of 8 categories (the system as shown has 3 output bits). Each of the basic patterns could occur in multitudes of positions, rotations, sizes, and with varying amounts of noise. The objective is to teach the system to identify these patterns correctly by showing it only a very small randomly-selected fraction of the total number of possible input patterns. If the first layer could be trained to produce a set of output signals which are close to being independent of rotation, translation, size, and noise, the second layer could be trained to produce the specific desired responses. To do all this, some of the ADALINES shown in Figure 11 might have to be MADALINES, or a third (or more) adaptive layer might have to be added. A procedure for adapting all the neurons in both layers must be such that it is possible to train-in a large number of specific responses, that the desired generalizations take place, and that the "load" is shared among the neurons automatically.

For simplicity, let the input be a 4×4 array, and let the input patterns be noise-free and in one position, only they could be

rotated in any one of four orientations spaced 90° apart. During the training process, a pattern is applied, and a 3-bit desired response is supplied. The system is not told which of the basic patterns is applied, nor which rotation the pattern is at. Only the 3-bit classification of the pattern is given. Typically, the weights have random initial values.

An adaption procedure that has been found by experiment to work well is the following. A pattern and the desired response bits are applied. The first layer is adapted first in an attempt to get the second layer outputs to agree with the desired outputs. The first-layer neurons to be adapted are chosen to minimize the number of adaptions, and if there are alternate choices with equal numbers of neurons to be adapted, the choice is made to minimize the total distance that the confidence levels need be displaced. The particular choice of first-layer neurons to be adapted and the directions that they should be adapted in is based on knowledge of the weights of the second-layer neurons. If no combination of adaptions of the first-layer neurons produces the desired outputs, the second-layer neurons should then be adapted to yield the desired outputs. This procedure has the tendency to force the first-layer neurons to produce independent responses which are insensitive to rotation. All adaptions are minimum mean-square error. Again, *responsibility is assigned to the neuron or neurons that can most easily* assume it.

Wide varieties of specific responses have been successfully trained into a small-scale version of the system of Figure 11. Three 4×4 ADALINES were used in the first layer, and a single 3-input ADALINE was used in the second layer. This is like a 3-neuron MADALINE with an adaptive second layer. The above procedure and many variants upon it are currently being tested with larger networks, for the purpose of studying memory capacity, learning rates, and relationships between structural configuration, training procedure, and nature of specific responses and generalizations that can be trained in.

G. Realization of Adaptive Circuits by the Electrochemical Memistor

In large networks of adaptive neurons, it is imperative that the adapt processes be fully automated. The structure of the ADALINE neuron and the adaption procedures used with it are sufficiently simple, that it has been possible to develop an electronic automatically-adapted neuron which is reliable, contains few parts, and is suitable for mass production. In such a neuron, it was necessary to be able to store weight values, analog quantities which could be positive or negative, in such a way that these values could be changed electronically.

A new electrochemical circuit element called the MEMISTOR (a resistor with memory) has been devised by this author and M. E. Hoff for the realization of automatically-adapted ADALINES. The MEMISTOR provides a single variable gain element. Each neuron therefore employs a number of MEMISTORS equal to the number of input lines, plus one for the threshold. A variety of magnetic devices functionally akin to the MEMISTOR have been developed, and one of the most successful of these is the multiaperature MAD device of A. E. Brain [4]. This has been used in a Rosenblatt Perceptron [5] at the Stanford Research Institute. The MEMISTOR has the disadvantage of being about an order of magnitude slower in adapt speed, but has the advantages of being about an order of magnitude faster in sensing speed, and of using much simpler, more reliable circuitry. It is cheaper to implement, and it requires about 4 orders of magnitude less power to operate.

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 terminal, rather than by its instantaneous value, as in the transistor. Highly reproducible elements have been made which are continuously variable (thousands of possible analog storage levels), and which typically vary in resistance from 50 ohms to 2 ohms, and cover this range in about 5 seconds with several tenths of a milliampere 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 Figure 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 Figure 12.

