Stationary and Nonstationary Learning Characteristics of the LMS Adaptive Filter

BERNARD WIDROW, FELLOW, IEEE, JOHN M. McCOOL, SENIOR MEMBER, IEEE,
MICHAEL G. LARIMORE, STUDENT MEMBER, IEEE, C. RICHARD JOHNSON, JR., STUDENT MEMBER, IEEE

Abstract—This paper describes the performance characteristics of the LMS adaptive filter, a digital filter composed of a tapped delay line and adjustable weights, whose impulse response is controlled by an adaptive algorithm. For stationary stochastic inputs, the mean-square error, the difference between the filter output and an externally supplied input called the "desired response," is a quadratic function of the weights, a paraboloid with a single fixed minimum point that can be sought by gradient techniques. The gradient estimation process is shown to introduce noise into the weight vector that is proportional to the speed of adaptation and number of weights. The effect of this noise is expressed in terms of a dimensionless quantity "misadjustment" that is a measure of the deviation from optimal Wiener performance. Analysis of a simple nonstationary case, in which the minimum point of the error surface is moving according to an assumed first-order Markov process, shows that an additional contribution to misadjustment arises from "lag" of the adaptive process in tracking the moving minimum point. This contribution, which is additive, is proportional to the number of weights but inversely proportional to the speed of adaptation. The sum of the misadjustments can be minimized by choosing the speed of adaptation to make equal the two contributions. It is further shown, in Appendix A, that for stationary inputs the LMS adaptive algorithm, based on the method of steepest descent, approaches the theoretical limit of efficiency in terms of misadjustment and speed of adaptation when the eigenvalues of the input correlation matrix are equal or close in value. When the eigenvalues are highly disparate ($\lambda_{max}/\lambda_{min} > 10$), an algorithm similar to LMS but based on Newton's method would approach this theoretical limit very closely.

I. Introduction

OUR PURPOSE IS to derive relationships between speed of adaptation and performance of adaptive systems. In general, faster adaptation leads to more noisy adaptive processes. When the input environment of an adaptive system is statistically stationary, best steady-state performance results from slow adaptation. However, when the input statistics are time variable, best performance is obtained by a compromise between fast adaptation (necessary to track variations in input statistics) and slow adaptation (necessary to contain the noise in the adaptive process). These issues will be studied both analytically and by computer simulation. The context of this study will be restricted to adaptive digital filters "driven" by the LMS adaptation algorithm of Widrow and Hoff [1], [2]. This algorithm and similar algorithms have been used for many years in a wide variety of practical applications [3]–[26].

We are attempting to formulate a "statistical theory of adaptation." This is a very difficult subject and the present work should be regarded as only a beginning. Stability and rate of convergence are analyzed first, then gradient noise and its effects upon performance are assessed. The concept of "misadjustment" is defined and used to establish design criteria for an adaptive predictor. Extension of the concept to the analysis of a useful but relatively simple form of nonstationary adaptation leads to criteria governing optimal choice of speed of adaptation.

The results reported here have been gradually developed in our laboratory during the past 15 years and are being extended and applied by ongoing research.

II. An Adaptive Filter

The filter considered here comprises a tapped delay line, variable weights (variable gains) whose input signals are the signals at the delay-line taps, a summer to add the weighted signals, and an adaptation process that automatically seeks an optimal impulse response by adjusting the weights. Fig. 1 illustrates the adaptive filter as used in modeling an unknown dynamic system.

In addition to the usual input signals, another input signal, the "desired response," must be supplied to the adaptive filter during the adaptation process. In Fig. 1, essentially the same input is applied to the adaptive filter as to the unknown system to be modeled. The output of this system provides the desired response for the adaptive filter. In other applications, considerable ingenuity may be required to obtain a suitable desired response for an adaptive process.

III. The Performance Surface

The analysis of the adaptive filter is developed by considering the "adaptive linear combiner" of Fig. 2, a subsystem of the adaptive filter of Fig. 1, comprising its most significant part.[1]

In Fig. 2, a set of input signals is weighted and summed to form an output signal. The inputs occur simultaneously and discretely in time. The jth input vector is

$$X_j = [x_{1j}, x_{2j}, \cdots, x_{lj}, \cdots, x_{nj}]^T.$$

The set of weights is designated by the vector $W^T = [w_1, w_2, \cdots, w_l, \cdots, w_n]$. The jth output signal is

$$y_j = \sum_{l=1}^{n} w_l x_{lj} = W^T X_j = X_j^T W. \quad (1)$$

The input signals and desired response are assumed to be stationary ergodic processes. Denoting the desired response as

Manuscript received September 29, 1975; revised March 9, 1976. This work was supported in part by the National Science Foundation under Grant ENGR 74-21752.

B. Widrow, M. Larimore, and C. R. Johnson are with the Information Systems Laboratory, Department of Electrical Engineering, Stanford University, Stanford, CA 94305.

J. M. McCool is with the Fleet Engineering Department, Naval Undersea Center, San Diego, CA 92132.

[1]This combinational system can be connected to the elements of a phased array antenna to make an adaptive antenna [5]–[9], or to a quantizer to form an adaptive threshold element ("Adaline" [1], [3] or TLU [2]) for use in adaptive logic and pattern-recognition systems. It can also be used as the adaptive portion of certain learning control systems [10], [11]; as a key portion of adaptive filters for channel equalization [12]–[16]; for adaptive noise cancelling [17], [18]; or for adaptive systems identification [19]–[26].

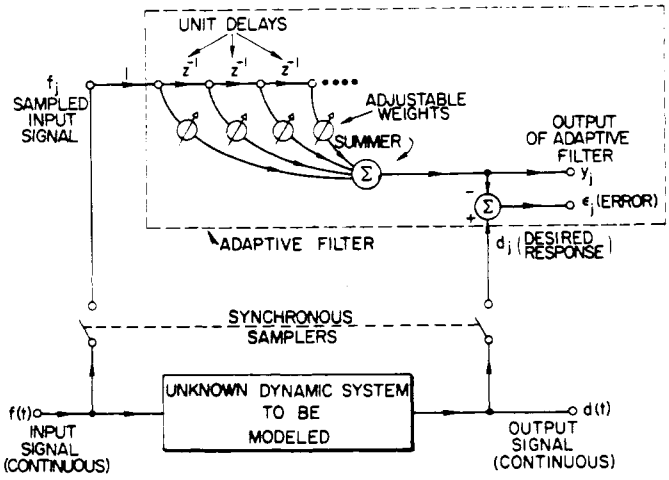

Fig. 1. Modeling an unknown system by a discrete adaptive filter.

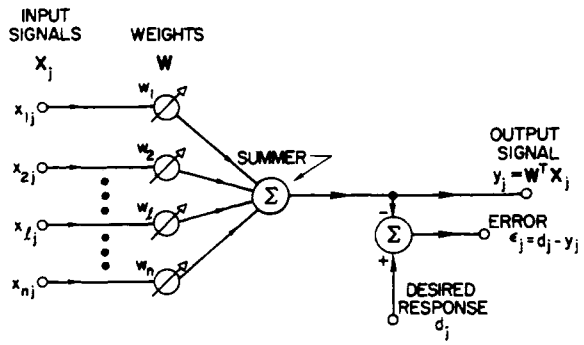

Fig. 2. Adaptive linear combiner.

d_j, the error at the jth time is

$$\epsilon_j = d_j - y_j = d_j - W^T X_j = d_j - X_j^T W. \tag{2}$$

The square of this error is

$$\epsilon_j^2 = d_j^2 - 2 d_j X_j^T W + W^T X_j X_j^T W. \tag{3}$$

The mean-square error ξ, the expected value of ϵ_j^2, is

$$\begin{aligned} \xi \triangleq E[\epsilon_j^2] &= E[d_j^2] - 2E[d_j X_j^T] W + W^T E[X_j X_j^T] W \\ &= E[d_j^2] - 2P^T W + W^T R W \end{aligned} \tag{4}$$

where the cross correlation vector between the input signals and the desired response is defined as

$$E[d_j X_j] = E\begin{bmatrix} d_j x_{1j} \\ d_j x_{2j} \\ \cdot \\ \cdot \\ \cdot \\ d_j x_{nj} \end{bmatrix} \triangleq P \tag{5}$$

and where the symmetric and positive definite input correlation matrix R of the x-input signals is defined as

$$E[X_j X_j^T] = E\begin{bmatrix} x_{1j}x_{1j} & x_{1j}x_{2j} & \cdots \\ x_{2j}x_{1j} & x_{2j}x_{2j} & \cdots \\ \cdot & \cdot & \\ \cdot & \cdot & \\ \cdot & \cdot & \\ & \cdots & x_{nj}x_{nj} \end{bmatrix} \triangleq R. \tag{6}$$

It may be observed from (4) that the mean-square-error (mse) performance function is a quadratic function of the weights, a "bowl-shaped" surface; the adaptive process will be continuously adjusting the weights, seeking the bottom of the bowl. This may be accomplished by steepest descent methods [27], [28] discussed below.

In the nonstationary case, the adaptive process must track the bottom of the bowl, which may be moving. An analysis of a simple nonstationary case is presented in Section XI.

