Tracking Properties and Steady-State Performance of RLS Adaptive Filter Algorithms

EVANGELOS ELEFTHERIOU AND DAVID D. FALCONER, FELLOW, IEEE

Abstract—Adaptive signal processing algorithms derived from LS (least squares) cost functions are known to converge extremely fast and have excellent capabilities to "track" an unknown parameter vector. This paper treats analytically and experimentally the steady-state operation of RLS (recursive least squares) adaptive filters with exponential windows for stationary and nonstationary inputs. A new formula for the "estimation-noise" has been derived involving second- and fourth-order statistics of the filter input as well as the exponential windowing factor and filter length. Furthermore, it is shown that the adaptation process associated with "lag effects" depends solely on the exponential weighting parameter \( \lambda \). In addition, the calculation of the excess mean square error due to the lag for an assumed Markov channel provides the necessary information about tradeoffs between speed of adaptation and steady-state error. It is also the basis for comparison to the simple LMS algorithm. In a simple case of channel identification, it is shown that the LMS and RLS adaptive filters have the same tracking behavior. Finally, in the last part, we present new RLS restart procedures applied to transversal structures for mitigating the disastrous results of the third source of noise, namely, finite precision arithmetic.

I. INTRODUCTION

A n area which is of strong current practical importance and research interest is the adaptive channel estimation and equalization of rapidly time-varying channels.

Adjustment algorithms for adaptive filtering derived from LS (least squares) cost functions are known to converge extremely fast and have excellent capabilities working in a time-varying environment. Although various simulation results confirming these facts can be found in the literature, there is little theoretical work published describing the steady-state performance characteristics of the RLS (recursive least squares) adaptive filter.

It is known that all adaptive filters capable of adapting at real-time rates experience losses in performance because their adjustments are based on statistical averages taken with limited sample sizes [1]. For exponentially windowed RLS algorithms, these losses expressed through the excess MSE (mean square error) are a result of two main sources of error. The first source of error is attributed to the exponential weighting of the squared error sequence and therefore to the exponential nature of the estimators used to estimate the correlation matrix and the cross-correlation vector [3], i.e., finite window effect. This error, which we shall call "estimation-noise," results in a misadjustment of the coefficient vector of the adaptive filter from its optimal setting. An analogy to the previous situation is encountered with the LMS (least mean square) adaptive filter because of the gradient noise [5]. The second source of error is associated with filtering nonstationary signals. This error, which has been called "lag error," is caused by the attempt of the adaptive system to track variations of the input signal. At this point, we should emphasize that finite-precision arithmetic also contributes to the excess MSE, and in that sense could be considered as a third source of noise. More important, however, is the fact that roundoff errors trigger numerical instabilities posing potential problems in implementing the RLS adaptive algorithms.

The analysis of the stationary and nonstationary characteristics of the LMS algorithm can be found in the pioneering work of Widrow et al. in [5]. In [4], the tracking ability of a wide class of adaptive signal processing algorithms has been studied. That work develops an upper bound on the squared error between the parameter vector being tracked and the value obtained by the algorithm. Also, in [6] and [7], a more or less qualitative analysis of a preliminary experimental examination of the response of an adaptive lattice predictor and a Kalman estimator to nonstationary inputs has been presented, respectively. Very recently, the analysis of the behavior of an orthogonalized LMS adaptive filter in a time-varying environment appeared in [8]. A discussion of the steady-state operation limitations of RLS adaptive algorithms, because of the finite precision arithmetic and possible solutions, can be found in the work of Cioffi and Kailath [10] and Lin [9]. Finally, work more closely related to the subject of the present paper, and especially to the part dealing with the calculation of the "lag error" time constant, has been presented by Ling and Proakis in [11].

In this paper we present an attempt at quantitative understanding of the steady-state performance characteristics of adaptive filters driven by RLS algorithms. A new, more accurate expression for the "estimation-noise" than the one given in [3] and [11] has been derived. Furthermore, the general tracking ability and the explicit calculation of...
the excess MSE due to the lag effects for an assumed Markov channel provides the necessary information about the tradeoffs between speed of adaptation and steady-state performance. A number of simplifying assumptions is made in order to obtain simple results which give insight into the adaptation process. Finally, we describe new restart procedures which guarantee smooth reinitialization and steady-state operation of the adaptive algorithm without introducing any significant performance-degrading transients. These restart procedures appear to make the RLS or FRLS (fast recursive least squares) transversal filter algorithms viable for adaptive equalization or channel estimation of time-varying channels.

Throughout our analysis for the estimation and lag error noises, we use the mathematically more tractable transversal filter realization as opposed to a lattice filter realization, without any loss of generality, since it has been shown, e.g., [12], that these algorithms perform similarly, neglecting errors due to finite word length, and they are all equally fast converging.

The remainder of this paper is organized as follows. Section II is a brief review of the RLS adaptive filters. This section cites the general concepts of RLS adaptive algorithms and establishes the relevant notation that will be used in the sequel. Section III derives a new formula for the excess error due to the "estimation-noise." Section IV presents analytical results concerning the behavior of the RLS adaptive filters in a time-varying environment. Section V discusses restart procedures for RLS transversal algorithms as a remedy to numerical roundoff errors. Section VI presents experimental results of computer simulations that verify the developed theory. Finally, Section VII is a brief conclusion.

II. RECURSIVE LEAST SQUARES ALGORITHMS

In this section we describe briefly the basic principles of the LS adaptation problem. The general adaptive filter is shown in Fig. 1. The system is considered to be of the tapped delay line form of length \( N \). The impulse response of the system is denoted by the \( N \times 1 \) complex vector \( C(n) \) (where \( n \) is the discrete time index). The adaptive system is acting on the input signal samples \( x(n) \), while the other input \( d(n) \) is the so-called "desired response." If at time \( n \) the adaptive filter has the \( N \) latest inputs stored denoted by the vector

\[
X(n) = (x(n)^*, x(n - 1)^*, \ldots, x(n - N + 1)^*)^*,
\]

(2.1)

then the adaptive filter output at time \( n \), \( (C(n - 1)^* X(n)) \), may differ from the ideal output \( d(n) \) by an error \( e(n) \)

\[
e(n) = d(n) - (C(n - 1)^* X(n)).
\]

(2.2)

The objective of the LS algorithms is to generate that tap-coefficient vector \( C(n) \) at time \( n \) which minimizes the weighted cumulative squared error [12], [13],

\[
J(n) = \sum_{k=0}^{n} \lambda^{n-k} |d(k) - C(n)^* X(k)|^2.
\]

(2.3)

In the sense of this error minimization criterion then, the recursive least squares estimation algorithm makes the best possible use of all data \( \{X(k), d(k)\} \) up to time \( n \); therefore, in this sense, it converges and tracks "as fast as possible." The parameter \( \lambda \) is some positive constant close to but less than 1 used for exponential weighting of the past. Roughly speaking, \( 1/(1 - \lambda) \) represents the memory of the algorithm.

The minimizing vector is the solution of the discrete time, Wiener–Hopf equation

\[
C(n) = \hat{R}(n)^{-1} \hat{V}(n)
\]

(2.4)

where

\[
\hat{R}(n) = \sum_{k=0}^{n} \lambda^{n-k} X(k) X(k)^*
\]

(2.5)

and

\[
\hat{V}(n) = \sum_{k=0}^{n} \lambda^{n-k} X(k) d(k)^*.
\]

(2.6)

It can be seen that for \( \lambda = 1 \) and large \( n \), \( (1/n) \hat{R}(n) \) is a consistent estimate of the input signal vector autocorrelation matrix and \( (1/n) \hat{V}(n) \) is a consistent estimate of the cross correlation between the input signal vector and the desired response [12].

