Pattern Recognition and Adaptive Control

BERNARD WIDROW
MEMBER IEEE

Summary: Adaptive or self-optimizing systems utilize feedback principles to achieve automatic performance optimization. These principles have been applied to both control systems and adaptive logic structures. The Adaline (adaptive linear threshold element) is essentially the same as an adaptive sampled-data system with quantized input and output signals. A digital controller made of adaptive neurons comprises a pattern-recognizing control system. When the state of a control system is represented as a pattern, learning to make the control decisions actually becomes the same as learning to classify the patterns.

ADAPTIVE or self-optimizing systems have the ability to modify their structures automatically to achieve a near optimal performance. An adaptive capability is particularly useful in cases where the nature of system input signals is not known, even statistically. In other cases, the nature of the input might be known to be changeable; for example, input statistics can be nonstationary. An adaptive system that continually searches for the optimum within its allowed class of possibilities by an orderly trial-and-error process would have a performance vastly superior to that of a fixed system in many of these instances.

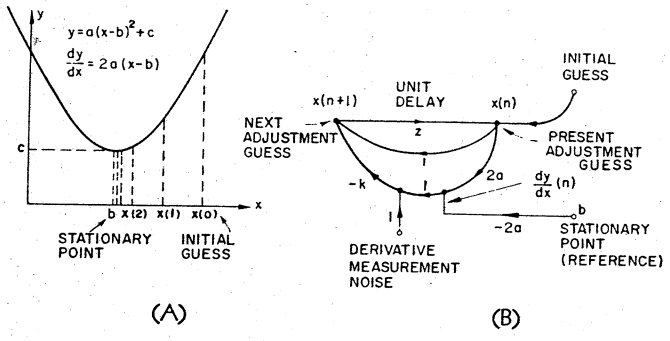


Fig. 1. One-dimensional surface searching

A—Side view of graph
B—Top view of graph

Several ways of classifying adaptation schemes have been proposed in the literature. This paper will consider only closed-loop and open-loop adaptation processes. The open-loop adaptation process involves making measurements of input or environmental characteristics, applying this information to a formula or a computational algorithm, and using the results to set the adjustments of the adaptive system. Closed-loop adaptation, on the other hand, involves automatic experimentation with these adjustments to optimize a measured system performance. Where open-loop adaptation can be used, it is usually simpler to implement; closed-loop adaptation is more fundamental and more generally applicable.

One purpose of this paper is to study adaptation, particularly closed-loop adaptation, with the objective of gaining an understanding of how automatic system synthesis can be achieved using "performance feedback."

Feedback and Trial-and-Error Processes

Iterative or trial-and-error processes are integral parts of adaptive systems. They provide the mechanism of adaptation. It is often convenient to represent such processes as feedback systems: the error of trial and error is analogous to the "error" of feedback control. Many of the relaxation and iterative methods employed by numerical analysts appear to be linear feedback systems when represented in this manner. Surface exploration for stationary points is one example of importance in this discussion.

Many of the commonly used gradient methods search the surfaces by making changes in the independent variables (starting with an initial guess) in proportion to measured partial derivatives to obtain the next guess, and so forth. These methods give rise to geometric (exponential) decays in the independent variables as they approach a stationary point for second-degree or quadratic surfaces. One-dimensional surface searching is illustrated in Fig. 1. The surface being explored in Fig. 1 is given by equation 1. The first and second derivatives are given by equations 2 and 3.

$$y=a(x-b)^2+c \tag{1}$$

$$\frac{dy}{dx}=2a(x-b) \tag{2}$$

$$\frac{d^2y}{dx^2}=2a \tag{3}$$

A sampled-data feedback model of the iterative process is shown in Fig. 1(B).[1-3] The flow graph can be reduced, and the transfer function from any point to any other point can thus be found. The resulting characteristic equation can be expressed as follows:

$$(2ak-1)z+1=0 \tag{4}$$

In order to choose the loop gain k to get a specific transient decay rate, one would have to measure the second derivative, $2a$, at some point on the curve.

The first and second derivatives are given by equations 5 and 6. These relations are precise for parabolas, and are approximate for higher-degree curves (Fig. 2).

$$\left.\frac{dy}{dx}\right|_{x_B}=\frac{1}{2\delta}(C-A) \tag{5}$$

$$\left.\frac{d^2y}{dx^2}\right|_{x_B}=\frac{1}{\delta^2}(C-2B+A) \tag{6}$$

A 2-dimensional parabolic surface is described by

$$y=ax_1^2+bx_2^2+cx_1+dx_2+ex_1x_2+f \tag{7}$$

the partial derivatives by

$$\frac{\partial y}{\partial x_1}=2ax_1+c+ex_2$$

$$\frac{\partial y}{\partial x_2}=2bx_2+d+ex_1 \tag{8}$$

and the second partial derivatives by

$$\frac{\partial^2 y}{\partial x_1^2}=2a$$

$$\frac{\partial^2 y}{\partial x_1 \partial x_2}=e$$

$$\frac{\partial^2 y}{\partial x_2^2}=2b \tag{9}$$

A vector flow-graph model of a 2-dimensional iterative surface-searching process is given in Fig. 3(A). The branches in this graph are capable of carrying 2-dimensional samples, indicated by column matrices. This flow graph can be reduced straightforwardly by using the rules of matrix algebra. There are as many natural frequencies (decay rates) as there are independent co-ordinates. The multidimensional loop gain in this case is determined by choice of the matrix of k's.