IV. The Gradient and the Wiener Solution

The method of steepest descent uses gradients of the performance surface in seeking its minimum. The gradient at any point on the performance surface may be obtained by differentiating the mse function, equation (4), with respect to the weight vector. The gradient vector is

$$\nabla = -2P + 2RW. \tag{7}$$

Set the gradient to zero to find the optimal weight vector W^*:

$$W^* = R^{-1} P \tag{8}$$

which is the Wiener–Hopf equation in matrix form.

The minimum mse is obtained from (8) and (4):

$$\xi_{\min} = E[d_j^2] - P^T W^*. \tag{9}$$

Substituting (9) into (4) yields a useful formula for mse:

$$\xi = \xi_{\min} + (W - W^*)^T R (W - W^*). \tag{10}$$

Define V as the difference between W and the Wiener solution W^*:

$$V \triangleq (W - W^*). \tag{11}$$

Therefore,

$$\xi = \xi_{\min} + V^T R V. \tag{12}$$

Differentiation of (12) yields another form for the gradient:

$$\nabla = 2RV. \tag{13}$$

The input correlation matrix, being symmetric and positive definite, may be represented as

$$R = Q \Lambda Q^{-1} = Q \Lambda Q^T \tag{14}$$

where Q is the orthonormal modal matrix of R and Λ is its diagonal matrix of eigenvalues:

$$\Lambda = \text{diag}\,[\lambda_1, \lambda_2, \cdots, \lambda_p, \cdots, \lambda_n]. \tag{15}$$

Equation (12) may be reexpressed as

$$\xi = \xi_{\min} + V^T Q \Lambda Q^{-1} V. \tag{16}$$

Define a transformed version of V as

$$V' \triangleq Q^{-1} V \quad \text{and} \quad V = QV'. \tag{17}$$

Accordingly, equation (12) may be put in normal form as

$$\xi = \xi_{\min} + V'^T \Lambda V'. \tag{18}$$

The primed coordinates are therefore the principal axes of the quadratic surface. Transformation (17) may be applied to the weight vector itself,

$$W' = Q^{-1} W \quad \text{and} \quad W = QW'. \tag{19}$$

V. The Method of Steepest Descent

The method of steepest descent makes each change in the weight vector proportional to the negative of the gradient vector:

$$W_{j+1} = W_j + \mu(-\nabla_j). \tag{20}$$

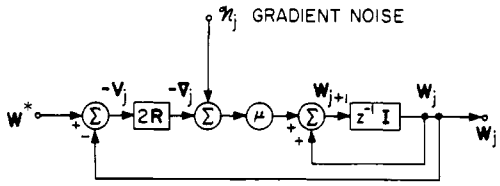

Fig. 3. Feedback model of steepest descent.

The scalar parameter μ is a convergence factor that controls stability and rate of adaptation. The gradient at the jth iteration is ∇_j. Using (13), (14), and (17), equation (20) becomes

$$V'_{j+1} - (I - 2\mu\Lambda)V'_j = 0. \tag{21}$$

This homogeneous vector difference equation is uncoupled. It has a simple geometric solution in the primed coordinates [5]:

$$V'_j = (I - 2\mu\Lambda)^j V_0 \tag{22}$$

where V'_0 is an initial condition:

$$V'_0 = W'_0 - W^{*\prime}. \tag{23}$$

For convergence, it is necessary that

$$1/\lambda_{\max} > \mu > 0 \tag{24}$$

where $\lambda_{\max}$ is the largest eigenvalue of R. From (22), we see that transients in the primed coordinates will be geometric; the geometric ratio of the pth coordinate is

$$r_p = (1 - 2\mu\lambda_p). \tag{25}$$

An exponential envelope can be fitted to a geometric sequence. If the basic unit of time is considered to be the iteration cycle, time constant τ_p can be determined as follows:

$$r_p = \exp\left(-\frac{1}{\tau_p}\right) = 1 - \frac{1}{\tau_p} + \frac{1}{2!\tau_p^2} - \cdots. \tag{26}$$

The case of general interest is slow adaptation; i.e., large τ_p. Therefore,

$$r_p = (1 - 2\mu\lambda_p) \simeq 1 - \frac{1}{\tau_p}$$

or

$$\tau_p \simeq \frac{1}{2\mu\lambda_p}. \tag{27}$$

Equation (27) gives the time constant of the pth mode.

Steepest descent can be regarded as a feedback process where the gradient plays the role of vector error signal. The process, if stable, tends to bring the gradient to zero.[2] Fig. 3 shows a feedback model for a stationary quadratic mse surface being searched by the method of steepest descent. The model is equivalent to the following set of relations.

$$\begin{aligned} W_j &= W_{j+1}\,|_{\text{delayed one iteration}} \\ W_{j+1} &= W_j + \mu(-\nabla_j) \\ \nabla_j &= 2R(W_j - W^*) = 2RV. \end{aligned} \tag{28}$$

This feedback model is used subsequently in a study of nonstationary adaptation. Notice an input not mentioned earlier, "gradient noise." Because gradients are estimated at each iteration cycle with finite amounts of input data, they will be imperfect or noisy.

VI. The LMS Algorithm

The LMS algorithm is an implementation of steepest descent using measured or estimated gradients:

$$W_{j+1} = W_j + \mu(-\hat{\nabla}_j). \tag{29}$$

The estimate of the true gradient is $\hat{\nabla}$.

The gradient estimate used by LMS takes the gradient of the square of a single error sample. Thus

$$\hat{\nabla}_j = -2\epsilon_j X_j. \tag{30}$$

The LMS algorithm can be written as

$$W_{j+1} = W_j + 2\mu\epsilon_j X_j. \tag{31}$$

If we assume that X_j is uncorrelated over time (i.e., that $E[X_j X_{j+l}^T] = 0$, $\forall l \neq 0$), an assumption common in the field of stochastic approximation [30], [31], then the expected value of the gradient estimate equals the true gradient, and the weight-vector mean is convergent to the Wiener solution of (8), as shown in [4] and [5].

Condition (24) is necessary and sufficient for convergence of the LMS algorithm. However, in practice, the individual eigenvalues are rarely known so that (24) is not always easy to apply. Since $\operatorname{tr} R$ is the total input power to the weights, a generally known quantity, and since $\operatorname{tr} R > \lambda_{\max}$ as R is positive definite, a sufficient condition for convergence is

$$1/\operatorname{tr} R > \mu > 0. \tag{32}$$

VII. The Learning Curve and its Time Constants

During adaptation, the error ϵ_j is nonstationary as the weight vector adapts toward W^*. The mse can be defined only on the basis of ensemble averages. From (18), we obtain

$$\xi_j = \xi_{\min} + V_j'^T \Lambda V'_j. \tag{33}$$

Imagine an ensemble of adaptive processes, each having individual stationary ergodic inputs drawn from the same statistical population, with all initial weight vectors equal. The mse ξ_j is a function of iteration number j, obtained by averaging over the ensemble at iteration j.

Using (22), but assuming no noise in the weight vector, equation (33) becomes

$$\begin{aligned} \xi_j &= \xi_{\min} + V_0'^T \Lambda (I - 2\mu\Lambda)^{2j} V'_0 \\ &= \xi_{\min} + V_0^T (I - 2\mu R)^j R (I - 2\mu R)^j V_0. \end{aligned} \tag{34}$$

When the adaptive process is convergent, it is clear from (34) that

$$\lim_{j\to\infty} \xi_j = \xi_{\min}$$

and that the geometric decay in ξ_j going from ξ_0 to $\xi_{\min}$ will, for the pth mode, have a geometric ratio of r_p^2 and a time constant

$$\tau_{p_{\text{mse}}} \triangleq \frac{1}{2}\tau_p = \frac{1}{4\mu\lambda_p}. \tag{35}$$

The result obtained by plotting mse against number of iterations is called the "learning curve." Due to noise in the weight vector, actual practice will show ξ_j to be higher than indicated by (34).

[2] This has been called performance feedback [1], [29].

VIII. Gradient and Weight-Vector Noise

Gradient noise will affect the adaptive process both during initial transients and in steady state. The latter condition is of particular interest here.

Assume that the weight vector is close to the Wiener solution. Assume, as before, that X_j and d_j are stationary and ergodic and that X_j is uncorrelated over time; i.e.,

$$E[X_j X_{j+k}] = 0, \qquad k \neq 0. \tag{36}$$

The LMS algorithm uses a gradient estimate

$$\hat{\nabla} = -2\epsilon_j X_j = \nabla_j + N_j \tag{37}$$

where ∇_j is the true gradient and N_j is a zero-mean gradient estimation noise vector. When $W_j = W^*$, the true gradient is zero, but the gradient would be estimated according to (30) and is equal to the gradient noise:

$$N_j = -2\epsilon_j X_j. \tag{38}$$

According to Wiener filter theory, when $W_j = W^*$, ϵ_j and X_j are uncorrelated. If they are assumed zero-mean Gaussian, ϵ_j and X_j are statistically independent. As such, the covariance of N_j is

$$\begin{aligned} \text{cov}\,[N_j] &= E[N_j N_j^T] = 4E[\epsilon_j^2 X_j X_j^T] \\ &= 4E[\epsilon_j^2]\,E[X_j X_j^T] \\ &= 4E[\epsilon_j^2]R. \end{aligned} \tag{39}$$

When $W_j = W^*$, $E[\epsilon_j^2] = \xi_{\min}$. Accordingly,

$$\text{cov}\,[N_j] = 4\xi_{\min} R. \tag{40}$$

As long as $W_j \simeq W^*$, we assume that the gradient noise covariance is given by (40) and that this noise is stationary and uncorrelated over time. The latter assumption is based on (36) and (38).