At this point it should be noted that the solution of (2.4) could follow basically along one of two possible directions. The first procedure is based on the fact that (2.5) and (2.6) can be written recursively in time and, therefore, that the sequence \( C(n) \) obeys the following time update relation [12], [13]:

\[
C(n) = C(n - 1) + K(n) e(n)^*
\]

(2.7)

where \( K(n) \) is the Kalman gain defined as

\[
K(n) = \hat{R}(n)^{-1} X(n).
\]

(2.8)

The inverse estimated correlation matrix \( \hat{R}(n)^{-1} \) displayed in (2.8) greatly accelerates the adaptation process since it
performs the appropriate orthogonalization of the autocorrelation matrix as a means to very fast adaptation [14]. This approach, i.e., (2.7) and (2.8), for solving (2.4) leads to the transversal adaptive filter structure; that is the adaptive filter model we are going to use throughout our analysis.

A second approach for solving (2.4) is by lattice methods [15], [18]. In this method, certain properties of the estimated autocorrelation matrix are used in order to obtain order-updated equations. We should emphasize that the RLS algorithms of lattice or transversal filter form perform the same minimization of (2.3) and, therefore, offer the same convergence and tracking ability. The difference lies in the manner and the complexity with which the minimization is achieved [12] and the numerical stability properties as well.

Both approaches may be implemented with fast recursive least squares (FRLS) algorithms which exploit the shift relationship between successive input vectors \(X(n)\) to limit the computational complexity of the order of \(N\) [10], [12], [13], [15], [18].

III. RLS ADAPTIVE FILTERS AND THE "ESTIMATION-NOISE"

During the adaptation process, the filter is trying to match its coefficients \(C(n)\) with those of some unknown parameter vector. For instance, in a channel estimation situation this parameter vector will be the unknown impulse response to be identified, while in the equalization case it will be the impulse response that minimizes the overall mean squared error.

It has been shown [10] that, for \(\lambda = 1\) and \(n \to \infty\), LS adaptation algorithms realize the optimum Wiener solution \(C_{\text{opt}}\). Unfortunately, things change drastically if one uses exponentially windowed RLS algorithms in order to track possible variations in the unknown channel. In that case, the inconsistency of the exponential weighted estimators, and generally any finite window estimators, causes noise in the tap coefficients and causes them to be on the average misadjusted from their optimal values. This noise in the tap coefficient vector \(C(n)\) is the source of an excess error (the error above the MSE of the Wiener solution), namely, "estimation-noise" or "misadjustment-noise."

We recall [2], [5], [14] that the excess MSE is

\[
E\{\Delta C(n)^* \cdot R \cdot \Delta C(n)\}
\]

Here \(R\) is a positive definite matrix with elements

\[
r_{ij} = E\{x(n - i + 1) x(n - j + 1)^*\} \quad 1 \leq i, j \leq N
\]

and \(\Delta C(n)\) is an \(N \times 1\)-dimensional tap-coefficient error vector, i.e.,

\[
\Delta C(n) = C(n) - C_{\text{opt}}
\]

where \(C_{\text{opt}}\) is the solution of the following matrix equation:

\[
C_{\text{opt}} = R^{-1} V,
\]

and \(V\) denotes another \(N \times 1\)-dimensional vector with elements

\[
u_i = E\{x(n - i + 1) d(n)^*\}. \quad (3.4)
\]

Using the adaptation equation (2.7) along with the definition of the Kalman gain and the error sequence \(e(n)\), we can easily show that

\[
\hat{R}(n) C(n) = \lambda \hat{R}(n - 1) C(n - 1) + X(n) d(n)^*\quad (3.5)
\]

Defining the optimum error at time \(n\) as

\[
e_{\text{opt}}(n) = d(n) - C_{\text{opt}}^* X(n) \quad (3.6)
\]

and using the fact that

\[
\hat{R}(n) = \lambda \hat{R}(n - 1) + X(n) X(n)^*\quad (3.7)
\]

equation (3.5) yields the following recursive relation for the tap-coefficient error vector:

\[
\hat{R}(n) \Delta C(n) = \lambda \hat{R}(n - 1) \Delta C(n - 1) + X(n) e_{\text{opt}}(n)^*\quad (3.8)
\]

The solution to this equation reads

\[
\Delta C(n) = \lambda^n \hat{R}(n)^{-1} \hat{R}(0) \Delta C(0) + \hat{R}(n)^{-1} \sum_{k=1}^{n} \lambda^{n-k} X(k) e_{\text{opt}}(k)\quad (3.9)
\]

where \(\Delta C(0)\) and \(\hat{R}(0)\) are iterations starting fixed quantities. In particular, for the exponential windowed RLS algorithm \(\hat{R}(0)\) is a diagonal matrix such that

\[
\hat{R}(0) = \delta I_{NN}\quad (3.10)
\]

where \(I_{NN}\) indicates the \(N \times N\) unit matrix and \(\delta\) is a small constant which ensures positive definiteness of \(\hat{R}(n)\) for all \(n\).

Although our ultimate aim is an expression for the excess mean square error, we shall begin with the average tap-coefficient error vector, a simpler quantity, but one which gives some insight on the initial convergence and the parameters involved in it. Taking expected values on both sides of (3.9), we get

\[
E\{\Delta C(n)\} = \lambda^n \delta E\{\hat{R}(n)^{-1}\} \Delta C(0) + E\left\{\hat{R}(n)^{-1} \sum_{k=1}^{n} \lambda^{n-k} X(k) e_{\text{opt}}(k)^*\right\}\quad (3.11)
\]

The optimum error sequence \(e_{\text{opt}}(n)\) can be shown to be orthogonal [19] with \(X(n)\). We also assume that \(\hat{R}(n)\) is independent of \(X(k)\) and \(e_{\text{opt}}(k)\). Therefore, the second term in (3.11) is approximately zero, leaving only the first term as the dominant one during initial convergence. The fact that \(\lambda^n \to 0\) for \(|\lambda| < 1\) (for the case of \(\lambda = 1\), see [10]), and the asymptotic stationarity of the estimated correlation matrix \(\hat{R}(n)\), indicates that the expected tap-coef-
ficient error vector converges always to zero as long as \( \hat{R}(n) \) retains its positive definite nature. We also observe that RLS tap-coefficient convergence is independent from the eigenvalue distribution. Only the product \( E\{R(n)^{-1} \hat{R}(n)\} \), which in a worst case scenario has a norm on the same order as the product of \( \delta \) and the inverse of the smallest eigenvalue of \( R \), affects in a proportional manner initial convergence. An excellent discussion on the choice of \( \delta \) can be found in [10]. The above-mentioned behavior is in contrast to the initial convergence of the LMS algorithm. In the latter case, the eigenvalues themselves form the convergence time constants and determine the value of the adaptation size \( \mu \) which ensures stability.

In the sequel, the asymptotic behavior of the mean square error is going to be studied. In particular, our major concern will be directed to the steady-state analysis of the average excess mean square error in RLS adaptive filters denoted by \( E\{\Delta C(n)^* \hat{R}C(n)\} \), i.e., the average of the error \( \Delta C(n)^* \hat{R}C(n) \) at time \( n \) with respect to all possible input sequences \( \{X(n)\} \) and \( \{d(n)\} \). In order to facilitate mathematical treatment, \( \Delta C(n-1) \) and \( X(n) \) are assumed independent of each other.

Let \( q_n \) be defined as

\[
q_n = \Delta C(n)^* R \Delta C(n). \tag{3.12}
\]

In Appendix A-1 we show that at the steady-state \( E\{q_n\} \) satisfies the following relation:

\[
E\{q_n\} = \lambda^2 E\{q_{n-1}\} + (1 - \lambda)^2 \text{trace} \{I_{NN} + E\{P(n)^2\}\} \varepsilon_{opt} \tag{3.13}
\]

where \( P(n) \) is a zero mean fluctuation matrix. The matrix \( S = E\{P(n)^2\} \) has entries which depend on some fourth-order statistics of the received signal [see (A2.14)]. Under these conditions, the asymptotic value of \( E\{q_n\} \) will be

\[
\varepsilon_{est} = E_{\infty} \{q_n\} = \frac{1 - \lambda}{1 + \lambda} \text{trace} \{I_{NN} + E\{P(n)^2\}\} \varepsilon_{opt}. \tag{3.14}
\]

The above formula is a general expression for the excess MSE due to "estimation-noise" in exponential weighted RLS adaptive algorithms.