There are many surface-searching methods in common use. Among these are the method of steepest descent, Newton's method, and the Southwell relaxation method. The flow graph of Fig. 3(A) can represent Newton's method, where the matrix of k's is the inverse of the matrix of second partials. Multidimensional transients die out completely in one step. In a modified Newton's method, the same matrix of k's is scaled by a factor less than unity. Transients die out geometrically, not in one step, and are of a single time constant. Successive adjustments proceed along a straight line in multidimensional space from the initial

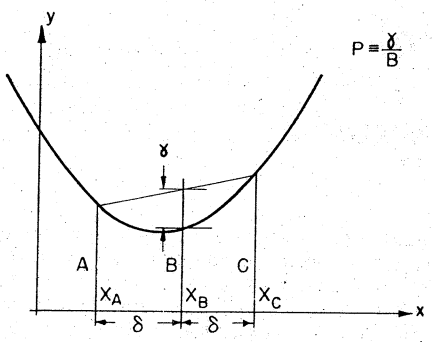


Fig. 2. Measurement of derivatives; definition of the perturbation

A paper recommended by the AIEE Feedback Control Systems Committee and approved by the AIEE Technical Operations Committee for presentation at the AIEE Joint Automatic Control Conference, New York, N. Y., June 27-29, 1962. Manuscript submitted June 29, 1962; made available for printing September 20, 1963.

BERNARD WIDROW is with Stanford University, Stanford, Calif.

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 under Air Force Contract *AF33(616)7726*, supported by Aeronautical Systems Division, Air Force System Command, Wright-Patterson Air Force Base, Ohio. The author would like to thank Prof. R. Cannon for suggesting the use of the 1-dimensional broom-balancing process for adaptive control.

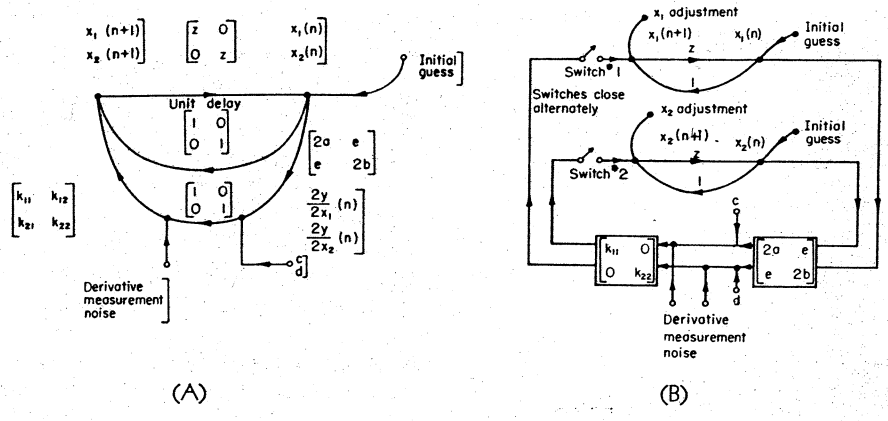


Fig. 3. Two-dimensional surface-searching models

A—Flow graph representing Newton's method
B—Flow graph representing Southwell's method

guess to the stationary point. Cross-coupling among the co-ordinates is eliminated.

The flow graph of Fig. 3(A) can also represent the method of steepest descent. Here the matrix of k's is a diagonal one, with identical elements on the main diagonal. This corresponds to vector changes in adjustment being proportional to the successive local gradient vectors. Cross-coupling is present.

The flow graph of Fig. 3(B) represents surface searching by the Southwell procedure. Adjustment along each co-ordinate is set every time to minimize y. This corresponds to the matrix of k's being diagonal, with $k_{11}=1/2a$ and $k_{22}=1/2b$. Cross-coupling is present, but transients are of a single time constant.

Approximate Analysis of an Adaptive Sampled-Data Predictor

Consider the general linear sampled-data system formed of a trapped delay line, shown in Fig. 4. This system is intended to be a statistical predictor. The present output sample $g(n)$ is a linear combination of present and past input samples. The constants in this combination are h_0, h_1, h_2, etc., the predictor impulse-response samples, or the gains associated with the delay-line taps; their choice constitutes the variable part of the predictor design. They may be adjusted by applying a mean square reading meter to $\epsilon(n)$, the difference between the present input and the delayed prediction. This meter will measure mean square error in prediction. Then $h_0, h_1, h_2, \ldots$, are adjusted until the meter reading is minimized.

The problem of adjusting the h's is not trivial, because their effects upon performance interact. Suppose that the predictor has only two impulses in its impulse response, h_0 and h_1. The mean square error for any setting of h_0 and h_1 can be readily derived:

$$\epsilon(n)=f(n)-h_0f(n-1)-h_1f(n-2)$$

$$\bar{\epsilon^2}(n)=\phi_{ff}(0)h_0^2+\phi_{ff}(0)h_1^2-2\phi_{ff}(1)h_0-2\phi_{ff}(2)h_1+2\phi_{ff}(1)h_0h_1+\phi_{ff}(0) \quad (10)$$

The discrete autocorrelation function of the input is $\phi_{ff}(k)$. The mean square error is a parabolic function of the predictor adjustments h_0 and h_1.

The optimum m-impulse predictor can be derived analytically by setting the partial derivatives of $\bar{\epsilon^2}$ of equation 10 equal to zero. This is the discrete analog of Wiener's optimization[4] of continuous filters. Finding the optimum system experimentally is the same as finding a minimum of a paraboloid. This could be done manually by having a human operator read the meter and set the adjustments; it could be done automatically by using the iterative gradient methods for surface searching, as described in the previous section. When either of these schemes is employed, an adaptive system results that consists essentially of a "worker" and a "boss." The worker in this case predicts, whereas the boss has the job of adjusting the worker.