Projecting the gradient noise,

$$N_j' = Q^{-1} N_j \tag{41}$$

its covariance becomes

$$\begin{aligned} \text{cov}\,[N_j'] = E[N_j' N_j'^T] = E[Q^{-1} N_j N_j^T Q] &= Q^{-1}\ \text{cov}\,[N_j]\,Q \\ &= 4\xi_{\min} Q^{-1} R Q \\ &= 4\xi_{\min} \Lambda. \end{aligned} \tag{42}$$

Although the components of N_j are correlated with each other, those of N_j' are mutually uncorrelated and can, therefore, be handled more easily.

Gradient noise propagates and causes noise in the weight vector. Accounting for gradient noise, the LMS algorithm can be expressed as

$$W_{j+1}' = W_j' + \mu(-\hat{\nabla}_j') = W_j' + \mu(-\nabla_j' + N_j'). \tag{43}$$

This equation can be written in terms of V_j' as

$$V_{j+1}' = V_j' + \mu(-2\Lambda V_j' + N_j'). \tag{44}$$

Near the minimum point of the error surface in steady-state, the mean of V_j' is zero and the covariance of the weight-vector noise is [18, appendix D, section B]

$$\text{cov}\,[V_j'] = \mu\xi_{\min} I \tag{45}$$

where the components of the weight-vector noise are of equal variance and are mutually uncorrelated. It has been found, however, that (45) closely approximates measured weight-vector covariances under a considerably wider range of conditions than the assumptions above imply.

IX. Misadjustment due to Gradient Noise

Random noise in the weight vector causes an excess mse. If the weight vector were noise free and converged such that $W_j = W^*$, then the mse would be $\xi_{\min}$. However, this does not occur in actual practice so that the weight vector is on the average "misadjusted" from its optimal setting.

An expression for mse in terms of V_j' is given by (33), from which we obtain an expression for excess mse:

$$(\text{excess mse}) = V_j'^T \Lambda V_j'. \tag{46}$$

The average excess mse is an important quantity:

$$E[V_j'^T \Lambda V_j'] = \sum_{p=1}^{n} \lambda_p E[(v_{pj}')^2] \tag{47}$$

where v_{pj}' is the pth component of V_j'. After adaptive transients die out, $E[V_j'] = 0$. Therefore, from (45) we have

$$E[(v_{pj}')^2] = \mu\xi_{\min}, \forall p. \tag{48}$$

Substitution into (47) yields the average excess mse,

$$E[V_j'^T \Lambda V_j'] = \mu\xi_{\min} \sum_{p=1}^{n} \lambda_p = \mu\xi_{\min}\ \text{tr}\,R. \tag{49}$$

We define the "misadjustment" due to gradient noise as the dimensionless ratio of the average excess mse to the minimum mse,

$$M \triangleq \frac{\text{average excess mse}}{\xi_{\min}}. \tag{50}$$

For the LMS algorithm, under the conditions assumed above,

$$M = \mu\ \text{tr}\,R. \tag{51}$$

This formula works well for small values of misadjustment, 25 percent or less, so that the assumption

$$W_j \simeq W^* \tag{52}$$

is satisfied. The misadjustment is a useful measure of the cost of adaptability. A value of $M = 10$ percent means that the adaptive system has a mse only 10 percent greater than $\xi_{\min}$.

It is useful to relate misadjustment to the speed of adaptation and the number of weights being adapted. Since tr R equals the sum of the eigenvalues,

$$M = \mu \sum_{p=1}^{n} \lambda_p = \mu n \lambda_{\text{ave}} \tag{53}$$

where λ_{ave} is the average of the eigenvalues. From (35),

$$\lambda_p = \frac{1}{4\mu}\left(\frac{1}{\tau_{p_{\text{mse}}}}\right) \quad \text{or} \quad \lambda_{\text{ave}} = \frac{1}{4\mu}\left(\frac{1}{\tau_{p_{\text{mse}}}}\right)_{\text{ave}}. \tag{54}$$

Substituting into (53) yields

$$M = \frac{n}{4}\left(\frac{1}{\tau_{p_{\text{mse}}}}\right)_{\text{ave}}. \tag{55}$$

The special case where all eigenvalues are equal is an important one. The learning curve has only one time constant τ_{mse}, and the misadjustment is given by

$$M = \frac{n}{4\tau_{mse}}. \tag{56}$$

When the eigenvalues are sufficiently similar for the learning curve to be approximately fitted by a single exponential, its time constant may be applied to (56) to give an approximate value of M.

Since transients settle in about four time constants, equation (56) leads to an approximate "rule of thumb:" the misadjustment equals the number of weights divided by the settling time. A 10-percent misadjustment would be satisfactory for many engineering designs. Operation with 10-percent misadjustment can generally be achieved with an adaptive settling time equal to ten times the memory time span of the adaptive transversal filter.

X. A Design Example/Choosing Number of Filter Weights for an Adaptive Predictor

Fig. 4 is a block diagram of an adaptive predictor.[3] Its adaptive filter converts the delayed input $x_{j-\Delta}$ into x_j as best possible. If the adaptive-filter weights are copied into an auxiliary filter having a tapped delay-line structure identical to that of the adaptive filter and the input x_j is applied to this auxiliary filter, the resulting output will be a linear least squares estimate of $x_{j+\Delta}$ (limited by finite filter length and misadjustment).

A computer implementation of the adaptive predictor was made using a simulated input signal x_j obtained by bandpass filtering a white Gaussian signal and adding this to another independent white Gaussian signal. Prediction was one time sample in the future, i.e., $\Delta = 1$, using an adaptive filter with five weights, all initially set to zero.

Fig. 5 depicts three learning curves. For each adaptive step, the mse ξ_j corresponding to the current weight vector W_j was calculated from (10) using known values of R and ξ_{min}, giving the "individual learning curve." The smooth "ensemble average learning curve" is simply the average of 200 such individual curves and approximates the adaptive behavior in the mean. The third curve calculated from (34) shows how the process would evolve if perfect knowledge of the gradient were available at each step. It is a noiseless "steepest descent learning curve."

Of particular interest is the residual difference between the ensemble learning curve and the steepest descent learning curve after convergence. The latter, of course, converges to ξ_{min}. The difference is the excess mse due to gradient noise, in this case, giving a measured misadjustment of 3 percent. The theoretical misadjustment was $M = 2.5$ percent. The minor discrepancy is due mainly to the fact that the input samples are highly correlated in violation of the assumption that $E[X_j X_{j+k}^T] = 0$, $\forall k \neq 0$, used in the derivation of misadjustment formula (56).

The ensemble average learning curve has an effective measured time constant τ_{mse} of about 50 iterations since it falls to within 2 percent of its converged value at around iteration 200.

[3]This same predictor was described by Widrow in [5]; it has been used for data compression and speech encoding [32] and for "maximum entropy" spectral estimation [33].

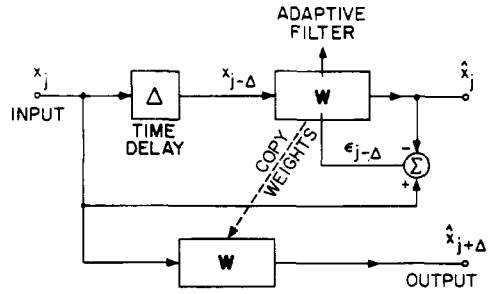

Fig. 4. An adaptive predictor.

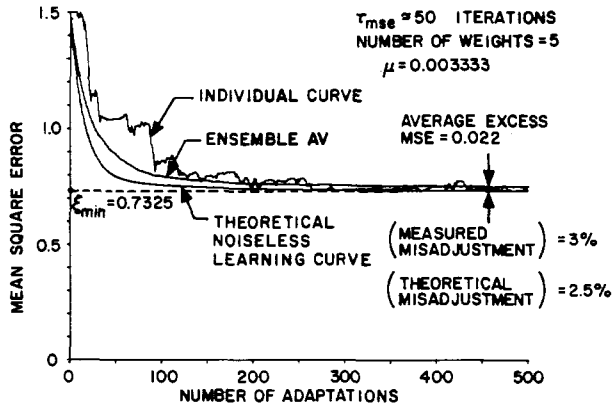

Fig. 5. Learning curves for adaptive predictor.

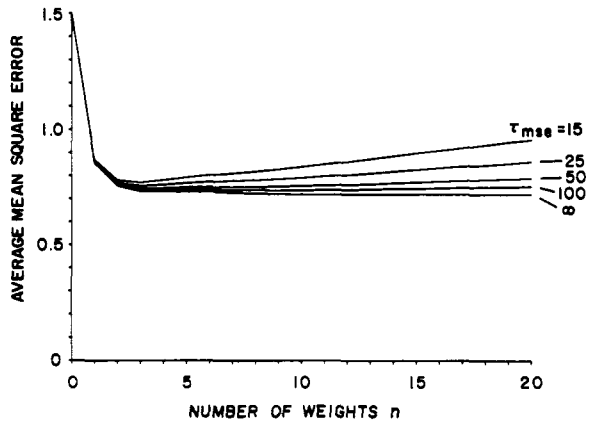

Fig. 6. Performance versus number of weights and adaptive predictor time constant.