A simplification of (3.14), which retains the essence and emphasizes more the quantitative characterization of the "estimation-noise," can be achieved if we further assume that in the steady state the diagonal elements only of \( R \) and \( \hat{R}(n) \) contribute to the estimation error [see Appendix A-2, (A2.7)]. Thus, asymptotically, \( S \) will be of diagonal form with elements

\[
s_{ij} = \begin{cases} 
\frac{(1 - \lambda)^2 |x_i|^2}{1 + \lambda (\sigma_i^2)^2} & \text{if } i = j \\
0 & \text{if } i \neq j
\end{cases} \tag{3.15}
\]

where

\[
\gamma = \frac{\sigma^2 |x|^2}{(\sigma_i)^2} \tag{3.16a}
\]

and

\[
(\sigma_e^2) = \text{var} (|x(n)|^2) \tag{3.16b}
\]

\[
(\sigma^2) = \text{var} (x(n))^2. \tag{3.16c}
\]

The statistical parameter \( \gamma \) can be computed in a straightforward manner for given channel and signal statistics. In that case, it can easily be shown that the steady-state excess mean square error will be

\[
\varepsilon_{est} = \frac{1 - \lambda}{1 + \lambda} \left( \frac{1 - \lambda}{1 + \lambda} \gamma \right) N \varepsilon_{opt}. \tag{3.17}
\]

This formula is an expression for the "estimation-noise." It gives a quantitative insight of the excess MSE due to the exponential nature of the RLS estimators. If we define the "misadjustment" as the dimensionless ratio of the excess MSE to the minimum MSE [5], we get

\[
M = \frac{1 - \lambda}{1 + \lambda} \left( \frac{1 - \lambda}{1 + \lambda} \right) N. \tag{3.18}
\]

For values of \( \lambda \) very close to 1 and \( \gamma \) reasonably small (for instance, if the input samples are assumed to be Gaussian distributed then \( \gamma = 2 \)), it can be shown easily that

\[
1 + \frac{1 - \lambda}{1 + \lambda} \gamma \approx 1. \tag{3.19}
\]

Therefore, (3.17) becomes

\[
\varepsilon_{est} = \frac{1 - \lambda}{1 + \lambda} N \varepsilon_{opt} \tag{3.20}
\]

and the corresponding misadjustment formula will be

\[
M = \frac{1 - \lambda}{1 + \lambda} N. \tag{3.21}
\]

The result of (3.20) had first appeared in [3]. A different proof was also given in [11]. That expression is less general than the ones we derived [(3.14), (3.17)] because its applicability is restricted to values of \( \lambda \) very close to 1. The key assumption in its derivation is (3.19), which essentially is equivalent to modeling the estimated autocorrelation matrix as follows:

\[
\hat{R}(n)^{-1} R = (1 - \lambda) I_{NN} \quad \text{as} \quad n \to \infty. \tag{3.22}
\]

This model is a degenerate form of the relation (A2.4) in Appendix A-2, where the asymptotic variance of the elements of the perturbation matrix is considered to be very small or zero. Apparently, as we decrease \( \lambda \), the above assumption can no longer be justified. Thus, the influence of the statistical parameter \( \gamma \) becomes noticeable.

In any case, we see that as the memory of the system decreases so the misadjustment increases. In general, fast adaptation leads to a more noisy adaptation process. In a stationary environment, the best steady-state performance results from slow adaptation which corresponds to \( \lambda = 1 \).

Concluding, we should emphasize again that the formulas (3.14) and (3.17) we derived for the excess MSE
indicate that the level of the estimation noise does not depend solely on the nature of the "windowing" but on some fourth-order statistics of the input signal as well. The latter is more noticeable in fast adaptation where \( \lambda \) may take values as low as 0.9.

IV. RLS Adaptive Filter Response to Time-Varying Inputs

A. Tracking Behavior

The problem we examine here involves adaptive filtering of an unknown time-varying system using an RLS transversal filter of length \( N \). As a result of the nonstationarity of the environment, the optimum solution will be an \( N \)-dimensional vector dependent on the discrete-time index \( n \), \( C_{\text{opt}}(n) \), reflecting variations of the autocorrelation matrix and cross-correlation vector, respectively.

The degree of mathematical difficulty in analyzing nonstationary cases is directly related to the particular application. In a general situation such as channel equalization, both the autocorrelation matrix \( R(n) \) and the cross-correlation vector \( V(n) \) will be functions of time. In the system identification case, however, only the cross-correlation vector will depend on time.

For an RLS transversal filter, the adaptation equation (2.7) yields

\[
C(n) = C(n-1) + K(n) e(n)^* 
\]

\[
= C(n-1) + \hat{R}(n)^{-1} X(n) X(n)^* (C_{\text{opt}}(n-1) - C(n-1)) + K(n) e_{\text{opt}}(n)^* \quad (4.1) 
\]

where we have defined

\[
d(n) = C_{\text{opt}}(n-1)^* X(n) + e_{\text{opt}}(n) \quad (4.2) 
\]

and \( e_{\text{opt}}(n) \) represents the minimum (in a mean square sense) error at time \( n \). In studying tracking behavior, we may exclude the influence of the estimation noise, since the deviation of \( E\{C(n)\} \) from \( C_{\text{opt}}(n) \) determines the response of the adaptive algorithm to the nonstationarities of the environment. Taking expected values on both sides of (4.1) and assuming that \( \hat{R}(n) \) is independent of \( X(n) \) and \( e_{\text{opt}}(n) \), also using the fact [19] that \( X(n) \) and \( e_{\text{opt}}(n) \) are orthogonal, we get

\[
E\{C(n)\} = E\{C(n-1)\} + E\{\hat{R}(n)^{-1} X(n) X(n)^*\} \cdot (C_{\text{opt}}(n-1) - E\{C(n-1)\}). \quad (4.3)
\]

In Appendix A-3 we show that if the variations of the environment are slow with respect to the memory of the adaptive algorithm then, independently of the adaptive system structure (i.e., equalizer or channel estimator), we approximately have

\[
E\{\hat{R}(n)^{-1} X(n) X(n)^*\} = (1 - \lambda) I_{NN}. \quad (4.4)
\]

Substituting (4.4) into (4.3), we finally get

\[
E\{C(n)\} = E\{C(n-1)\} + (1 - \lambda) \cdot (C_{\text{opt}}(n-1) - E\{C(n-1)\}). \quad (4.5)
\]

If \( \Delta(C) \) is the \( z \) transform of the lag error vector \( \Delta(C(n)) = E\{C(n)\} - C_{\text{opt}}(n) \), then

\[
\Delta(C) = \frac{z^{-1} - 1}{1 - 1/z^2} C_{\text{opt}}(z). \quad (4.6)
\]

The above transfer function has a zero at \( z = 1 \) and a pole at \( z = \lambda \). Thus, under general time-varying conditions, the lag error vector is a geometric converging process. The time constant of the process given by

\[
\tau_i = \frac{1}{1 - \lambda} \quad (4.7)
\]

is the same for all coordinates \( i \), \( 1 \leq i \leq N \). It is also independent of the spread of the eigenvalues of the autocorrelation matrix. This is in contrast to the LMS algorithm where the time constant is, in general, different for each component and corresponds to a particular eigenvalue of \( R(n) \) [5]. The independence of the geometrically converging process from the eigenvalue distribution causes the tracking ability of the RLS algorithms to be always at least as good as that of the LMS algorithm.