Fig. 5 is a block-diagram representation of such a basic adaptive unit. The boss continually seeks a better worker. Adaptation is a multidimensional feedback process. The error signal is the gradient of mean square error with respect to adjustment.

Noise enters the adaptation feedback system because the input process cannot be continued indefinitely for each measurement of mean square error (A, B, and C of Fig. 2, needed for gradient measurement); it thereby places a basic limitation upon adaptability. It will be shown that the slower the adaptation is, the more precise it will be. The faster the adaptation, the noisier (and poorer) the adjustments will be.

Consider that the adaptive model has only a single adjustment. A plot of mean square error versus h_0 for this simplest system would be a parabola, analogous to the parabola of Fig. 1. During each cycle of adjustment, the derivative of $y=\bar{\epsilon^2}$ with respect to $x=h_0$ would have to be measured according to the scheme of Fig. 2.

Noise in the system adjustment causes loss in steady-state performance. It is useful to define a dimensionless parameter M, the "misadjustment," as the ratio of the mean increase in mean square error to the minimum mean square error. It is a measure of how the system performs on the average, after adapting transients have died out, compared with the fixed optimum system. With regard to the curve of Fig. 1,

$$M=\frac{\bar{y}-c}{c} \quad (11)$$

Consideration of equation 1 shows that $(\bar{y}-c)$, the average increase in y, is equal to the variance in x multiplied by a. This variance is due to derivative measure-

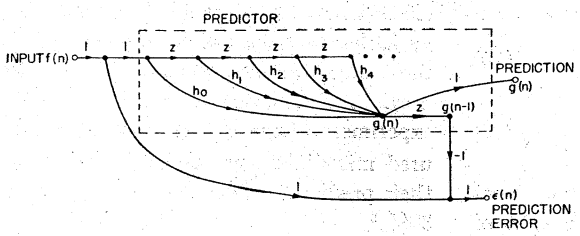


Fig. 4 (left). Adjustable sampled-data predictor

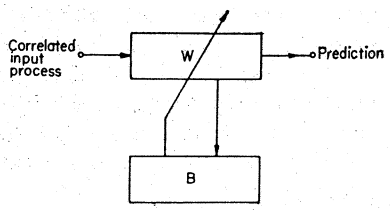
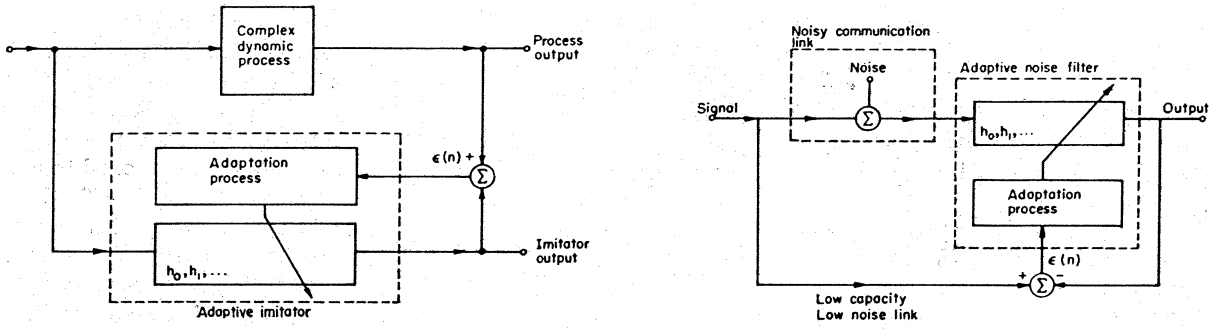


Fig. 5 (right). Adaptive predictor



A—Adaptive imitator

B—Adaptive noise filter

Fig. 6. Adaptive system

ment noise which propagates in the iterative surface-searching process.

The noise propagation path is shown in the flow graph of Fig. 1(B). Assuming that derivative measurement noises are statistically independent from one iteration cycle to the next, the variance in x equals the variance in derivative noise multiplied by $1/(8a^2r)$, which is an approximation to the sum of squares of the impulses of the impulse response from the noise injection point to the adjustment x. The time constant τ is defined so that, if $\tau=1$, adaptation transients decay by a factor $(1/\epsilon)$ with each iterative cycle.

Equation 5 gives the derivatives as the difference between forward and backward measured values of y multiplied by $1/2\delta$. Noise in the measurements of y (due to finite sample size) causes noisy derivative measurements. A detailed analysis of the variance in derivative measurement is given in reference 5. The result is that

$$\text{Variance in derivative measurement} = \frac{ac}{NP} \tag{12}$$

where N is the number of forward or backward measurements per cycle, and P is the perturbation (a dimensionless measure of system disturbance from derivative measurement). Equation 12 is based on several assumptions: that the adjustment x is in the vicinity of the minimum, that the prediction error signal is Gaussian-distributed (equation 12 is quite insensitive to the shape of this distribution density, however), and that the prediction error samples are uncorrelated (correction for correlation less than 90% is very small).

If the nature of the physical process permits data repeating, i.e., if it is possible to apply the same input data to the system for both forward and backward measurements, the variance of the derivative measurement noise will not depend upon the amplitude of the perturbation. When the assumptions made previously are repeated, the expression for the variance with data repeating becomes

$$\text{Variance in derivative measurement} = \frac{4ac}{N} \tag{13}$$

It should be noted that in this case N is the total number of error samples per iteration cycle. The misadjustment equals $1/8a\tau c$ multiplied by the variance in derivative measurement noise. Accordingly,

$$M = \frac{1}{8N\tau P} = \frac{1}{4(2N\tau)P} \tag{14}$$

For the data repeating case,

$$M = \frac{1}{2(N\tau)} \tag{15}$$

The $N\tau$ product is related to the total number of samples seen by the system in adapting to a step transient in input process statistics. A given effect could be achieved by using many samples per cycle (large N) and few cycles with large steps to adapt (small τ), or by using few samples per cycle (small N) and proceeding towards the optimum with small steps (large τ).