When all eigenvalues are equal, equation (35) becomes

$$\tau_{mse} = \frac{1}{4\mu\lambda} = \frac{n}{4\mu \operatorname{tr} R}. \tag{57}$$

Using (57) in the present case (although the eigenvalues range over a 10 to 1 ratio) yields $\tau_{mse} = 50$, which agrees with experiment. Equation (57) gives a formula for an "effective time constant," useful even when the eigenvalues are highly disparate.

The performance of the adaptive filter may improve with an increase in the number of weights. However, for a fixed rate of convergence, larger numbers of weights increase misadjustment. Fig. 6 shows these conflicting effects. The lowest curve for $\tau_{mse} = \infty$ represents idealized noise-free adaptation, providing the minimum mse $\xi_{min}(n)$ for each value of n. The other curves include average excess mse due to gradient noise. We define the "average mse" to be the sum of the minimum mse

TABLE I
COMPARISON OF THEORETICAL AND EXPERIMENTAL ADAPTIVE PREDICTOR PERFORMANCE

NUMBER OF WEIGHTS n	APPROX TIME CONSTANT τ_{mse}	AVERAGE MSE Theoretical/Experimental		MISADJUSTMENT Theoretical/Experimental	
5	100	.742	.751	1.3%	2.5%
5	50	.751	.754	2.5%	3.0%
5	25	.769	.781	5.0%	6.6%
5	15	.794	.824	8.3%	12.6%
10	100	.737	.745	2.5%	3.5%
10	50	.755	.764	5.0%	6.2%

and the average excess mse. Thus

$$(\text{average mse}) = [1 + M]\xi_{\min}(n). \tag{58}$$

Using this formula, theoretical curves have been plotted in Fig. 6 for approximate values of τ_{mse} of 100, 50, 25, and 15 iterations. It is apparent from these curves that increasing the number of weights does not always guarantee improved system performance. Experimental points derived by computer simulation have compared very well with theoretical values predicted by (58). Typical results are summarized in Table I.

XI. Response of the LMS Adaptive Filter in a Nonstationary Environment

Filtering nonstationary signals is a major area of application for adaptive techniques, especially when the stochastic properties of the signals are unknown *a priori*. Although the utility of adaptive filters with nonstationary inputs has been demonstrated experimentally, very little of this work has been published, perhaps due to the inherently complex mathematics associated with such problems [34], [35]. The nonstationary situations to be studied here are highly simplified, but they retain the essence of the problem that is common to more complicated and realistic situations.

The example considered here involves modeling or identifying an unknown time-variable system by an adaptive LMS transversal filter. The unknown system is assumed to be a transversal filter of same length n whose weights (impulse response values) undergo independent stationary ergodic first-order Markov processes, as indicated in Fig. 7. The input signal x_j is assumed to be stationary and ergodic. Additive output noise, assumed to be stationary, of mean zero, and of variance $\xi_{\min}$, prevents a perfect match between the unknown system and the adaptive system. The minimum mse is, therefore, $\xi_{\min}$, achieved whenever the weights of the adaptive filter W_j match those of the unknown system. The latter are at every instant the optimal values for the corresponding weights of the adaptive filter and are designated W_j^*, the subscript indicating that the unknown "target" to be tracked is time variable.

According to the scheme of Fig. 7, minimizing mse causes the adaptive weight vector W_j to attempt to best match the unknown W_j^* on a continual basis. The R matrix, dependent only on the statistics of x_j, is constant even as W_j^* varies. The desired response of the adaptive filter d_j is nonstationary, being the output of a time-variable system. The minimum mse $\xi_{\min}$ is constant. Thus the mse function, a quadratic bowl, varies in position while its eigenvalues, eigenvectors, and $\xi_{\min}$ remain constant.

In order to study this form of nonstationary adaptation both analytically and by computer simulation, a model comprising an ensemble of nonstationary adaptive processes has been defined and constructed as illustrated in Fig. 8. The unknown

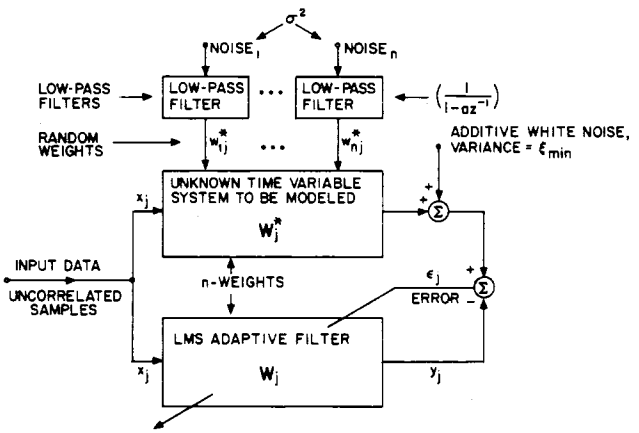

Fig. 7. Modeling an unknown time-variable system.

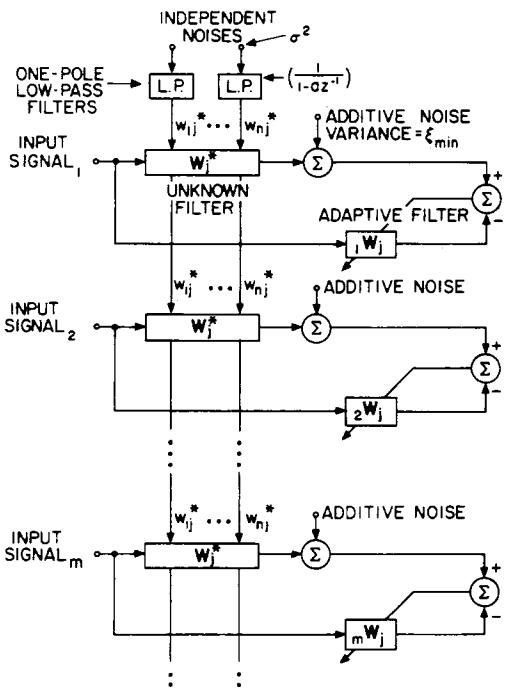

Fig. 8. An ensemble of nonstationary adaptive processes.

filters to be modeled are all identical and have the same time-variable weight vector W_j^* throughout the ensemble. Each ensemble member has its own independent input signal going to both the unknown system and the corresponding adaptive system. The effect of output noise in the unknown systems is obtained by the addition of independent noise of variance $\xi_{\min}$. All of the adaptive filters are assumed to start with the same initial weight vector W_0; each develops its own weight

vector over time in attempting to pursue the Markovian target W_j^*.

For a given adaptive filter, the weight-vector tracking error at the jth instant is $(W_j - W_j^*)$. This error is due to both the effects of gradient noise and weight-vector lag, and may be expressed as

$$\text{(Weight-vector error)}_j = (W_j - W_j^*) \equiv \underbrace{(W_j - E[W_j])}_{\substack{\text{weight-vector} \\ \text{noise}}} + \underbrace{(E[W_j] - W_j^*)}_{\substack{\text{weight-vector} \\ \text{lag}}}. \quad (59)$$

The expectations are averages over the ensemble. The components of error are identified in (59). Any difference between the ensemble mean of the adaptive weight vectors and the target value W_j^* is due to lag in the adaptive process, while the deviation of the individual adaptive weight vectors about the ensemble mean is due to gradient noise.

Weight-vector error causes an excess mse. The ensemble average excess mse at the jth instant is

$$\begin{pmatrix}\text{average excess} \\ \text{mse}\end{pmatrix}_j = E[(W_j - W_j^*)^T R (W_j - W_j^*)]. \quad (60)$$

Using (59), this can be expanded as follows:

$$\begin{aligned}\begin{pmatrix}\text{average excess} \\ \text{mse}\end{pmatrix}_j &= E[(W_j - E[W_j])^T R (W_j - E[W_j])] \\ &\quad + E[(E[W_j] - W_j^*)^T R (E[W_j] - W_j^*)] \\ &\quad + 2E[(W_j - E[W_j])^T R (E[W_j] - W_j^*)]. \quad (61)\end{aligned}$$

Expanding the last term of (61) and simplifying since W_j^* is constant over the ensemble,

$$\begin{aligned}&2E[W_j^T R E[W_j] - W_j^T R W_j^* - E[W_j]^T R E[W_j] + E[W_j]^T R W_j^*] \\ &= 2[E[W_j]^T R E[W_j] - E[W_j]^T R E[W_j] \\ &\quad - E[W_j]^T R W_j^* + E[W_j]^T R W_j^*] \\ &= 0. \quad (62)\end{aligned}$$

Therefore, (61) becomes

$$\begin{aligned}\begin{pmatrix}\text{average excess} \\ \text{mse}\end{pmatrix}_j &= E[(W_j - E[W_j])^T R (W_j - E[W_j])] \\ &\quad + E[(E[W_j] - W_j^*)^T R (E[W_j] - W_j^*)]. \quad (63)\end{aligned}$$

The average excess mse is thus a sum of components due to both gradient noise and lag:

$$\begin{aligned}\begin{pmatrix}\text{average excess} \\ \text{mse due to lag}\end{pmatrix}_j &= E[(E[W_j] - W_j^*)^T R (E[W_j] - W_j^*)] \\ &= E[(E[W_j'] - W_j^{*\prime})^T \Lambda (E[W_j] - W_j^{*\prime})] \quad (64)\end{aligned}$$

$$\begin{aligned}\begin{pmatrix}\text{average excess} \\ \text{mse due to} \\ \text{gradient noise}\end{pmatrix}_j &= E[(W_j - E[W_j])^T R (W_j - E[W_j])] \\ &= E[(W_j' - E[W_j'])^T \Lambda (W_j' - E[W_j'])]. \quad (65)\end{aligned}$$

Fig. 9 is a feedback diagram adapted from Fig. 3, illustrating the two sources of weight-vector error. From the feedback diagram, it can be seen that the "output" W_j attempts to track the time variable "input" W_j^*. Tracking error $(W_j - W_j^*)$ is caused by the propagation of gradient noise and by the response of the adaptive process to the random variations of W_j^*. It will be shown that increasing the time constant of the adaptive process diminishes the propagation of gradient noise but simultaneously increases the lag error that results from the random changes in W_j^*.