B. Calculation of the "Excess Lag" MSE

In this section we are going to derive an expression for the "excess lag" MSE concerning a particular application, namely, channel estimation.

The model for the unknown system to be used in this investigation is a tapped delay line filter structure with time-variant coefficients and length \( N \). It is assumed, furthermore, that the unknown coefficients obey a first-order Markov difference equation, i.e.,

\[
C_{\text{opt}}(n) = \alpha C_{\text{opt}}(n-1) + W(n) \quad (4.8)
\]

where \( |\alpha| < 1 \) and \( W(n) \) is a zero mean Gaussian random vector with covariance matrix

\[
E\{W(k) W(l)^*\} = \delta_{kl} \quad (4.9)
\]

for all \( k, l \) where \( \delta_{ij} \) is the Kronecker delta, which is 1 for \( k = l \) and 0 otherwise. Such a model has been used in [5] and [8] for the study of the nonstationary characteristics of the LMS and orthogonalized LMS algorithm, respectively.

Clearly, a common region of operation for the adaptive algorithm will be \( 1/(1 - \alpha) \gg 1/(1 - \lambda) \). Therefore, in cases of interest, \( \alpha \) will be very close to but less than 1. Using this assumption along with (4.6), (4.8), and (4.9), the covariance matrix \( \text{cov} \{\Delta(C(n))\} \) of the lag error vector can be calculated as

\[
\text{cov} \{\Delta(C(n))\} = \sum_{j=0}^{\infty} (\lambda^2)^j \sigma_w^2 I_{NN} = \frac{1}{1 - \lambda^2} \sigma_w^2 I_{NN}. \quad (4.10)
\]

We observe that the lag error covariance matrix, and subsequently the excess lag MSE \( \epsilon_{\text{lag}} \), are geometrical converging processes with time constant \( 1/(1 - \lambda^2) \). Fur-
thermore, since in most practical situations $\lambda$ usually takes values no lower than 0.9, we can use the approximation

$$\frac{1}{1 - \lambda^2} \approx \frac{1}{2(1 - \lambda)} \quad (4.11)$$

Finally, using the fact that

$$\epsilon_{\text{lag}} = E\{\Delta C(n)^t R \Delta C(n)\} = E\{\text{tr} (R \Delta C(n) \cdot \Delta C(n)^t)\} \quad (4.12)$$

where $\text{tr} (\cdot)$ stands for the trace of the matrix in the brackets, we get

$$\epsilon_{\text{lag}} = \frac{\sigma_w^2}{2(1 - \lambda)} \text{tr} (R). \quad (4.13)$$

The expression given by (4.13) indicates that the excess MSE due to the lag, $\epsilon_{\text{lag}}$, is directly related to the variance of the source of nonstationarity, to the power of the input signal, and to the memory of the RLS algorithm as well. As $\lambda$ approaches 1, the lag error approaches infinity. The extreme situation of $\lambda = 1$ corresponds to an equal weighting of all past information in calculating an updated coefficient vector. For $\lambda < 1$, the past is attenuated geometrically, and therefore, the algorithm can track variations of the input statistics.

The total excess MSE is the sum of the excess MSE due to estimation noise and due to lag. Combining (3.20) and (4.13), we get

$$\epsilon_{\text{tot}} = \frac{1 - \lambda}{1 + \lambda} \epsilon_{\text{opt}} + \frac{1}{2(1 - \lambda)} \text{tr} (R) \sigma_w^2 \quad (4.14)$$

The formula derived for the total excess MSE can be optimized with respect to the parameter $\lambda$. Differentiating (4.14) and setting the result equal to zero, also noticing that

$$Na_i^2 = \text{tr} (R), \quad (4.15)$$

we get

$$\lambda_{\text{opt}} = \frac{1 - \beta}{1 + \beta} \quad (4.16)$$

where

$$\beta = \sqrt{\frac{\sigma_2^2}{4\epsilon_{\text{opt}}}} \quad (4.17)$$

with the constraint $\beta < 1$. It is interesting to note that the optimum $\lambda_{\text{opt}}$ is independent of the length of the adaptive filter. It depends on the statistics of the input signal, on the statistics of the nonstationarity driving noise, and on the minimum MSE as well. In Fig. 2, the total excess MSE has been plotted for a hypothetical RLS adaptive filter of length $N = 10$ and $\epsilon_{\text{opt}} = 10^{-3}$. The two curves shown in Fig. 2 correspond to two different levels of $\sigma_w^2$, i.e., $\sigma_w^2 = 10^{-5}$ and $\sigma_w^2 = 10^{-6}$. It appears that the higher the level of the $\sigma_w^2$ is, the smaller the $\lambda_{\text{opt}}$ should be.

It is of great importance to compare the above results to those derived by Widrow et al. for the LMS algorithm in [5]. We recall that for the LMS adaptive filter, the time constant $\tau_{LMS}$ and the total excess MSE are given by

$$\tau_{LMS} = \frac{1}{2\rho_i} \quad 1 \leq i \leq N \quad (4.18)$$

and

$$\epsilon_{\text{tot},LMS} = \mu \text{tr} (R) \epsilon_{\text{opt}} + \frac{Na_i^2}{4\mu}, \quad (4.19)$$

respectively, where $\mu$ is the adaptation step size and $\rho_i$ is the eigenvalue corresponding to the $i$th component of the tap-coefficient vector. We observe that if all eigenvalues of $R$ are equal and $\text{tr} (R) = N$, then by choosing $\mu = (1 - \lambda)/2$, the time constants and the total excess MSE corresponding to the LMS and RLS adaptive algorithm become equal. Hence, one would expect the steady-state performance of the two algorithms for white inputs to be exactly the same at least when the unknown coefficients obey a first-order Markov stochastic difference equation. In many cases, the value of $\mu$, which predicts similar tracking behavior with that of the RLS algorithm, might exceed the critical value for a stable operation. For an LMS adaptive algorithm with white inputs such that $\text{tr} (R) = N$, it can be shown [16], [28] that the step size $\mu$ must always satisfy the following inequalities:

$$0 < \mu < \frac{1}{N}. \quad (4.20)$$

For example, in the hypothetical situation we discussed previously, we have seen that if $\sigma_w^2 = 10^{-5}$, then $\lambda_{\text{opt}} = 0.904$. This value suggests $\mu = 0.048$. If an adaptive filter with length $N \geq 21$ is to be used, then obviously the optimum value of $\mu$ which gives same tracking performance violates the stability bound established in (4.20). When the maximum stable value of the step size is inserted instead, then there is a significant tracking advantage of the RLS algorithms. Thus, for relatively long fil-

\[\text{Equality holds also for the covariance of the lag error vector} \quad (\Delta C(n))_{\text{of the two algorithms.}}\]
ters used in time-varying environments, RLS algorithms yield superior tracking performance. However, for fairly short system-identification filters, where the LMS step size $\mu$ which guarantees similar tracking behavior does not violate the stability bound of (4.20), there is no tracking advantage in using the more complex RLS algorithm.

V. Finite Precision Arithmetic Considerations

A. Steady-State Performance

A crucial factor in addition to the tracking ability and complexity of an adaptive filter algorithm is its numerical stability. Numerical instabilities or finite precision problems pose potential problems for any recursive least squares adaptation algorithm transversal or lattice, which minimizes (2.3); since according to (2.3) this minimization explicitly or implicitly forms and inverts an $N \times N$ matrix whose elements are found by accumulating weighted sums of all previous filter inputs. Previous simulations have reported that the transversal FRLS adaptation algorithm tend to become unstable (the coefficients "blew up"), especially for $\lambda$ less than unity (fast adaptation) [10], [12]. The same phenomenon has also been observed with the much more computation-intensive RLS algorithm. The most stable reported recursive least squares transversal adaptive filter is the "square-root Kalman" filter [22], but even in this case, it was considered necessary to reinitialize certain variables of the algorithm about once every 100 symbol intervals to avoid instabilities due to roundoff errors. It is worth noting that the relatively slow converging simple LMS adaptive transversal filter does not fail prey to numerical stability resulting from roundoff error. However, digital implementation of the LMS as a fractionally spaced equalizer must be carefully designed to avoid problems due to bias in some types of digital arithmetic [23].