Let the number of samples that elapse in one time constant of adaptation be called the adaptation time constant Γ. Where data repeating is not practiced, $\Gamma=2N\tau$. Where data is repeated, $\Gamma=N\tau$. Equations 14 and 15 become equations 16 and 17, respectively:

$$M = \frac{1}{4\Gamma P} \tag{16}$$

$$M = \frac{1}{2\Gamma} \tag{17}$$

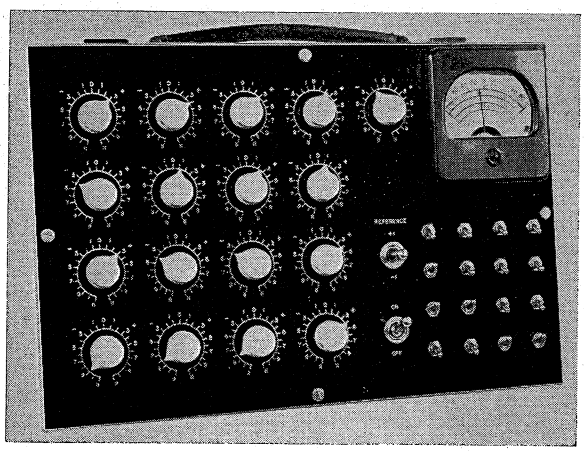
This can be applied to multidimensional adaptation by using the flow graphs of Fig. 3. Let n be the number of adjustments. The misadjustment increases with n^2 when Newton's method is used; it increases with n when data are repeated for any single time-constant method. One such method, Southwell's, is easy to implement because there is no matrix inversion. The misadjustment is given by multiplying equation 17 by n.

These principles may be applied in a variety of situations, two of which are illustrated in Fig. 6. Performance feedback is used in the system of Fig. 6(A) to achieve imitation of an unknown complex system. The adaptive system learns of the characteristics of the unknown system by imitating its behavior as best it can. If the input is stationary and the unknown system is linear, the mean square error will be a parabolic function of the adjustments. A combination of imitation and prediction enables an adaptive system to predict the output of an unknown dynamic system by making use of both its input and output signals. A conventional predictor would use only the output signal. In Fig. 6(B), a scheme is shown which combines a low-noise low-capacity link for performance feedback with a high-capacity noisy communication link. An adaptive filter is used to separate noise and signal. The mean square error is again a parabolic function of the adjustments, and the rate of adaptation is limited by the low-capacity link.

The misadjustment gives a measure of the effectiveness of adaptation. It gives no information on the magnitude of the minimum mean square error or on the effectiveness of the choice of adjustment variables (whether there are enough of them and whether they are the best to use). The results of many simulated experiments have shown that the measured misadjustments usually differ from their predicted values by less than 20 or 30%.[5]

Misadjustment formulas are quite accurate when applied to the situations for which they have been derived. These first-order formulas serve as rules of thumb when performance criteria other than minimization of mean square error are used, and when the worker is nonlinear.[6] With these principles, solutions to many practical problems in the control and communications fields are attainable. As evidence of this, a 10-impulse filter could adapt to a major change in input process statistics after seeing 200 process samples and would have a steady-state misadjustment of about 10%.

Fig. 8. Knobby Adaline



Adaline, an Adaptive Logic Element

Another application of these principles has been their use in adaptive switching circuits. Performance feedback can be used to adapt logical learning machines. In Fig. 7, a combinatorial logical circuit called Adaline is shown; this circuit is analogous to the adaptive sampled-data systems described, although with quantized inputs and output.

The binary signals on the individual input lines have values of $+1$ or -1, rather than the usual values of 1 or 0. Within the element, a linear combination of the input signals is formed. The weights are the gains $a_1, a_2, \ldots$, 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. A constant added to the linear combination of input signals varies with a_0.

For fixed gain settings, each of the 2^4 possible input combinations would cause either a $+1$ or -1 output. Thus, all possible inputs are classified into two categories. The input–output relationship is determined by choice of the gains $a_0, \ldots, a_4$. In the adaptive element, these gains are set during the training procedure.

In general, there are 2^{2^4} different input–output relationships or truth functions by which the four input variables can be mapped into the single output variable. Only a subset of these, the linearly separable truth functions,[7,8] can be realized by all possible choices of the gains of the element in Fig. 7.

Although this subset is not all-inclusive (it becomes a vanishingly small fraction of all possible switching functions as the number of inputs increases), it is a useful subset and is "searchable." In other words, 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 which is quite simple to implement and which can be analyzed by statistical methods that were originally developed for the analysis of adaptive sampled-data systems.

An adaptive pattern classification machine has been constructed for the purpose of illustrating adaptive behavior and artificial learning. This machine, which is an adjustable threshold element called "Knobby Adaline," is illustrated in Fig. 8.

During a training phase, simple geometric patterns are fed to the machine by setting the toggle switches in the 4×4 input switch array. The system learns something from each pattern and accordingly experiences a small 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 1-pass basis to make the iterative process converge 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.

In the actual machine, the quantizer is not built in as a device, but is effected by the operator in viewing the output meter. Different quantizers (2-level, 3-level, or 4-level) are realized by using the appropriate meter scales. Adaline can be used to classify patterns into several categories by using multilevel quantizers and by following exactly the same adaptive procedure.