In the circuit of Figure 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

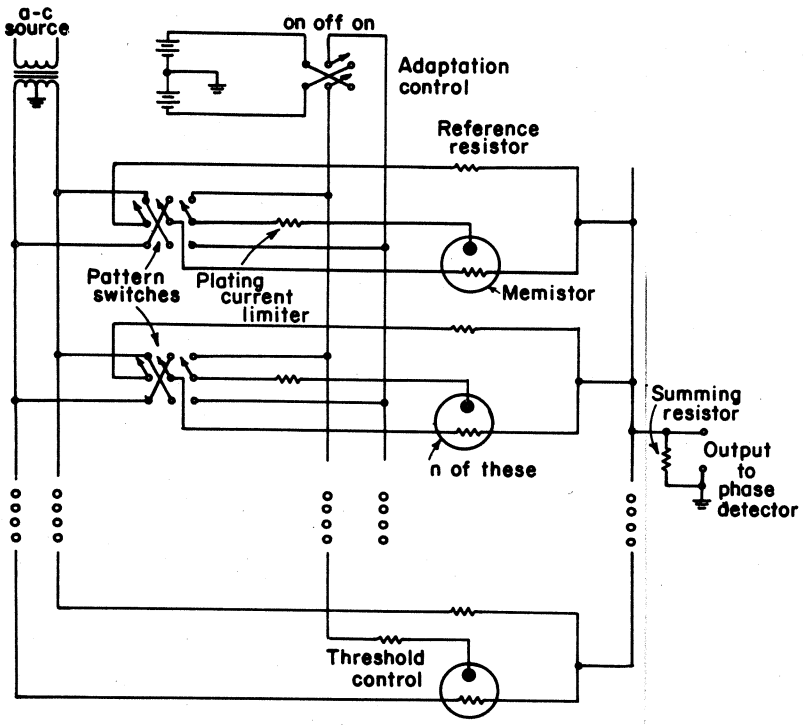


Figure 12. Circuit of a MEMISTOR ADALINE.

resistor. The currents are proportional to the conductance 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.

In Figure 13, a photograph of the latest development in MEMISTORS is shown. MEMISTORS of this type are now commercially available. On a single sheet of glass, 21 elements are printed. The actual substrates can be seen in the cell cylinders. Caps and anodes are yet to be installed, and the entire unit is then encapsulated in epoxy. Each cell has a volume of about 2 drops. Similar cells that were made over eleven months ago are still working like they did when first constructed, and have been taken through hundreds of thousands of plating-stripping cycles with no effect upon electrical characteristics. These cells are essentially insensitive to temperature and to shock and vibration. They have been stored with no deterioration over temperatures of -15° C to +75° C.

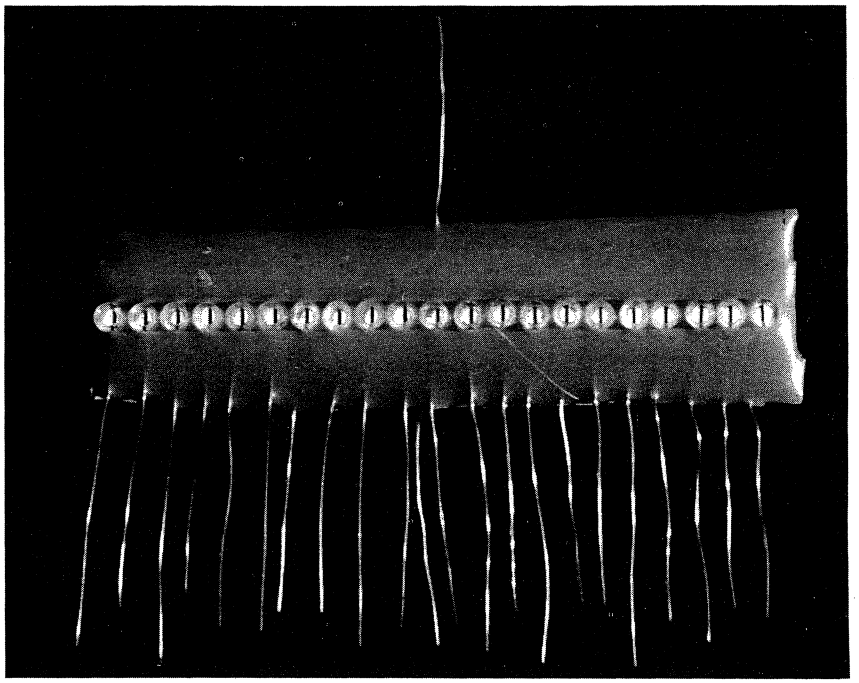


Figure 13. A Partially-Fabricated Sheet of Memistors.

In Figure 14, a photograph of MADALINE I is shown. This machine, constructed at Stanford University, is the largest MEMISTORIZED machine that has been built. It has 6 ADALINES that can be independently adapted. The inputs are in parallel, and the present input array is 4×4. This is being expanded to a 7×7. It contains 102 MEMISTORS.

This machine was constructed rapidly over a one and one half month period. The MEMISTORS were not tested before installation in the machine, and some were defective at manufacture. A number of wiring errors were made; some weights were adapting to diverge rather than converge. There were a number of short circuits, open circuits, cold solder, joints, etc. This machine worked well when first turned on, and has functioned with very little attention over the past 8 months. It took two weeks of experimentation before suspicions were aroused and the weights were checked. Twenty-five per cent of them were not adapting. Yet the machine was able to adapt around its own internal flaws and to be trained to make very complex pattern discriminations. These errors were corrected, and the capacity of the machine increased accordingly.