Among lattice adaptive filters, the FRLS lattice has been found to exhibit better numerical stability than its FRLS transversal counterpart [12]. Still more stable is a normalized version of the FRLS lattice [24], which unfortunately is more computation intensive, requiring square roots as well as multiplications and divisions. We should emphasize that adaptive lattice algorithms all require many more divisions than do the transversal filter algorithms; thus, their complexity if implemented with most existing or contemplated digital signal processing devices is even greater.

It is unfortunate that the FRLS or "fast Kalman" adaptation algorithm, which is the least computation intensive of all the equally fast converging RLS and FRLS algorithms is also the most numerically unstable. A recent study of this type of algorithm suggests modifications to it which may improve its numerical stability and even decrease its complexity somewhat [10].

Our remedy for numerical stability problems of FRLS transversal filter is periodic reinitialization. We describe new restart procedures which guarantee smooth reinitialization of the adaptive algorithm without introducing any significant performance-degrading transients. These restart procedures appear to make the FRLS transversal filter algorithm viable for adaptive equalization or channel estimation of time-varying channels. The new restart procedures have some essential differences from those reported in [9] and [10]. We shall illuminate these differences in the next paragraph.

B. Periodic Restart for FRLS Algorithms

To avoid buildup of roundoff errors with time, we propose interrupting and restarting the FRLS with critical internal variables cleared to zero, at periodic intervals, say, every $M$ symbol intervals. Immediately following such a restart at time $M$, a simple LMS adaptation algorithm, which has been initialized with the tap-coefficient vector $C(M - 1)$ that the FRLS algorithm has obtained just before the restart was initiated, provides the filter output temporarily until the restarted FRLS takes over again after a short time.

FRLS methods have been developed which, aided by the LMS algorithm's tap coefficients, converge faster during restart than the "unaided" FRLS algorithm does during initial startup at $n = 0$. The transition period is so short ($N-1.5N$ time units) and smooth that the momentary reliance on the slower adapting LMS algorithm causes little or no performance degradation.

It can be shown [10] that following a restart at time $M$, the adaptive FRLS algorithm minimizes for $n \geq M$

\[
J(n) = \sum_{k=M}^{n} \lambda^{n-k+M} |d(k) - C(n)^* X(k)|^2 + \delta \lambda^{n-M} (C(n) - C(M))^* A_n (C(n) - C(M))^4
\]

(5.1)

where $C(M)$ is the value of the tap coefficients updated just prior to restart, and $\delta(\delta > 0)$ is a factor used to weight the effect of the initial condition upon the cost criterion. In that sense, $C(M)$ can be set to zero or not. All other relevant internal variables will be initialized as during the startup at time $n = 0$. The input vector $\hat{x}(n)$ seen by the FRLS algorithm for adaptation purposes will have components

\[
\hat{x}(n) = \begin{cases} x(n), & n > M \\ 0, & n \leq M. \end{cases}
\]

(5.2)

The key idea behind the restarting methods is to remove from the desired response sequence $\{d(n)\}$ the influence of those components of the input vector $\hat{x}(n)$ which are not available to the restarted FRLS. This modification on the desired response will mitigate the effect of the truncation of the input vector $\hat{x}(n)$ on the tracking ability of the algorithm. This is the major feature distinguishing our restart procedures from those of [9] and [10].

To see how the $\{d(n)\}$ should be modified, we note that

$A_n$ is the $N \times N$ diagonal matrix $A_n = \text{Diag}(1, \lambda^{-1}, \lambda^{-2}, \ldots, \lambda^{-N+1})$. 


if there has been no restart just prior to time \( nT \), then we could express \( d(n) \) as

\[
d(n) = C(n)'* X(n) + u(n)
\]

(5.3)

where \( u(n) \) is an error due to noise and random data, and where \( C(n)' \) is the value \( C(n) \) would have in the absence of restart. Now define the \( N \)-dimensional vector

\[
X(n)^{-+} = X(n) - \hat{X}(n),
\]

(5.4)

i.e., \( X(n)^{-+} \) contains only the components of \( X(n) \) which occurred prior to \( n = M \). Thus,

\[
d(n) = C(n)'* X(n)^{-+} = C(n)'* \hat{X}(n) + u(n).
\]

(5.5)

Since the restarted FRLS algorithm only has the vectors \( \hat{X}(n) \) available, and since the adapted tap-coefficient vector \( C(n) \) should ideally approach \( C(n)' \), (5.5) suggests that during the restart period, the desired outputs used for FRLS adaptation should be

\[
\hat{d}(n) = d(n) - C_{LMS}(n)^* X(n)^{-+}.
\]

(5.6)

where \( C_{LMS}(n) \) is the current set of tap coefficients from the auxiliary LMS algorithm. Note that when \( n \) exceeds \( M \) by a number of symbol intervals equal to the filter's memory, \( X(n)^{-+} \) is zero and \( d(n) = \hat{d}(n) \). Another type of modification also suggested by (5.5) is the following:

\[
\hat{d}(n) = \begin{cases} 
C_{LMS}(n)^* \hat{X}(n) & \text{during restart interval} \\
\hat{d}(n) & \text{after restart interval.}
\end{cases}
\]

(5.7)

The two suggested modifications on the desired response, along with the fact that \( C(n) \) can be set to zero or not at the initialization of the restart, give four possible variations of the periodic restart procedures. These procedures are based on the modified desired response methods we described previously, and all rely on an auxiliary LMS adaptation algorithm.

**Restart Procedure 1:** The FRLS adaptive filter tap coefficients \( C(n) \) were not reinitialized at the initiation of each restart \( n = M \). The desired filter output for this procedure was that of (5.6).

**Restart Procedure 2:** In this restart procedure, the adaptive filter's tap-coefficient vector \( C(n) \) was set to zero at the initialization of the restart. Again, (5.6) was used to provide the desired response.

**Restart Procedure 3:** Like procedure 1, \( C(n) \) were not reinitialized, but (5.7) was used for the desired output.

**Restart Procedure 4:** In this procedure, \( C(n) \) were set to all zeros. Equation (5.7) provides the desired response.

As suggested in [9], another way to continue the FRLS algorithms is to use the unwinned versions which do not assume that \( X(n) \) is zero for negative time arguments, since \( X(n) \) is not zero when restarting in steady-state operation. In such a case, modification of the desired response \( d(n) \) is not necessary. Unfortunately, unwinned algorithms are more computationally intensive than the exponential windowed ones requiring approximately \( 3N \) more multiplications per iteration.

### VI. Simulation Results

The theory developed in the previous sections has been tested through computer simulations. The adaptive filters used for this investigation were driven by FRLS transversal algorithms as reformulated in [10].

For the evaluation of the estimation-noise formula, a DFE (decision feedback equalizer) was implemented. The second part of our work concerning the lag effects has been tested by running a channel estimator. In both cases, double precision on a 36-bit Honeywell Level 66 Computer (28-bit mantissa) has been used in order to delay, as much as possible, the onset of numerical instabilities.

Finally, the restarting methods have been tested in specific applications such as decision feedback equalization, using real recorded HF data [26].

#### A. Estimation-Noise

In this simulation, we used an FRLS DFE with \( N_1 = 8 \) forward and \( N_2 = 2 \) feedback taps. The number of coefficients in the finite impulse response discrete channel is three with a maximum-to-minimum eigenvalue ratio of \( p_{\text{max}}/p_{\text{min}} = 56 \). Binary signaling was also assumed, i.e., \( d(n) = \pm 1 \), while the level of the variance of the zero-mean additive white Gaussian noise was fixed to \( \sigma_n^2 = 0.01 \). For the given channel and signal statistics, the statistical parameter \( \gamma \) has been computed as \( \gamma = 0.8 \).