The iterative searching (training) routine is as follows. A pattern is fed to the 4×4 machine. All gains, including the threshold level, are to be changed by the same absolute magnitude so 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

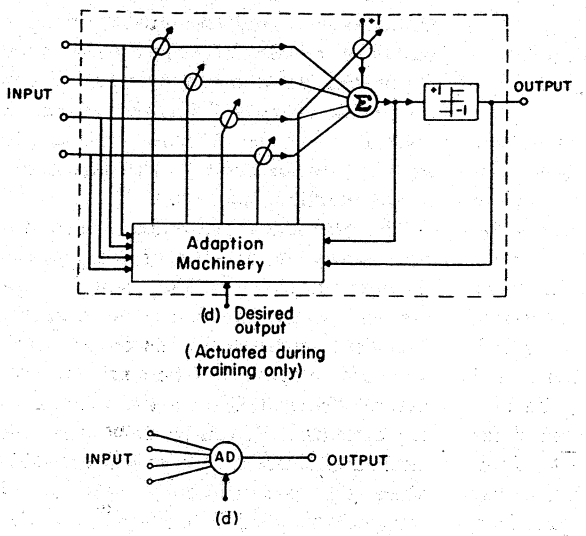


Fig. 7. Adaptive threshold element

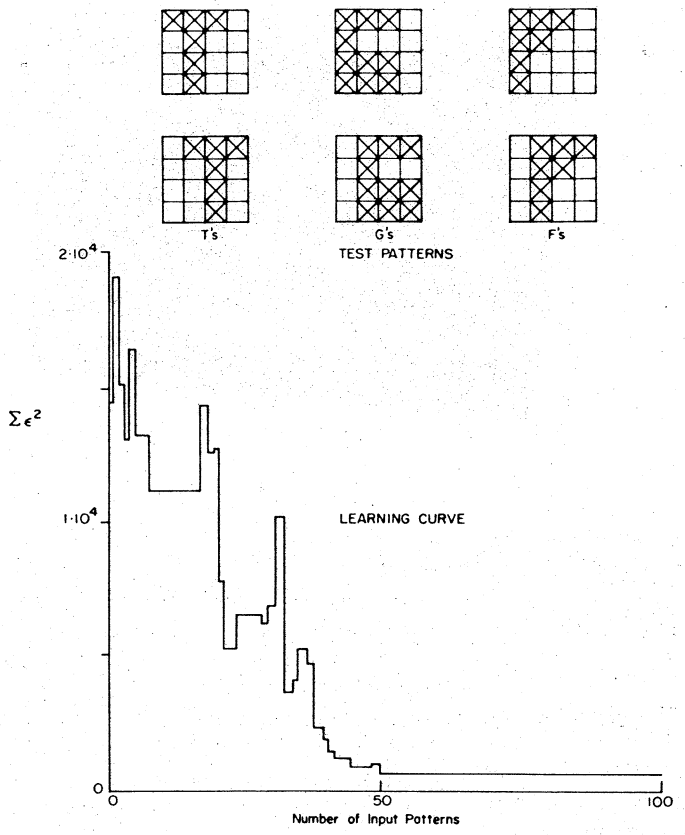


Fig. 9. Relations between digital errors and analog errors

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 will be zero. The weights associated with switches in the "up" position (+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 in the opposite direction from the desired meter needle rotation. The next pattern and its desired output are 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 a like manner. Convergence is indicated by small errors (before adaptation), with small fluctuations about stable weight values. It may be noted that adaption is indicated even if the quantized neuron output is correct. If, for example, the desired response is +1, the element 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 later in the paper.

The results of a typical adaption on six noiseless patterns are given in Fig. 9. During adaption, the patterns were selected in a random sequence, and were classified into three categories. Each T was to be mapped to +30 on the meter dial, each G to 0, and each F to −30. After each adaptation, as a measure of performance, all six patterns were read in (without adaptation) and six errors were read. The sum of their squares, denoted by $\Sigma\epsilon^2$, was computed and plotted. Fig. 9 shows the learning curve for the case in which all gains were initially zero.

Statistical Theory of Adaptation for Adaptive Threshold Elements

The analog error signal measured and used in adaptation is the difference between desired output and the sum before quantization. This error is indicated by ϵ in Fig. 10. The actual error, indicated by ϵ_n in Fig. 10, is the difference between the digital output and the desired output. The object of adaption is to find, given a collection of input patterns and the associated desired outputs, the best set of weights $a_0, a_1, \ldots, a_m$ to minimize the mean square of the error $\overline{\epsilon_n^2}$. Individual errors could only have the values of +2, 0, and −2 with a 2-level quantizer. Minimization of $\overline{\epsilon_n^2}$ is therefore equivalent to the minimizing of the average number of decision errors.

The simple adaption procedure described in the paper minimizes $\overline{\epsilon^2}$ rather than $\overline{\epsilon^{u2}}$. The analog error ϵ has a zero mean (a consequence of the minimization of ϵ^2) and will be assumed to be Gaussian-distributed. By making use of certain geometric arguments, it can be shown that $\overline{\epsilon^2}$ is a monotonic function of $\overline{\epsilon^2}$ under most conditions and that minimization of $\overline{\epsilon^2}$ is equivalent to minimization of $\overline{\epsilon_n^2}$ and to minimization of the probability of error. The ratio of these mean squares has been calculated; it is plotted in Fig. 10 as a function of error probability.