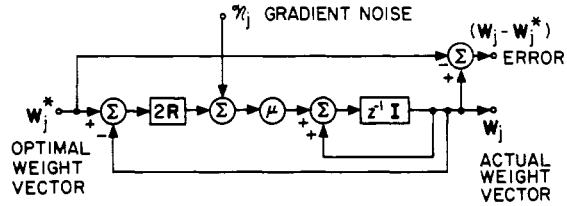

Fig. 9. Feedback diagram of steepest descent showing sources of weight tracking error.

The gradient-noise covariance for the stationary case (40) is a function of R. Since R is constant, equation (40) is a good representation of covariance for the type of nonstationarity under study. Furthermore, Fig. 9 shows that the propagation of gradient noise in the linear feedback system representing the adaptive process is not affected by variability of W_j^*. Therefore, equation (49) can be used to provide an evaluation of (65), the excess mse from gradient noise. The next step is an evaluation of (64), the excess mse due to lag. Statistical knowledge of $(E[W_j'] - W_j^{*\prime})$ will be required. In finding lag effects, we may eliminate gradient noise from consideration so that $E[W_j'] = W_j'$. Knowledge of $(W_j' - W_j^{*\prime})$ will be sufficient.

Without gradient noise, the method of steepest descent and the LMS algorithm are represented by (13) and (20). With variable W_j^*, they become

$$W_{j+1} - (I - 2\mu R) W_j = 2\mu R W_j^*. \quad (66)$$

Premultiplying both sides by Q^{-1} transforms (66) into the primed coordinates,

$$W_{j+1}' - (I - 2\mu \Lambda) W_j' = 2\mu \Lambda W_j^{*\prime}. \quad (67)$$

We have assumed for our present study that all components of W_j^* are stationary, ergodic, independent, and first-order Markov; they all have the same variances and the same autocorrelation functions. Since $W_j^{*\prime} = Q^{-1} W_j^*$ and Q^{-1} is orthonormal, all components of $W_j^{*\prime}$ are independent and have the same autocorrelation functions as the components of W_j^*. Therefore, equation (67), being in diagonal form and having a driving function whose components are independent, may be treated as an array of n independent first-order linear difference equations.

Let the z transform of W_j' be $\mathcal{W}'(z)$. The z transform of (67) is then

$$z\mathcal{W}'(z) - (I - 2\mu\Lambda)\mathcal{W}'(z) = 2\mu\Lambda \mathcal{W}^{*\prime}(z). \quad (68)$$

Solving (68) yields the transform of W_j':

$$\mathcal{W}'(z) = 2\mu\Lambda(zI - I + 2\mu\Lambda)^{-1} \mathcal{W}^{*\prime}(z). \quad (69)$$

The weight tracking error $(W_j' - W_j^{*\prime})$ is of direct interest. Its transform is obtained from (69) as

$$\mathcal{W}'(z) - \mathcal{W}^{*\prime}(z) = [2\mu\Lambda(zI - I + 2\mu\Lambda)^{-1} - I]\mathcal{W}^{*\prime}(z). \quad (70)$$

The transfer function connecting $W_j^{*\prime}$ to the weight tracking error is thus

$$2\mu\Lambda(zI - I + 2\mu\Lambda)^{-1} - I. \quad (71)$$

Since (71) is diagonal, the scalar transfer function of its pth

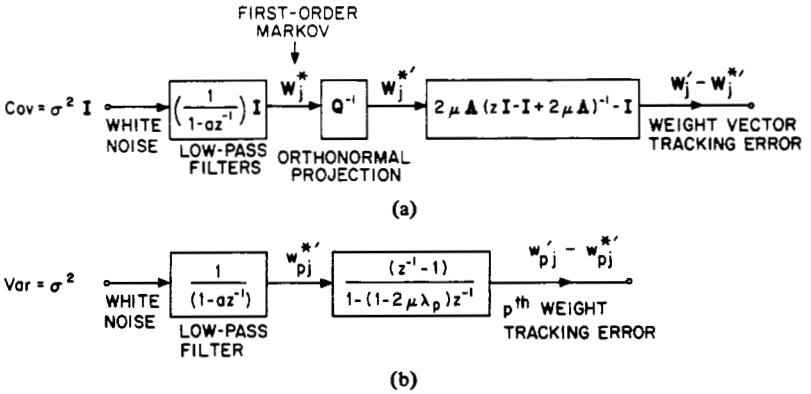

Fig. 10. Origin of W_j^* and its propagation into weight-lag error. (a) All channels. (b) pth channel.

diagonal element may be written as

$$2\mu\lambda_p(z - 1 + 2\mu\lambda_p)^{-1} - 1 = \frac{(z^{-1} - 1)}{1 - (1 - 2\mu\lambda_p)z^{-1}}. \quad (72)$$

This transfer function has a zero at $z = 1$ and a pole whose impulse response has a geometric ratio of $(1 - 2\mu\lambda_p) = r_p$.

Fig. 10(a) shows the origin of the vector W_j^* as a first-order Markov process and its propagation into the weight tracking error. W_j^* is assumed to originate from independent stationary ergodic white-noise excitation (of variance σ^2) to a bank of one-pole filters, all having transfer function $1/(1 - az^{-1})$. The pth channel of this process is shown in Fig. 10(b). Its scalar transfer function is

$$\frac{(z^{-1} - 1)}{(1 - az^{-1})(1 - (1 - 2\mu\lambda_p)z^{-1})} = \frac{(z^{-1} - 1)}{(1 - az^{-1})(1 - r_p z^{-1})}$$
$$= \frac{\left(\frac{1 - a}{a - r_p}\right)}{(1 - az^{-1})} + \frac{\left(\frac{r_p - 1}{a - r_p}\right)}{(1 - r_p z^{-1})}. \quad (73)$$

The sampled impulse response of this transfer function is obtained by inversion of (73) into the time domain. From it, the variance of the lag error of the pth component of the primed weight vector can be computed as the sum of the squares of the samples of the impulse response multiplied by σ^2. The sum of squares is given by

$$\text{sum squares} = \sum_{j=0}^{\infty}\left[\left(\frac{1 - a}{a - r_p}\right)a^j + \left(\frac{r_p - 1}{a - r_p}\right)r_p^j\right]^2$$
$$= \left(\frac{1}{a - r_p}\right)^2\left[\left(\frac{1 - a}{1 + a}\right) + \left(\frac{1 - r_p}{1 + r_p}\right) + \frac{2(1 - a)(r_p - 1)}{(1 - ar_p)}\right]. \quad (74)$$

In cases of interest, τ_p is large so that $r_p \lesssim 1$. From (27),

$$\tau_p = \frac{1}{1 - r_p} = \frac{1}{2\mu\lambda_p}. \quad (75)$$

Furthermore, we assume that the time constant of nonstationarity τ_{W^*} is also large, so that $a \lesssim 1$

$$\tau_{W^*} = \frac{1}{1 - a}. \quad (76)$$

A common operating region would be where

$$\tau_{W^*} >> \tau_p, \ \forall p. \quad (77)$$

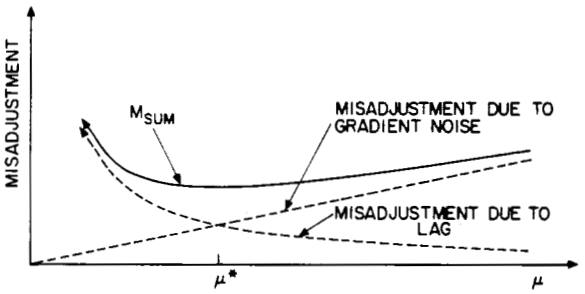

Fig. 11. Net misadjustment versus LMS convergence factor μ.