Table I shows the results of our investigation comparing theoretical and experimental values as a function of the exponential windowing factor \( \lambda \). The first column corresponds to the misadjustment \( M \) estimated over a sample of 1600 iterations. Measurements were taken only after initial transients had faded away. Second and third columns correspond to the predicted theoretical values by using the new formula (3.18) and the approximate one (3.21), respectively.

Fig. 3 illustrates the same results in a decibel scale. The two solid curves represent theoretical values, while the dashed curve represents experimental ones. The small vertical lines on top of each experimental point indicate the minimum and maximum range of variation of the estimated misadjustment \( M \) over a longer sample of 4000 iterations.

The above simulation results indicate that the experimental misadjustment values are closer to the theoretical ones predicted by the new formula (3.18). This conclusion is also in agreement with the experimental results reported in [11] for similar kinds of simulation.

#### B. Lag Effects

In the second part of our investigation, we simulated an FRLS transversal channel estimator. The number of tap coefficients used for the unknown time-varying systems and the adaptive filter was \( N = 3 \). As input signal, we used a white sequence taking values \( \pm 1 \) so that \( R = I_{33} \). The additive noise variance was fixed to \( \sigma_n^2 = c_{\text{opt}} = 0.01 \).

Figs. 4 and 5 summarize the results of the simulations, comparing theory and experiments for different values of...
TABLE I

<table>
<thead>
<tr>
<th>( \lambda )</th>
<th>( \hat{M} )</th>
<th>( M, (3.18) )</th>
<th>( M, (3.21) )</th>
</tr>
</thead>
<tbody>
<tr>
<td>0.99</td>
<td>0.059</td>
<td>0.0504</td>
<td>0.0503</td>
</tr>
<tr>
<td>0.98</td>
<td>0.109</td>
<td>0.1018</td>
<td>0.1010</td>
</tr>
<tr>
<td>0.97</td>
<td>0.164</td>
<td>0.1541</td>
<td>0.1523</td>
</tr>
<tr>
<td>0.96</td>
<td>0.220</td>
<td>0.2074</td>
<td>0.2041</td>
</tr>
<tr>
<td>0.95</td>
<td>0.279</td>
<td>0.2616</td>
<td>0.2564</td>
</tr>
<tr>
<td>0.94</td>
<td>0.340</td>
<td>0.3169</td>
<td>0.3093</td>
</tr>
<tr>
<td>0.93</td>
<td>0.403</td>
<td>0.3739</td>
<td>0.3630</td>
</tr>
<tr>
<td>0.92</td>
<td>0.473</td>
<td>0.4305</td>
<td>0.4170</td>
</tr>
</tbody>
</table>

Fig. 3. Misadjustment versus \( \lambda \) for a decision feedback equalizer.

Fig. 4. SNR versus \( \lambda \) for \( \sigma_r^2 = 10^{-2} \).

Fig. 5. SNR versus \( \lambda \) for \( \sigma_r^2 = 10^{-4} \).

Fig. 6. SNR versus time.

the nonstationarity parameters \( \lambda \) and \( \sigma_r^2 \). They illustrate the signal-to-noise ratio as a function of the adaptation parameter \( \lambda \). The SNR on the vertical axis is the ratio of 1 (mean squared input sample) to the observed mean square value or the ensemble mean squared value of the error \( e(n) \) between the output of the adaptive filter and the desired response.

Experimental values on those figures correspond to statistical time averages taken over 2000 iterations in two different ways. Points on the dashed lines indicate time averaging with equal weighting of all past information, while points on the dotted line indicate exponential weighting of the past information. The solid curves in all these figures represent theoretical results as have been predicted by using (4.14).

Experimental results came very close to what was theoretically expected with the exception of the case when \( \lambda \) approaches \( \alpha \). However, this outcome is not surprising since when \( \lambda \) approaches \( \alpha \), \( 1/(1 - \alpha) \) is not significantly greater than \( 1/(1 - \lambda) \), and therefore, (4.13) is inaccurate.

Fig. 6 shows the evolution of the SNR with time. The solid straight line represents the theoretical predicted
C. Restarting Procedures

The different restarting methods have been tested by implementing a decision feedback equalization (DFE) receiver. For this investigation we used complex baseband 2.4 kbits/s QPSK demodulated signal recorded from HF channels [25]. Preliminary simulations established the choice of the number of equalizer tap coefficients: \( N_1 = 20 \) \( T/2 \)-spaced forward tap coefficients and \( N_2 = 7 \) feedback tap coefficients, where \( T \) is the duration of one symbol. The exponential weighting factor and the auxiliary LMS adaptation step size were set to \( \lambda = 0.97 \) and \( \mu = 0.035 \), respectively.

The four variations of the periodic restart procedure of the previous section for the FRLS algorithms were tried. All relied on an auxiliary LMS adaptation algorithm to provide the equalizer output during each restart interval.

Fig. 7 shows an example of the application of restart methods initiated at \( n = 300 \)th symbol intervals for HF channel designated MDAO10. The SNR on the vertical axis is the ratio of 1 (the mean squared data symbol) to the observed mean squared value of the error \( e(n) \) between the decision-making quantizer's input and its output (decision-directed mode). The averaging of the mean squared error (MSE) at the \( n \)th interval is as follows:

\[
\text{MSE}(n) = 0.9 \text{MSE}(n-1) + 0.1|e(n)|^2 \quad (6.1)
\]

and the SNR at this time is

\[
\text{SNR}(n) = \frac{1}{\text{MSE}(n)} \quad (\text{dB}). \quad (6.2)
\]

In the case shown in Fig. 7, the restart interval is 30 symbol intervals, i.e., approximately equal to the length of the adaptive filter. During this period, the useful equalizer output (the input to the decision-making quantizer) is taken from an auxiliary DFE driven by an LMS algorithm, whose tap coefficients were initialized to the tap coefficients of the FRLS DFE at the beginning of the restart period. In Fig. 7, the SNR obtained from this LMS filter is shown by the solid line between \( n = 300 \) and \( n = 330 \). During the 30-symbol interval restart interval, the FRLS algorithm adapted from an initial state restarting at \( n = 300 \). After \( n = 330 \), the FRLS-adapting DFE took over from LMS-adapting DFE. Fig. 7 shows the SNR for the four restart procedures beyond \( n = 330 \). In this example, restart procedure 1 gave slightly better performance than restart procedure 2, which in turn was somewhat better than procedure 3 and 4. At this point, we should emphasize that the SNR's big dip just after the FRLS DFE took over from the auxiliary LMS DFE (approximately at \( n = 335 \)) is irrelevant to the restarting procedure, and it has been caused by a sudden fade. This is clear in Fig. 8 where the restart interval has been increased to 40 data symbol intervals, and thus, the SNR at \( n = 335 \) has been calculated using the output of the auxiliary LMS DFE.

The differences between these restart methods in the same example are emphasized in Fig. 9 which shows the measured SNR between \( n = 300 \) and \( n = 330 \), where the SNR is now defined as the reciprocal of the MSE, defined as in (6.2), for the still-restarting FRLS algorithm instead of the LMS algorithm. The MSE was initialized to zero at \( n = 300 \), which accounts for the large values of SNR near \( n = 300 \). It is clear that restart procedures 1 and 3,
in which the equalizer tap coefficient were not zeroed, caused little disruption to the FRLS adaptation during the restart period, while the procedures involving zeroed coefficients required at least 40 iterations to restore a reasonably low mean square error.

On the basis of simulation results like these, we conclude that there is no noticeable disruption as a result of the periodically initiated restarts.

VII. Conclusions

In this paper we have been concerned with the steady-state performance characteristics of the RLS adaptive algorithms. We have examined quantitatively the estimation-noise inherent in all exponential windowed RLS algorithms as well as their tracking ability in a time-varying environment. We have also studied the steady-state performance limitations due to roundoff noise effects.