Given any collection of input patterns and the associated desired outputs, the measured mean square error $\overline{\epsilon^2}$ must be a quadratic 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, which have values of +1 or −1, represent the n input components numbered in a fixed manner. The kth error is

$$\epsilon(k) = d(k) - a_0 - a_1 s_1(k) - a_2 s_2(k) - \ldots - a_n s_n(k) \quad (18)$$

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

$$\epsilon^2(k) = d^2(k) + a_0^2 + s_1^2(k)a_1^2 + s_2^2(k)a_2^2 - 2d(k)a_0 - 2d(k)s_1(k)a_1 - 2d(k)s_2(k)a_2 + 2s_1(k)a_0a_1 + 2s_2(k)a_0a_2 + 2s_1(k)s_2(k)a_1a_2 \quad (19)$$

The mean square error averaged over k is

$$\overline{\epsilon^2} = a_0^2 + \phi(s_1, s_1)a_1^2 + \phi(s_2, s_2)a_2^2 - 2\bar{d}a_0 - 2\phi(d, s_1)a_1 - 2\phi(d, s_2)a_2 + 2\bar{s}_1a_0a_1 + 2\bar{s}_2a_0a_2 + 2\phi(s_1, s_2)a_1a_2 + \phi(d, d) \quad (20)$$

The ϕ's are spatial correlations; thus, $\phi(s_1, s_2) = \overline{s_1, s_2}$, etc. It may be noted that $\phi(s_j, s_j) = \overline{s_j s_j} = 1$. 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 sample size, i.e., by the number of patterns seen in the searching process.

The method of searching which has proven most useful is the method of steepest descent. Vector adjustment 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 adjustments). It can be shown that the method of steepest descent will be stable when the proportionality constant k between the partial derivative and size

of 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 $1/2\,k$.

The method of adaption that has been used requires an extremely small sample size per iteration cycle, that is, 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.

The square of the error for a single pattern (the mean square error for a simple size of one) is given by equation 19. The partial derivatives can be listed as follows:

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

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

$$\frac{\partial \epsilon^2(k)}{\partial a_2} = s_2(k)[-2d(k)+2a_0+2s_1(k)a_1+ 2s_2(k)a_2] \quad (21)$$

Comparison of equation 21 with equation 18 shows that the derivatives are simply related to the analog error, and suggests that the derivative could be measured without squaring and averaging and without actual differentiation. The jth partial derivative is given by the following equation:

$$\frac{\partial \epsilon^2(k)}{\partial a_j} = -2s_j(k)\epsilon(k) \quad (22)$$

It follows that all partial derivatives have the same magnitude, and that their signs are determined by the error sign and the respective input signal signs. The procedure described for bringing $\epsilon(k)$ to zero with each successive input pattern gives the constant k a value of $1/2(n+1)$. It can be seen from the previous discussion 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 roughly 17 patterns (this is verified by the learning curve of Fig. 9). The search procedure could be readily modified to speed up or slow down the adaption process by adjusting k.

The misadjustment equation 17, when applied to the adaptive element, gives the per-unit increase in analog mean square error as a result of adapting on a finite number of patterns. Since the ratio of probability of neuron error to the mean square error $\overline{\epsilon^2}$ is essentially constant over a wide range of error probabilities (Fig. 10), the misadjustment may be interpreted as the ratio of the increase in error probability to the minimum error probability.

If adaption is accomplished by injection of a fresh pattern each iteration cycle, the misadjustment, as derived from equation 15, is

$$M=\frac{(n+1)}{2} \quad (23)$$

When the procedure of bringing $\epsilon(k)$ to zero each iteration cycle is followed, the misadjustment is

$$M=\frac{(n+1)}{2}=\frac{(n+1)}{2(n+1)}=\frac{1}{2} \quad (24)$$

If adaption is accomplished by taking a fixed collection of N patterns and repeating them over and over for several time constants (where the time constant is long, i.e. several times N), the misadjustment can be shown to be

$$M=\frac{(n+1)}{N} \quad (25)$$

Simulation tests have shown that the misadjustment equations, though approximate, apply to a very wide range of pattern and noise characteristics. A description is given in reference 9 of a typical experiment and its results.

The adaptive classifier can adapt after seeing remarkably few patterns. A misadjustment of 20% is acceptable in many applications. To achieve this, all one must do is supply the adaptive classifier with a number of patterns equal to five times the number of input lines, regardless of how noisy the patterns are and how difficult the "pure" patterns are to separate. The following rule of thumb applies to adaptive threshold elements: the number of patterns required to train an adaptive element is equal to several times the number of bits per pattern.

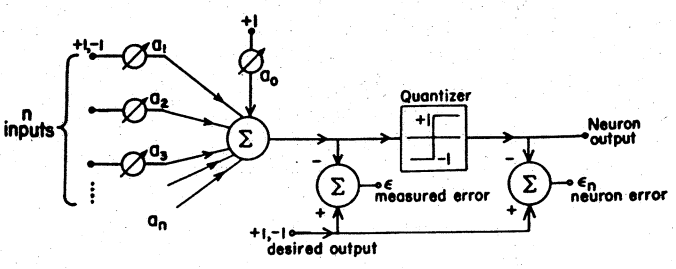
Madaline I, a Parallel Network of Adalines

Linearly separable pure patterns and noisy versions of the patterns are readily classified by the single element. Although nonlinearly separable pure patterns and their noisy equivalents also can be separated by a single element, absolute performance can be improved and the generality of the classification scheme can be increased greatly if more than one element is used.

Two Adalines were combined by using the following adaption procedure: If the desired output for a given input pattern applied to both elements was -1, then both elements were adapted in the usual manner to ensure this. If the desired output was $+1$, the element with the smallest analog error was assigned to adapt to give a $+1$ output, while the other element remained unchanged. If either or both elements gave outputs of $+1$, the pattern was classified as $+1$. If both elements gave -1 outputs, the pattern was classified as -1.