The value of μ is set so that the response times of the adaptive weights are short compared to the time constant of nonstationarity. Under these conditions, equation (74) reduces to

$$(\text{sum squares})_{\tau_{W^*} >> \tau_p} = \frac{1}{2}\tau_p = \frac{1}{4\mu\lambda_p}. \quad (78)$$

Using this relation, the covariance of the lag error is obtained as

$$\text{cov}\,[W_j' - W_j^{*\prime}]\Big|_{\substack{N=0 \\ \tau_{W^*} >> \tau_p}} = \frac{\sigma^2}{2}\begin{bmatrix}\tau_1 & & 0\\ & \ddots & \\ & \tau_p & \\ & & \ddots \\ 0 & & \tau_n\end{bmatrix} = \frac{\sigma^2}{4\mu}\Lambda^{-1}. \quad (79)$$

Making use of (64),

$$(\text{average excess mse due to lag}) = \frac{\sigma^2}{2}\sum_{p=1}^{n}\tau_p\lambda_p = \frac{n\sigma^2}{4\mu}. \quad (80)$$

Because of the ergodic properties of W_j^*, this average is not time variable. The misadjustment due to lag is

$$(M_L)_{\tau_{W^*} >> \tau_p} = \frac{\sigma^2}{2\xi_{\min}}\sum_{p=1}^{n}\tau_p\lambda_p = \left(\frac{n\sigma^2}{4\xi_{\min}}\right)\frac{1}{\mu}. \quad (81)$$

Under usual operating conditions, the misadjustment due to lag is inversely proportional to μ.

Set μ to a very small value so that the adaptive weight vector W_j does not track W_j^* but merely assumes the value of its time average. As $r_p \to 1$, equation (74) reduces to

$$(\text{sum squares})_{\mu \approx 0} = \tfrac{1}{2}\tau_{W^*}. \quad (82)$$

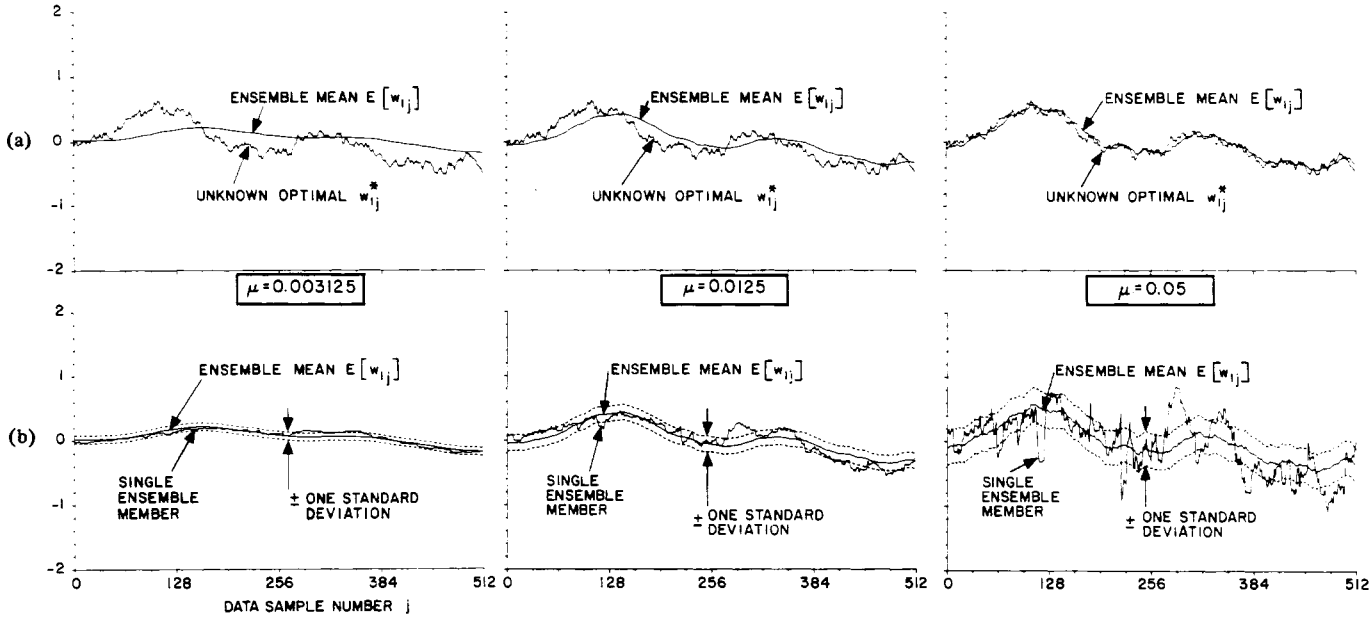

Fig. 12. Weight tracking in a nonstationary environment. (a) Plots demonstrate weight lag as a function of μ. (b) Plots demonstrate weight noise as a function of μ.

The misadjustment due to lag turns out to be

$$(NS) \triangleq (M_L)_{\mu \approx 0} = \frac{\sigma^2}{2\xi_{\min}} \tau_{W^*} \operatorname{tr} R. \tag{83}$$

Since there is no tracking, the misadjustment for this case is a measure of the "nonstationarity," NS, of the randomly moving hyperparaboloidal bowl.

An interesting special case is that of all equal eigenvalues. Combining (81) with (83),

$$(M_L)_{\tau_{W^*} >> \tau_p} = (NS)\left[\frac{\tau}{\tau_{W^*}}\right] = (NS)\left[\frac{2\tau_{\text{mse}}}{\tau_{W^*}}\right]. \tag{84}$$

This result has intuitive appeal. The misadjustment equals the product of nonstationarity and the ratio of the adaptive time constant to the time constant of nonstationarity.

From (63), the average excess mse is the sum of components due to gradient noise and lag. The total misadjustment is, therefore, the sum of two misadjustment components. Making use of (51) and (81),

$$(M_{\text{sum}})_{\tau_{W^*} >> \tau_p} = (\mu) \operatorname{tr} R + \left(\frac{1}{\mu}\right) \frac{n\sigma^2}{4\xi_{\min}}. \tag{85}$$

Optimizing the choice of μ results in minimum M_{sum} when the two right-hand terms are equal. The speed of adaptation is optimized when the loss of performance due to gradient noise equals the loss in performance due to weight-vector lag.[4] The optimal μ is

$$\mu^*|_{\tau_{W^*} >> \tau_p} = \left[\frac{n\sigma^2}{4\xi_{\min}(\operatorname{tr} R)}\right]^{1/2}. \tag{86}$$

A typical plot of M_{sum} versus μ is shown in Fig. 11, indicating the tradeoffs involved in adjusting μ for minimization of M_{sum}. In practice, μ^* might need to be approximated by trial and error, particularly when data are unavailable for application of (86).

The theory developed in this section has been tested extensively by computer simulation based on an ensemble of adaptive processes, as illustrated in Fig. 8. Every mathematical quantity discussed in this section has been measured. Typical experimental results are presented below.

Fig. 12 illustrates weight tracking and the associated errors. The adaptive filter had four weights. Responses are shown only for weight number one. The effects of weight lag are demonstrated by comparing the ensemble average of weight number one plotted over time against weight number one of W_j^*. Averages were taken over 128 ensemble members. The lag effect is highly evident in the first experiment with $\mu = 0.003125$. In the third experiment, with $\mu = 0.05$, the lag is quite small decreasing in proportion to μ.

The effects of gradient noise are demonstrated with the same experiment. The ensemble mean of weight number one is plotted as a function of time j. Theoretical one-standard-deviation lines for weight noise are shown about this mean. In addition, weight number one of W_j of a single ensemble member is plotted to indicate what occurred in an individual situation. It is clear that weight-noise power increases in proportion to μ.

In these experiments, the inputs were white and of unit power, so that $R = I$. The additive output noise power was $\xi_{\min} = 1$. Equation (85) has been used to obtain theoretical values of misadjustment and its components. Tables II, III, and IV summarize the results of three experiments, comparing theory and experiments for three values of μ, and fixing everything else. The input data were the same for all three experiments. Initial transients were allowed to die out before measurements were taken. Experimental values of misadjustment and its components were obtained by ensemble average measurements using (60), (64), and (65), normalizing with respect to $\xi_{\min}$, which in this case was 1. Theoretical and experimental results compared well, expect for lag misadjustment in the first experiment. In this case, where μ is very

[4] Another case has been analyzed by Widrow [29] where the fluctuation of W_j^* has a uniform low-pass power spectrum. In this case, the misadjustment due to lag is proportional to the square of μ; the speed of adaptation is optimized when the gradient-noise loss equals twice the loss due to lag. The misadjustment due to lag turns out to be quite sensitive to the spectral characteristics of the fluctuation of W_j^*.

TABLE II
FIRST EXPERIMENT, $\mu = 0.003125$

n = 4 weights τ_{mse} = 80 data samples τ_{W^*} = 125 data samples (NS) = 24.9%	Misadjustment Due to Weight Lag	Misadjustment Due to Gradient Noise
Theoretical	32.0%	1.25%
Experimental	13.5%	1.5%

TABLE III
SECOND EXPERIMENT, $\mu = 0.0125$

n = 4 weights τ_{mse} = 20 data samples τ_{W^*} = 125 data samples (NS) = 24.9%	Misadjustment Due to Weight Lag	Misadjustment Due to Gradient Noise
Theoretical	8.0%	5.0%
Experimental	5.6%	5.7%

TABLE IV
THIRD EXPERIMENT, $\mu = 0.05$

n = 4 weights τ_{mse} = 5 data samples τ_{W^*} = 125 data samples (NS) = 24.9%	Misadjustment Due to Weight Lag	Misadjustment Due to Gradient Noise
Theoretical	2.0%	20.0%
Experimental	1.8%	28.3%

small, equation (78) is inaccurate since τ_{W^*} is no longer much larger than τ_p.