We have shown that the level of the estimation-noise depends not only on the nature of the "windowing," but also on some fourth-order statistics of the input signal. The latter is more noticeable in fast adaptation where \( \lambda \) may take values as low as 0.9.

Additionally, it has been shown that the mechanism associated with the tracking ability of the RLS is a geometrical converging process. Furthermore, independent of the kind of the nonstationary model and the adaptive filter application (e.g., equalizer or channel estimator), the time constant of this process is the same for all components of the tap-coefficient vector and equal to the memory of the algorithm. It is also independent of the eigenvalues of the autocorrelation matrix. In that sense, the tracking ability of the RLS algorithms is superior than that of the LMS algorithm for the case of disparate eigenvalues.

Finally, assuming a first-order Markov time-varying model, the excess lag MSE for a channel estimator has been calculated. For this simple nonstationary situation where there is direct control on the eigenvalues of the constant autocorrelation matrix (all eigenvalues can be equal), it was shown that the simple LMS tracks as well as any of the more complex RLS algorithms, provided that the judiciously chosen step size does not violate the stability bound. Tradeoffs involved in choosing the optimum exponential weighting parameter \( \lambda_{opt} \) in order to obtain simultaneously acceptable low excess error and tracking ability are clearly illustrated in (4.16).

In the last part of our work we have been concerned with the problem of numerical instability due to roundoff noise in FRLS transversal algorithms. By restarting (zeroing) only the internal variables and modifying the desired adaptive filter outputs during the restart to account for the step function in the filter inputs during restart disruptions due to restart were minimized. The results indicate that by applying any one of these efficient restart procedures, we can circumvent the numerical instability problem to which FRLS algorithms are subject, without significantly compromising its ability to track rapidly time-varying channels.

**APPENDIX A-1**

**Derivation of (3.13):** We have already shown that the tap-coefficient error satisfies the following relation:

\[
\Delta c(n) = \lambda \hat{r}(n)^{-1} \hat{r}(n-1) \Delta c(n-1) + \hat{r}(n)^{-1} x(n) e_{opt}(n)^*.
\]

Therefore,

\[
E\{\Delta c(n)^* R \Delta c(n)\}
\]

\[
= \lambda^2 E\{\Delta c(n-1)^* \hat{r}(n-1) \hat{r}(n)^{-1} R \hat{r}(n)^{-1} \}
\]

\[
+ \lambda E\{\Delta c(n-1)^* \hat{r}(n-1)^{-1} R \hat{r}(n)^{-1} X(n) e_{opt}(n)^*\}
\]

\[
+ \lambda E\{e_{opt}(n) X(n)^* \hat{r}(n-1)^{-1} R \hat{r}(n)^{-1} \}
\]

\[
+ \Delta(c(n-1)) + E\{X(n)^* \hat{r}(n-1)^{-1} R \hat{r}(n)^{-1} X(n) | e_{opt}(n)|^2\}.
\]

(A1.2)

At the steady state, we can assume that \( \hat{r}(n) \) and \( \hat{r}(n-1) \) are almost equal. Thus, approximately

\[
\hat{r}(n)^{-1} \hat{r}(n-1) = I_{NN} \quad \text{as} \quad n \to \infty.
\]

(A1.3)

In Appendix A-2 we have also shown that, asymptotically, the inverse estimated autocorrelation matrix \( \hat{r}(n)^{-1} \) and the true autocorrelation matrix \( R \) satisfy approximately the following relation:

\[
\hat{r}(n)^{-1} R = (1 - \lambda) [I_{NN} - P(n)]
\]

(A1.4)

where \( P(n) \) is a zero mean fluctuation matrix.

Using (A1.3), (A1.4), and referring also to the independence assumption of \( P(n) \) from \( X(n) \) as well as to the fact that \( E\{P(n)\} = 0 \), the second term in (A1.2) yields

\[
\lambda E\{X(n)^* [Z I_{NN} - P(n)] X(n) e_{opt}(n)^*\}
\]

\[
= \lambda (1 - \lambda) E\{\Delta c(n-1)^* (I - P(n)^* X(n) e_{opt}(n)^*)
\]

\[
+ \lambda (1 - \lambda) E\{\Delta c(n-1)^* e_{opt}(n)^*\}
\]

\[
= \lambda (1 - \lambda) E\{\Delta c(n-1)^*\} E\{X(n) e_{opt}(n)^*\} = 0.
\]

(A1.5)

Note that at the final step in deriving (A1.5), the statistical independence assumption of \( \Delta c(n-1) \) with \( X(n) \), and the orthogonality property [19] which holds between \( X(n) \) and \( e_{opt}(n) \), have been used. We can similarly show that the third term of (A1.2) is zero. Combining (A1.2), (A1.3), (A1.4), and (A1.5), we get

\[
E\{\Delta c(n)^* R \Delta c(n)\} = \lambda^2 E\{\Delta c(n-1)^* R \Delta c(n-1)\}
\]

\[
+ (1 - \lambda)^2 E\{X(n)^* [I - P(n)]^2\}
\]

\[
\cdot R^{-1} X(n) | e_{opt}(n)|^2\}.
\]

(A1.6)

In general, \( e_{opt}(n) \) is uncorrelated with \( X(n) \). If we assume that they are also independent, (A1.6) yields

\[
E\{\Delta c(n)^* R \Delta c(n)\} = \lambda^2 E\{\Delta c(n-1)^* R \Delta c(n-1)\}
\]

\[
+ (1 - \lambda)^2 E\{X(n)^* [I - P(n)]^2\}
\]

\[
\cdot R^{-1} X(n) e_{opt}\}
\]

(A1.7)
where

\[ \epsilon_{\text{opt}} \triangleq E[|\epsilon_{\text{opt}}(n)|^2] \]  \hspace{1cm} (A1.8)

or, in a more compact form,

\[ E\{\Delta C(n)^* \Delta C(n)\} = \lambda^2 E\{\Delta C(n-1)^* \Delta C(n-1)\} \]
\[ \cdot (1 - \lambda^2) \text{trace } I_{\text{NN}} \]
\[ + E\{P(n)^2\} \cdot \epsilon_{\text{opt}}. \]  \hspace{1cm} (A1.9)

**APPENDIX A-2**

**Derivation of (A1.4) and Related Results:** The estimated autocorrelation matrix \( \hat{R}(n) \) has been defined as

\[ \hat{R}(n) = \sum_{k=0}^{n} \lambda^{n-k} X(k) X(k)^* \]  \hspace{1cm} (A2.1)

with elements

\[ \hat{r}_{ij}(n) = \sum_{k=0}^{n} \lambda^{n-k} x(k-i) x(k-j+1) \]
\[ 1 \leq i, j \leq N. \]  \hspace{1cm} (A2.2)

We observe that asymptotically and for stationary input,

\[ E\{\hat{R}(n)\} = \frac{1}{1 - \lambda} E\{X(n) X(n)^*\} = \frac{1}{1 - \lambda} R. \]  \hspace{1cm} (A2.3)

Thus, a reasonable model for \( \hat{R}(n) \) will be

\[ \hat{R}(n) = \frac{1}{1 - \lambda} R + \hat{R}(n) \]  \hspace{1cm} (A2.4)

where \( \hat{R}(n) \) is assumed to be a Hermitian perturbation matrix with entries: the zero mean random variables \( \hat{r}_{ij}(n) \), \( 1 \leq i, j \leq N \), independent from \( x(n) \). Combining (A2.2), (A2.3), and (A2.4), we get

\[ \hat{r}_{ij}(n) = \lambda \hat{r}_{ij}(n-1) + x(n-i) \]
\[ \cdot x(n-j+1)^* - r_{ij}. \]  \hspace{1cm} (A2.5)

It can easily be seen from (A2.4) that

\[ \hat{R}(n)^{-1} R = (1 - \lambda) [I_{\text{NN}} + (1 - \lambda) R^{-1} \hat{R}(n)]^{-1}. \]  \hspace{1cm} (A2.6)

Assuming that \( \| (1 - \lambda) R^{-1} \hat{R}(n) \| < 1 \) where \( \| \cdot \| \) denotes some kind of consistent vector or matrix norm, we get

\[ \hat{R}(n)^{-1} R \approx (1 - \lambda) [I_{\text{NN}} - (1 - \lambda) R^{-1} \hat{R}(n)]. \]  \hspace{1cm} (A2.7)

The randomness of (A2.7) is embodied in the existence of the perturbation matrix \( \hat{R}(n) \). The zero mean matrix

\[ P(n) \triangleq (1 - \lambda) R^{-1} \hat{R}(n) \]  \hspace{1cm} (A2.8)

manifests the fluctuation of the product \( \hat{R}(n)^{-1} R \) around the unit matrix \( I_{\text{NN}} \). Therefore, it is an expression for the estimation error involved in using exponential estimates for the autocorrelation matrix.