This procedure assigns specific responsibility to the element that can most easily assume it. If, at the beginning of adaption, a given element 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. It is not necessary for the teacher to assign responsibility. The combination does this automatically and requires only input patterns and the associated desired outputs, as with the single element.

Another combination of Adalines has been organized in the following fashion: The inputs of an odd number of Adalines, five, for example, are connected in parallel and their outputs are connected to a majority device. If the desired response is $+1$, a majority, at least three out of five,



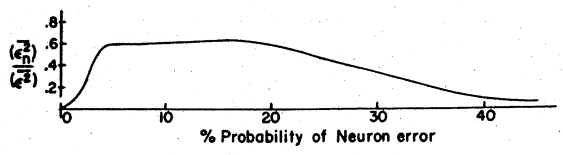


Fig. 10. Adaptive-element learning curve

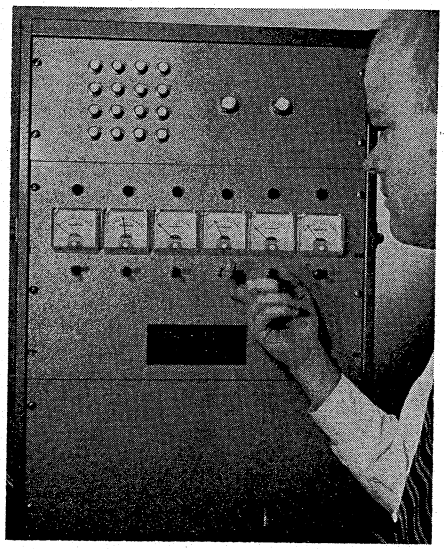


Fig. 11. Madaline I

must give the +1 response. If the majority response is −1, then the element whose sum (confidence level) is closest to zero is adapted to the +1 state. If the majority is now positive, the adaption is complete. If, on the other hand, another element must be adapted to make the majority positive, then the next element whose confidence level is closest to zero is adapted, etc. These parallel combinations of elements whose outputs are connected to fixed (nonadaptive) logical structures have been called Madalines (many Adalines). W. C. Ridgway III[10] has succeeded in proving that the adaption procedures described are convergent, and that if a given problem is solvable by such systems, then the given problem is guaranteed to be solved.

Memory Capacities of Adalines and Madalines

When experiments are first performed with an Adaline machine and it is trained to respond properly to given input patterns, the number of patterns that can be trained into the device should be determined. J. S. Koford has discovered by digital simulation that the number of random patterns (with their random desired responses) selected in a row that can be trained into an Adaline is equal to twice the number of adaptive weights. Moreover, when the same type of experiment is made with a Madaline I structure using either the "or" or the "majority" output logic element, the number of patterns that can be absorbed equals the number that can be absorbed per element multiplied by the number of elements. In other words, the capacity of a Madaline, like the capacity of an Adaline, is a number of patterns equal to twice the number of adaptive weights. It seems certain that the memory capacity per adaptive weight will turn out to be much higher in multilayered structures containing more than one adaptive layer.

Realization of Adaptive Circuits with Chemical Memistors

The structure of the Adaline neuron and its adaptation procedure is simple enough to allow an electronic fully automatic element to be developed. To have such an adaptive element, it is necessary to be able to store the gain values (analog quantities which can be positive or negative) in such a manner that these values could be changed electronically.

A new circuit element called the memistor (a resistor with memory)[11] has been devised by the author and M. E. Hoff for the realization of automatically adapted Adalines. A memistor provides a single variable gain factor. Each element, therefore, employs a number of memistors equal to the number of variable weights.

The memistor consists of a conductive substrate with insulated connection 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; they typically vary in resistance from 100 ohms to 1 ohm, accomplishing this in about 10 seconds with several milliamperes of plating current. Adaptation is achieved by direct current; the logical structure is sensed nondestructively by passing alternating currents through the array of memistor cells.

None of the element values or memistor characteristics is critical, because performance feedback in the adaptation process automatically finds the best weights in any event. These elements, have been built and have adapted, even with some defective and partially incorrectly wired memistor circuits.

The first working memistors were made of ordinary pencil leads immersed in test tubes containing copper-sulphate–sulphuric-acid plating baths. By using different baths, plating metals, geometries, and substrate materials, improvements have been made in lifetime and in electrical characteristics such as stability, relaxation, smoothness, and speed of adaptation. Memistor cells have already proven to be reliable over more than a year of service and are now commercially available, mounted in *TO-9* transistor cans. It is expected that memistors and other components that will appear in the future will have a substantial effect in making possible inexpensive, simple, and reliable systems, of both control and logical types.

Fig. 11 shows a machine containing six Adalines using a total of 102 memistors. This machine, Madaline I, has given excellent performance for the past year. When first constructed, very complex problems, such as 50 arbitrary 4×4 patterns with their desired responses, were able to be trained in, in spite of the fact that 25% of the weights were not adapting properly. This was the first produc-

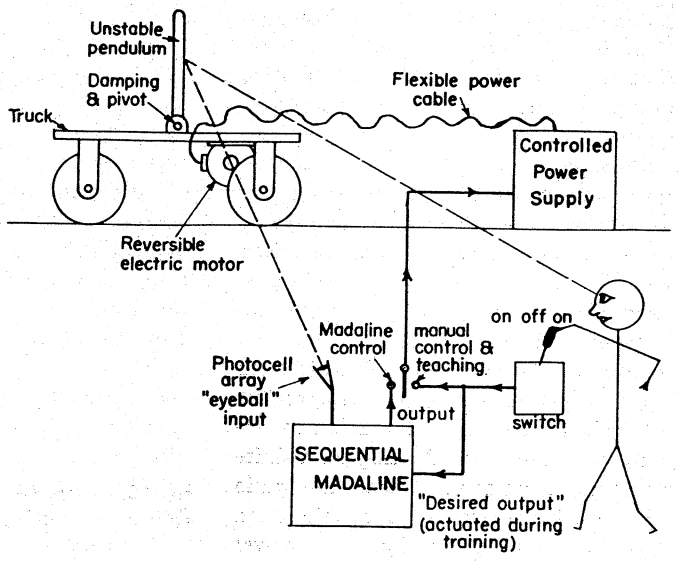


Fig. 12. One-dimensional broom-balancing machine

tion of memistors, and they were connected in without having been tested. Some were defective in construction, others were connected improperly, and others were victims of cold solder joints. This type of system, however, has the ability to adapt around its own defects.