Much more work needs to be done in the study of nonstationary adaptive behavior. We have presented a simplistic but meaningful beginning.

APPENDIX A
THE EFFICIENCY OF ADAPTIVE ALGORITHMS

We have analyzed the efficiency of the LMS algorithm from the point of view of misadjustment versus rate of adaptation. The question arises, could another algorithm be devised that would produce less misadjustment for the same rate of adaptation?

Suppose that an adaptive linear combiner is fed N independent input $n \times 1$ data vectors $X_1, X_2, \cdots, X_N$ drawn from a stationary ergodic process. Associated with these input vectors are their scalar desired responses $d_1, d_2, \cdots, d_N$, also drawn from a stationary ergodic process. Keeping the weights fixed, a set of N error equations can be written as

$$\epsilon_i = d_i - W^T X_i, \quad i = 1, 2, \cdots, N. \tag{A.1}$$

Let the objective be to find a weight vector that minimizes the sum of the squares of the error values based on a sample of N items of data.

Equation (A.1) can be written in matrix form as

$$\mathcal{E} = D - \mathcal{X} W \tag{A.2}$$

where $\mathcal{X}$ is an $N \times n$ rectangular matrix

$$\mathcal{X} \triangleq [X_1 X_2 \cdots X_N]^T \tag{A.3}$$

and where $\mathcal{E}$ is an N element error vector

$$\mathcal{E} \triangleq [\epsilon_1 \epsilon_2 \cdots \epsilon_N]^T. \tag{A.4}$$

A unique solution of (A.1), bringing $\mathcal{E}$ to zero, exists only if $\mathcal{X}$ is square and nonsingular. However, the case of greatest interest is that of $N >> n$. The sum of the squares of the errors is

$$\mathcal{E}^T\mathcal{E} = D^T D + W^T \mathcal{X}^T \mathcal{X} W - 2D^T \mathcal{X} W. \tag{A.5}$$

This sum multiplied by $1/N$ is an estimate $\hat{\xi}$ of the mse ξ. Thus

$$\hat{\xi} = \frac{1}{N}\mathcal{E}^T\mathcal{E} \quad \text{and} \quad \lim_{N\to\infty} \hat{\xi} = \xi. \tag{A.6}$$

Note that $\hat{\xi}$ is a quadratic function of the weights, the parameters of the quadratic form being related to properties of the N data samples. $(\mathcal{X}^T\mathcal{X})$ is square and positive semidefinite. $\hat{\xi}_{min}$ is the small-sample-size mse function, while ξ is the large-sample-size mse function. These functions are sketched in Fig. 13.

The function $\hat{\xi}$ is minimized by setting its gradient to zero:

$$\nabla\hat{\xi} = 2\mathcal{X}^T\mathcal{X} W - 2\mathcal{X}^T D. \tag{A.7}$$

The "optimal" weight vector based only on the N data samples is

$$\hat{W}^* \triangleq (\mathcal{X}^T\mathcal{X})^{-1}\mathcal{X}^T D. \tag{A.8}$$

This formula gives the position of the minimum of the small-sample-size bowl. The corresponding formula for the large-sample-size bowl is the Wiener–Hopf equation (8).

We could calculate $\hat{W}^*$ by a training process, a regression process, LMS, or some other optimization procedure. Taking the first block of N data samples, we obtain a small-sample-size function $\hat{\xi}_1$ whose minimum is at $\hat{W}_1^*$. This could be repeated with a second data sample, giving a function $\hat{\xi}_2$ whose minimum is at $\hat{W}_2^*$, etc. Typically, all the values of $\hat{W}^*$ would differ from the true optimum W^* and would, thereby, be misadjusted.

To analyze the misadjustment, assume that N is large and that the typical small-size curve approximately matches the large-sample-size curve. Therefore,

$$\hat{\xi} \approx \xi \quad \text{and} \quad (\xi - \hat{\xi}) \triangleq d\xi. \tag{A.9}$$

The true large-sample-size function is

$$\xi = \xi_{min} + V'^T \Lambda V'.$$

The gradient of this function expressed in the primed coordinates is

$$\nabla' = 2\Lambda V'.$$

A differential deviation in the gradient is

$$(d\nabla') = 2\Lambda(dV') + 2(d\Lambda)V'. \tag{A.10}$$

This deviation could represent the difference in gradients between small- and large-sample-size curves.

Refer to Fig. 13. Let $W' = W^{*\prime}$, then $V' = 0$. The gradient

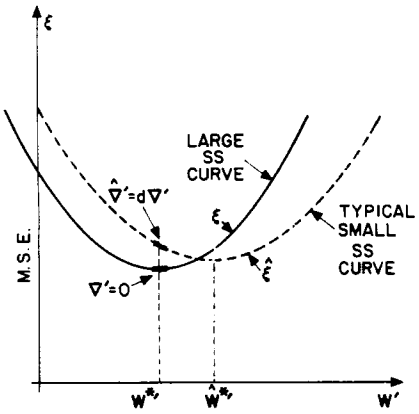

Fig. 13. Small- and large-sample-size mse curves.

of ξ is zero, while the gradient of $\hat{\xi}$ is $\hat{\nabla}' = d\hat{\nabla}$. Using (A.10),

$$(d\nabla') = 2\Lambda(dV'). \tag{A.11}$$

From (A.11), the deviation in gradient can be linked to the deviation in position of the small-sample-size curve minimum since $(dV') = (W^{*\prime} - \hat{W}^{*\prime})$. Taking averages of (A.11) over an ensemble of small-sample-size curves,

$$\text{cov}\,[d\nabla'] = 4\Lambda\,\text{cov}\,[dV']\Lambda. \tag{A.12}$$

Equation (42) indicates that the covariance of the gradient noise when $W' = W^{*\prime}$ is given by $4\xi_{\min}\Lambda$. If the gradient were estimated under the same conditions but using N independent error samples,

$$\text{cov}\,[d\nabla'] = \frac{4}{N}\xi_{\min}\Lambda. \tag{A.13}$$

Substituting this into (A.12) yields

$$\text{cov}\,[dV'] = \frac{1}{N}\xi_{\min}\Lambda^{-1}. \tag{A.14}$$

The average excess mse, an ensemble average, is

$$\begin{pmatrix}\text{average}\\ \text{excess}\\ \text{mse}\end{pmatrix} = E[(dV')^T\Lambda(dV')]. \tag{A.15}$$

Equation (A.14) shows cov $[dV']$ to be diagonal, so that

$$\begin{pmatrix}\text{average}\\ \text{excess}\\ \text{mse}\end{pmatrix} = \frac{n}{N}\xi_{\min}. \tag{A.16}$$

The misadjustment is, therefore,

$$M = \frac{n}{N} = \frac{\text{(number of weights)}}{\text{(number of independent training samples)}}. \tag{A.17}$$

This formula was first presented without detailed proof by Widrow and Hoff [1] in 1960. It has been used for many years in pattern recognition studies. For small values of M (less than 25 percent), it has proven to be very useful. A formula similar to (A.17), although based on somewhat different assumptions, was derived by Davisson [36] in 1970.

Although equation (A.17) has been derived for training with finite blocks of data, it can be used to assess the efficiency of steady-flow algorithms. Consider an adaptive transversal filter with stationary stochastic inputs, adapted by the LMS algorithm. For simplicity, let all eigenvalues of R be equal. As such

$$M = \frac{n}{4\tau_{\text{mse}}}. \tag{56}$$

The LMS algorithm exponentially weights its input data over time in determining current weight values. If an equivalent uniform averaging window is assumed equal to the adaptive settling time, approximately four time constants, the equivalent data sample taken at any instant by LMS is essentially $N_{\text{eq}} = 4\tau_{\text{mse}}$ samples. Accordingly for LMS,

$$M = \frac{n}{N_{\text{eq}}}. \tag{A.18}$$

A comparison of (A.18) and (A.17) shows that when eigenvalues are equal, LMS is about as efficient as a least squares algorithm can be.[5] However, with disparate eigenvalues, the misadjustment is primarily determined by the fastest modes while settling time is limited by the slowest modes. To sustain efficiency with disparate eigenvalues, algorithms similar to LMS have been devised based on Newton's method rather than on steepest descent [38], [39]. Such algorithms premultiply the gradient estimate each iteration cycle by an estimate of the inverse of R:

$$W_{j+1} = W_j + \mu\widehat{R^{-1}}\hat{\nabla}_j$$

or

$$W_{j+1} = W_j + 2\mu\widehat{R^{-1}}\epsilon_j X_j. \tag{A.19}$$

This process causes all adaptive modes to have essentially the same time constant. Algorithms based on this principle are potentially more efficient that LMS but are typically more difficult to implement.

ACKNOWLEDGMENT

The authors wish to acknowledge the helpful discussions and contributions of C. S. Williams and J. R. Treichler of Stanford University; Dr. O. L. Frost III of Argo Systems, Inc.; and Dr. M. E. Hoff, Jr. of the Intel Corporation. Special thanks are also due Diane Byron, who assisted in editing the paper.