Let \( s_{ij} \) be the entries of the matrix \( S = E\{P(n)^2\} \) and \( r_{ij}\) by the entries of the inverse autocorrelation matrix \( R^{-1} \). Then by using (A2.8), assuming that \( \hat{r}_{ij} \) are uncorrelated for all \( i, j \) and using the fact that \( \hat{R}(n) \) is Hermitian we get

\[ s_{ij} = (1 - \lambda)^2 \left[ \sum_{k=1}^{N} r_{ik} E\{|\hat{r}_{kj}(n)|^2\} \right] r_{kj} \]
\[ + \sum_{k=1}^{N} r_{ik} E\{|\hat{r}_{kj}(n)|^2\} r_{kj}. \]  \hspace{1cm} (A2.9)

Referring to the independence assumption between \( x(n) \) and \( \hat{r}_{ij}(n) \), (A2.5) yields

\[ \text{var} \{\hat{r}_{ij}(n)\} = \lambda^2 \text{var} \{\hat{r}_{ij}(n-1)\} + \text{var} \{x(n-i) \]
\[ \cdot x(n-j+1)^*\}. \]  \hspace{1cm} (A2.10)

Similarly,

\[ E\{\hat{r}_{ij}(n)^2\} = \lambda^2 E\{\hat{r}_{ij}(n-1)^2\} + E\{x(n-i) \]
\[ \cdot x(n-j+1)^* - \hat{r}_{ij}^2\}. \]  \hspace{1cm} (A2.11)

Thus, asymptotically,

\[ E\{|\hat{r}_{ij}(n)|^2\} = \text{var} \{\hat{r}_{ij}(n)\} \]
\[ = \frac{1}{1 - \lambda^2} \text{var} \{x(n-i) \]
\[ \cdot x(n-j+1)^*\}. \]  \hspace{1cm} (A2.12)

and

\[ E\{\hat{r}_{ij}(n)^2\} = \frac{1}{1 - \lambda^2} E\{x(n-i) \]
\[ \cdot x(n-j+1)^* - \hat{r}_{ij}^2\}. \]  \hspace{1cm} (A2.13)

Using (A2.12) and (A2.13) in (A2.9), we finally get

\[ s_{ij} = \left[ \frac{1 - \lambda}{1 + \lambda} \sum_{k=1}^{N} r_{ik} \text{var} \{x(n-k+1) \]
\[ \cdot x(n-j+1)^*\} r_{kj} \]
\[ + \sum_{k=1}^{N} r_{ik} E\{x(n-k+1) \]
\[ \cdot x(n-j+1)^* - \hat{r}_{ij}^2\} r_{kj} \]. \]  \hspace{1cm} (A2.14)

Observe that for a given channel and input signal statistics, the elements of \( S \) can explicitly be determined.

**APPENDIX A-3**

**Derivation of (4.4):** In the general case of nonstationary inputs, the autocorrelation matrix of the input vector \( X(n) \) will be function of the discrete time index \( n \), i.e.,

\[ R(n) = E\{X(n) X(n)^*\}. \]  \hspace{1cm} (A3.1)

Then, according to (A2.4), we can write

\[ \hat{R}(n) = \hat{R}(n) + \hat{R}(n) \]  \hspace{1cm} (A3.2)

or

\[ \hat{R}(n) = \sum_{k=1}^{n} \lambda^{n-k} R(k) + \hat{R}(n). \]  \hspace{1cm} (A3.3)

If the variations of the environment are slow with respect
to the memory of the adaptive algorithm then, asymptotically,
\[ \hat{R}(n) = \frac{1}{1 - \lambda} R(n) + \hat{R}(n). \]  
(A3.4)

Note that in a system identification situation, the above expression is valid with strict equality since, in such a case, the autocorrelation matrix does not depend on the time. Assuming that \( \| (1 - \lambda) R(n)^{-1} \hat{R}(n) \| < 1 \) where \( \| \cdot \| \) denotes some kind of consistent norm, it can be shown easily that
\[
E\{ \hat{R}(n)^{-1} X(n) X(n)^* \} = E\left\{ \left( \frac{1}{1 - \lambda} R(n) + \hat{R}(n) \right)^{-1} X(n) X(n)^* \right\}
\]
\[
= (1 - \lambda) E\{ R(n)^{-1} X(n) X(n)^* \} - (1 - \lambda)^2 E\{ R(n)^{-1} \hat{R}(n) R(n)^{-1} X(n) X(n)^* \}.
\]  
(A3.5)

Finally, using the independence assumption of \( \hat{R}(n) \) from \( X(n) \) along with the fact that \( E\{ \hat{R}(n) \} = 0 \), we get
\[
E\{ \hat{R}(n)^{-1} X(n) X(n)^* \} = (1 - \lambda) I_{m}. \]  
(A3.6)

ACKNOWLEDGMENT

We are grateful to J. G. Proakis and J. M. Cioffi for sending us prepublication versions of [11] and of [10], respectively, and also to J. M. Cioffi and R. Tront for very useful discussions and comments.

REFERENCES


Evangelos Eleftheriou was born in Aliveri, Evvoia, Greece, on February 15, 1955. He received the Diploma in electrical engineering from the University of Patras, Greece, in 1979 and the M. Eng. and Ph.D. degrees from Carleton University, Ottawa, Ont., Canada, in 1981 and 1985, respectively, all in electrical engineering. He joined IBM Zurich Research Laboratory, Rüschlikon, Switzerland, in 1986. His research interests include digital communications, adaptive filtering, and signal processing.

Dr. Eleftheriou is a member of the National Technical Chamber of Greece.

David D. Falconer (S'62-M'68-SM'83-F'86) was born in Moose Jaw, Sask., Canada, on August 15, 1940. He received the B.A.Sc. degree in engineering physics from the University of Toronto, Toronto, Ont., Canada, in 1962 and the S.M. and Ph.D. degrees in electrical engineering from the Massachusetts Institute of Technology, Cambridge, in 1963 and 1967, respectively.

After one year as a Postdoctoral Fellow at the Royal Institute of Technology, Stockholm, Sweden, he was with Bell Laboratories, Holmdel, NJ,
from 1967 to 1980, as a member of the Technical Staff and later as Group Supervisor. During 1976–1977 he was a Visiting Professor at Linkoping University, Linkoping, Sweden. Since 1980 he has been at Carleton University, Ottawa, Ont., Canada, where he is a Professor in the Department of Systems and Computer Engineering. His interests are in digital communications, signal processing, and communication theory.

Dr. Falconer is Editor for Digital Communications for the IEEE Transactions on Communications and is a member of the Communication Theory Technical Committee. He is also a member of the Association of Professional Engineers of Ontario, Sigma Xi, and Tau Beta Pi. He was awarded the Communications Society Prize Paper Award in Communications Circuits and Techniques in 1983 and 1986.