The Adalines in Madaline I are independently adapted under manual control. This allows experimentation with the adaption process; provision is made for connecting the Adaline outputs to either the "or" or "majority" output elements. The digital input signals can come either from toggle switches or from a 4×4 array of photocells that comprises the retina for an artificial eyeball. Optical images can be presented directly to Madaline.

Conclusions

This paper has shown how performance feedback can be used to achieve automatic self-optimization of control systems, and how the same principles can be used to adapt logic structures. The Adaline element is essentially the same as an adaptive sampled-data system with quantized input and output signals.

The adaptive logic systems described in this paper originated from elementary adaptive control systems. As adaptive pattern-recognizing systems, they may now be used in control systems that can be taught a variety of fairly sophisticated control functions. An example is illustrated in Fig. 12.

The objective of the arrangement in this figure is first to have the man learn to balance the unstable pendulum, and then to have him teach a sequential Madaline to do the same thing. As the man stabilizes the pendulum, Madaline "sees" the pendulum (which is illuminated) and, at the same time, senses how the man reacts. After training, Madaline will be able to take over and stabilize the broom handle by providing appropriate switching signals which are reactions to the sequential patterns seen by its artificial eyeball.

At the present time, the broom balancer is equipped with ordinary sensors that provide four analog state-variable signals: angle, angle rate, velocity, and position of the cart supporting the pendulum. These signals are quantized and encoded by means of linearly independent codes described by Smith.[12] In this system, a conventional fourth-order bang-bang controller performs the control function while a single Adaline element learns to imitate the controller; after several minutes of training, the Adaline controller can take over and control the broom balancer. Smith shows how Adalines and simple Madaline networks can be coupled to properly encoded sources of state-variable signals to provide optimal switching surfaces for bang-bang controllers. Future research will be concerned with using more complex input signals, such as the optical ones described, and with learning without the teacher, i.e., self-adaptation rather than adaptation by imitation.

It is expected that pattern-recognizing control systems will be extremely flexible, and that they will make possible economical and reliable automation and control of highly complex processes, ultimately including processes whose complexities defy detailed mathematical description and analysis.

References

1. Sampled-Data Control Systems (book), **F. R. Ragazzini, G. F. Franklin.** McGraw-Hill Book Company, Inc., New York, N. Y., 1958.

2. Feedback Theory—Some Properties of Signal Flow Graphs, **S. J. Mason.** *Technical Report 153*, Research Laboratory of Electronics, Massachusetts Institute of Technology, Cambridge, Mass., Feb. 2, 1953.

3. Feedback Theory—Further Properties of Signal Flow Graphs, **S. J. Mason.** *Technical Report 303*, Research Laboratory of Electronics, Massachusetts Institute of Technology, July 20, 1955.

4. Extrapolation, Interpolation, and Smoothing of Stationary Time Series with Engineering Applications (book), **N. Wiener.** John Wiley and Sons, Inc., New York, N. Y., 1949.

5. Adaptive Sampled-Data Systems—A Statistical Theory of Adaptation, **B. Widrow.** *WESCON Convention Record*, Institute of Radio Engineers, 1959, pt. 4, pp. 74–86.

6. A Universal Non-Linear Filter, Predictor, and Simulator Which Optimizes Itself by a Learning Process, **D. Gabor, W. P. L. Wiley, R. Woodcock.** *Proceedings*, Institution of Electrical Engineers, London, England, vol. 108B, July 1960, pp. 422–35.

7. Unate Truth Functions, **R. McNaughton.** *Technical Report No. 4*, Applied Mathematics and Statistics Laboratory, Stanford University, Stanford, Calif., Oct. 21, 1957.

8. A Self-Organizing Binary System, **R. L. Mattson.** *Proceedings*, Eastern Joint Computer Conference, Institute of Radio Engineers, 1959, pp. 212–18.

9. Adaptive Switching Circuits, **B. Widrow, M. E. Hoff.** *WESCON Convention Record*, Institute of Radio Engineers, 1960, pt. 4, pp. 96–105.

10. An Adaptive Logic System with Generalizing Properties, **W. C. Ridgway III.** *Report SEL-62-040 (Tr No. 1556–1)*, Stanford Electronics Laboratories, Stanford, Calif., Apr. 1962.

11. An Adaptive ADALINE Neuron Using Chemical Memistors, **B. Widrow.** *ERL Technical Report No. 1553-2* Electronics Research Laboratory, Stanford University, Oct. 17, 1960.

12. Contactor Control by Adaptive Pattern-Recognition Techniques, **F. W. Smith.** *Technical Report No. 6762-1*, Electronics Research Laboratory, Stanford University, Apr. 1964.

A reprint from **APPLICATIONS AND INDUSTRY**, published by
The Institute of Electrical and Electronics Engineers, Inc.

Printed in the United States of America

September 1964 issue