REFERENCES

[1] B. Widrow and M. E. Hoff, "Adaptive switching circuits," in *1960 WESCON Conv. Rec.*, pt. 4, pp. 96–140.
[2] N. Nilsson, *Learning Machines*. New York: McGraw-Hill, 1965.
[3] J. Koford and G. Groner, "The use of an adaptive threshold element to design a linear optimal pattern classifier," *IEEE Trans. Inform. Theory*, vol. IT-12, pp. 42–50, Jan. 1966.
[4] B. Widrow, P. Mantey, L. Griffiths, and B. Goode, "Adaptive antenna systems," *Proc. IEEE*, vol. 55, pp. 2143–2159, Dec. 1967.
[5] B. Widrow, "Adaptive filters," in *Aspects of Network and System Theory*, R. Kalman and N. DeClaris, Eds. New York: Holt, Rinehart, and Winston, 1971, pp. 563–587.
[6] S. P. Applebaum, "Adaptive arrays," Special Projects Lab., Syracuse Univ. Res. Corp., Rep. SPL 769.
[7] L. J. Griffiths, "A simple adaptive algorithm for real-time processing in antenna arrays," *Proc. IEEE*, vol. 57, pp. 1696–1704, Oct. 1969.
[8] O. L. Frost III, "An algorithm for linearly constrained adaptive array processing," *Proc. IEEE*, vol. 60, pp. 926–935, Aug. 1972.
[9] W. F. Gabriel, "Adaptive arrays—An introduction," *Proc. IEEE*, vol. 64, pp. 239–272, Feb. 1976.

[5] Attempts have been made to devise algorithms more efficient than LMS by using variable μ [37]. Initial values of μ are chosen high for rapid convergence; final values of μ are chosen low for small misadjustment. This works as long as input statistics are stationary. This procedure and the methods of stochastic approximation on which it is based will not perform well in the nonstationary case.

[10] F. W. Smith, "Design of quasi-optimal minimum-time controllers," *IEEE Trans. Automat. Contr.*, vol. AC-11, pp. 71-77, Jan. 1966.
[11] B. Widrow, "Adaptive model control applied to real-time blood-pressure regulation," in *Pattern Recognition and Machine Learning*, Proc. Japan-U.S. Seminar on the Learning Process in Control Systems, K. S. Fu, Ed. New York: Plenum Press, 1971, pp. 310-324.
[12] R. Lucky, "Automatic equalization for digital communication," *Bell Syst. Tech. J.*, vol. 44, pp. 547-588, Apr. 1965.
[13] M. DiToro, "A new method of high-speed adaptive serial communication through any time-variable and dispersive transmission medium," in *Conf. Record*, 1965 IEEE Annual Communications Convention, pp. 763-767.
[14] R. Lucky and H. Rudin, "An automatic equalizer for general-purpose communication channels," *Bell Syst. Tech. J.*, vol. 46, pp. 2179-2208, Nov. 1967.
[15] R. Lucky *et al.*, *Principles of Data Communication*. New York: McGraw-Hill, 1968.
[16] A. Gersho, "Adaptive equalization of highly dispersive channels for data transmission," *Bell Syst. Tech. J.*, vol. 48, pp. 55-70, Jan. 1969.
[17] M. Soudhi, "An adaptive echo canceller," *Bell Syst. Tech. J.*, vol. 46, pp. 497-511, Mar. 1967.
[18] B. Widrow *et al.*, "Adaptive noise cancelling: Principles and applications," *Proc. IEEE*, vol. 63, pp. 1692-1716, Dec. 1975.
[19] P. E. Mantey, "Convergent automatic-synthesis procedures for sampled-data networks with feedback," Stanford Electronics Laboratories, Stanford, CA, TR no. 7663-1, Oct. 1964.
[20] P. M. Lion, "Rapid identification of linear and nonlinear systems," in *Proc. 1966 JACC*, Seattle, WA, pp. 605-615, Aug. 1966; also *AIAA Journal*, vol. 5, pp. 1835-1842, Oct. 1967.
[21] R. E. Ross and G. M. Lance, "An approximate steepest descent method for parameter identification," in *Proc. 1969 JACC*, Boulder, CO, pp. 483-487, Aug. 1969.
[22] R. Hastings-James and M. W. Sage, "Recursive generalized-least-squares procedure for online identification of process parameters," *Proc. IEE*, vol. 116, pp. 2057-2062, Dec. 1969.
[23] A. C. Soudack, K. L. Suryanarayanan, and S. G. Rao, "A unified approach to discrete-time systems identification," *Int. J. Control*, vol. 14, no. 6, pp. 1009-1029, Dec. 1971.
[24] W. Schaufelberger, "Der Entwurf adaptiver Systeme nach der direckten Methode von Ljapunov," *Nachrichtentechnik*, Nr. 5, pp. 151-157, 1972.
[25] J. M. Mendel, *Discrete Techniques of Parameter Estimation: The Equation Error Formulation*. New York: Marcel Dekker, Inc., 1973.
[26] S. J. Merhav and E. Gabay, "Convergence properties in linear parameter tracking systems," *Identification and System Parameter Estimation—Part 2*, Proc. 3rd IFAC Symp., P. Eykhoff, Ed. New York: American Elsevier Publishing Co., Inc., 1973, pp. 745-750.
[27] R. V. Southwell, *Relaxation Methods in Engineering Science*. New York: Oxford, 1940.
[28] D. J. Wilde, *Optimum Seeking Methods*. Englewood Cliffs, NJ: Prentice-Hall, 1964.
[29] B. Widrow, "Adaptive sampled-data systems," in *Proc. First Intern. Cong. Intern. Federation of Automatic Control*, Moscow, 1960.
[30] H. Robbins, and S. Monro, "A stochastic approximation method," *Ann. Math. Statist.*, vol. 22, pp. 400-407, 1951.
[31] A. Dvoretzky, "On stochastic approximation," in *Proc. Third Berkeley Symp. Math. Statist. and Probability*, J. Neyman, Ed. Berkeley, CA: University of California Press, 1956, pp. 39-55.
[32] J. Makhoul, "Linear prediction: A tutorial review," *Proc. IEEE*, vol. 63, pp. 561-580, Apr. 1975.
[33] L. J. Griffiths, "Rapid measurement of digital instantaneous frequency," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-23, pp. 207-222, Apr. 1975.
[34] Y. T. Chien, K. S. Fu, "Learning in non-stationary environment using dynamic stochastic approximation," in *Proc. 5th Allerton Conf. Circuit and Systems Theory*, pp. 337-345, 1967.
[35] T. P. Daniell and J. E. Brown III, "Adaptation in nonstationary applications," in *Proc. 1970 IEEE Symp. Adaptive Processes (9th)*, Austin, TX, paper no. XXIV-4, Dec. 1970.
[36] L. D. Davisson, "Steady-state error in adaptive mean-square minimization," *IEEE Trans. Inform. Theory*, vol. IT-16, pp. 382-385, July 1970.
[37] T. J. Schonfeld and M. Schwartz, "A rapidly converging first-order training algorithm for an adaptive equalizer," *IEEE Trans. Inform. Theory*, vol. IT-17, pp. 431-439, July 1971.
[38] K. H. Mueller, "A new, fast-converging mean-square algorithm for adaptive equalizers with partial-response signaling," *Bell Syst. Tech. J.*, vol. 54, pp. 143-153, Jan. 1975.
[39] L. J. Griffiths and P. E. Mantey, "Iterative least-squares algorithm for signal extraction," in *Proc. Second Hawaii Int. Conf. System Sciences*, Western Periodicals Co., pp. 767-770, 1969.

An Adaptive Nonparametric Linear Classifier

GUSTAV N. WASSEL, MEMBER, IEEE AND JACK SKLANSKY, SENIOR MEMBER, IEEE

Abstract—**The equalized-error ("EE") training procedure, introduced in this paper, is a new nonparametric training procedure for linear classifiers in a multiple-feature stochastic environment. This procedure is a form of stochastic approximation that minimizes the sum of the expected normalized first moments of the falsely classified pattern vectors about the decision hyperplane. This sum is the "EE loss function." The minimization is achieved by a simply implemented recursive equation. We show that the sequence of decision hyperplanes generated by this recursive equation converges in mean square and with probability one to a hyperplane that minimizes the EE loss function.**

We provide preliminary qualitative and quantitative evidence that the EE training procedure converges rapidly and achieves low asymptotic error probabilities over a wide range of overlapping pairs of class densities and nonlinearly separable pairs of class densities.

Manuscript received March 2, 1976; revised March 31, 1976. This work was supported in part by the National Institute of General Medical Sciences under U.S. Public Health Service Grant GM-17632, in part by the U.S. Air Force of Scientific Research under Grant AFOSR 69-1813, and in part by the National Science Foundation under Science Faculty Fellowship 60196.

G. N. Wassel is with the Department of Electrical Engineering, California State Polytechnic University, Pomona, CA 91766.

J. Sklansky is with the School of Engineering, University of California, Irvine, CA 92664.

I. INTRODUCTION

THE NEED FOR automatic classification occurs in a tremendous range of engineering and social problems: e.g., navigation, medical diagnosis, aerial reconnaissance, satellite photography, communication systems, psychological