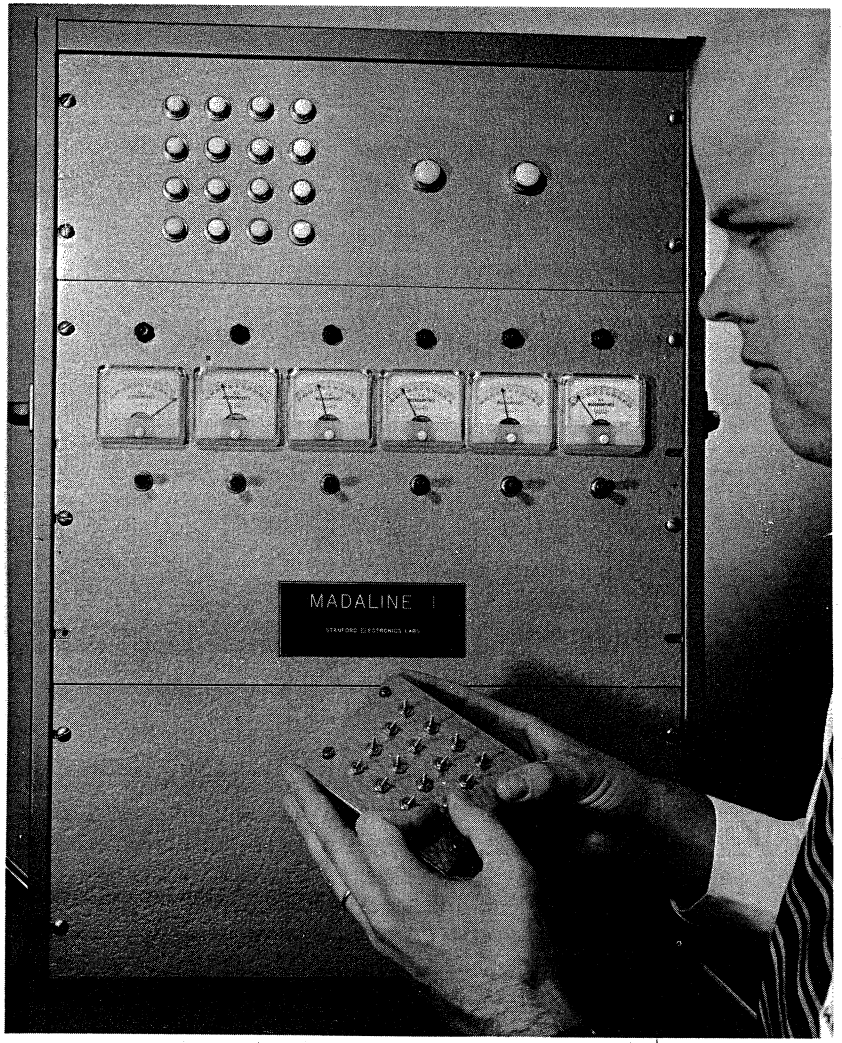


Figure 14. MADALINE I and W. C. RIDGWAY, III.

At the time of this writing, MADALINE I has just been put under the control of an IBM 1620 computer. The computer stores the patterns and desired responses, and controls the training of the neurons. It tabulates the number of patterns seen and number of adaptations made, and error probability as the learning process progresses. The combination of the neurons and the computer is 3 times faster than is the digital simulation of the same neurons on the 1620 alone. A larger MADALINE is being

planned which will contain 1500 adaptive weights. When connected to the 1620 computer, it will be faster at neuron simulation than an IBM 7090. On this scale, a neuron simulation facility consisting of a small computer and memistorized neurons is ten times cheaper than an all-digital simulation facility.

The fundamental objective in connecting adaptive neurons to a computer is to develop a new type of computer, one as different from the digital computer as the digital computer is different from the analog computer. This new type of machine might be called the *Adaptive Computer*. The basic "flip-flop" for this machine is ADALINE. The adaptive computer is taught rather than programmed to solve problems. The job of the "programmer" is to establish suitable training examples. This machine will be taught by men (so that it will solve the problems of men) in the environment and with the language of men, not with a machine language. The learning experience derived from human teachers will provide reasonable initial conditions, upon which the machine could subsequently improve from its own systematic experimentation and experience gathering.

ACKNOWLEDGMENT

This work was performed under 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 Research), and Air Force Contract AF33(616) 7726 supported by Aeronautical Systems Division, Air Force System Command, Wright-Patterson Air Force Base.

REFERENCES

1. B. Widrow, "Adaptive Sampled-Data Systems—A Statistical Theory of Adaption," 1959 WESCON Convention Record, Part 4.
2. W. C. Ridgway III, "An Adaptive Logic System with Generalizing Properties," Technical Report No. 1556-1, Stanford Electronics Laboratories, Stanford University, Stanford, California; April, 1962.
3. B. Widrow and M. E. Hoff, "Adaptive Switching Circuits," 1960 WESCON Convention Record, Part IV, pp. 96-104; August 23, 1960.
4. A. E. Brain, "The Simulation of Neural Elements by Electronical Networks Based on Multi-Aperature Magnetic Cores," Proc. IRE, Vol. 49, pp. 49-52; January, 1961.
5. F. Rosenblatt, "Principles of Neurodynamics," Spartan Books, Washington, D. C., 1962.

Conference on

SELF-ORGANIZING SYSTEMS 1962

Edited By:

MARSHALL C. YOVITS, Office of Naval Research
GEORGE T. JACOBI, Armour Research Foundation
GORDON D. GOLDSTEIN, Office of Naval Research

SPARTAN BOOKS
6411 CHILLUM PLACE, N. W. • WASHINGTON 12, D. C.

DAK