Convergence Properties of an Adaptive Digital Lattice Filter

MICHAEL L. HONIG AND DAVID G. MESSERSCHMITT, SENIOR MEMBER, IEEE

Abstract—Convergence properties of a continuously adaptive digital lattice filter used as a linear predictor are investigated for both an unnormalized and a normalized gradient adaptation algorithm. The PARCOR coefficient mean values and the output mean-square error (MSE) are approximated and a simple model is described which approximates these quantities as functions of time. Calculated curves using this model are compared with simulation results. Results obtained for a two-stage lattice are then compared with the two-stage least mean-square (LMS) transversal filter algorithm, demonstrating that it is possible but unlikely for the transversal filter to converge faster than the analogous lattice filter.

I. Introduction

The ADAPTIVE digital lattice filter has recently received much attention in the contexts of channel equalization, where it can effectively compensate for linear channel distortion [3], and in LPC speech processing, where it can be used as a linear predictor [13], [14]. When used as a linear predictor, the lattice coefficients, known as partial correlation (PARCOR) or "reflection" coefficients [14], can be adapted to minimize the output mean-squared prediction error either by processing blocks of data or continuously using either a least mean-square (LMS) gradient algorithm or a recursive version of the least squares (LS) block processing method [9], [10].

When compared to the simpler adaptive transversal filter, the lattice filter appears to have superior convergence properties and reduced sensitivity to finite wordlength effects [15], [16]. Simulation studies have shown that the lattice gradient algorithm converges substantially faster than the comparable transversal algorithm [3], [5]. No analytical studies of the convergence properties of the adaptive lattice have appeared, however, presumably due to the highly nonlinear nature of the adaptation.

This paper presents a first attempt at quantitative understanding of the behavior of a lattice linear predictor using an LMS gradient adaptation algorithm. A number of simplifying assumptions are made in order to obtain simple results which give insight into the convergence process. The culmination of this effort is a simple model for lattice convergence which predicts the mean value trajectories of the PARCOR coefficients and the output mean-squared error (MSE) in a multistage adaptive lattice filter. Simulations show that the model gives reasonably accurate results.

In Section II the lattice structure and gradient algorithms are discussed, in Section III the convergence behavior of one stage of the adaptive lattice is examined, and Section IV presents the model for multistage lattice convergence. In Section V earlier results are utilized to determine the dependence of lattice convergence speed upon the input signal statistics. In particular a two-stage adaptive lattice is analyzed in some detail. Although the general n-stage case is significantly more complicated, the basic ideas used to discuss the two-stage case should carry through. Finally, the two-stage adaptive lattice and two-stage adaptive transversal filter are compared. Although the adaptive lattice filter generally converges faster than the analogous transversal filter, this is not universally the case as is demonstrated by counterexample.

Throughout this paper the filter input random process is assumed to be stationary. The resulting analysis will give insight into adaptation of the filter for a nonstationary input where variations are slow relative to the adaptation speed of the filter.

Manuscript received June 12, 1980; revised October 31, 1980. This work was supported by the National Science Foundation under Contract ENG78-16966.

The authors are with the Department of Electrical Engineering and Computer Science, University of California, Berkeley, CA 94720.

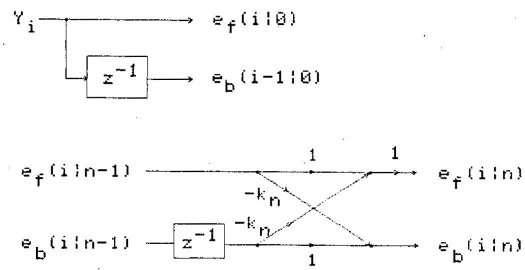


Fig. 1. Lattice filter structure.

II. Lattice Structure

The lattice filter structure shown in Fig. 1 is characterized by the recursive equations

$$e_f(i|n)=e_f(i|n-1)-k_n(i)e_b(i-1|n-1) \quad (2.1a)$$

$$e_b(i|n)=e_b(i-1|n-1)-k_n(i)e_f(i|n-1) \quad (2.1b)$$

where $e_f(i|n)$ and $e_b(i|n)$ are, respectively, the forward and backward prediction errors at the output of the nth stage at the ith sampling interval, $e_b(i|0)=e_f(i|0)=y_i$, a stationary filter input sequence, and $k_n(i)$ is the nth stage PARCOR coefficient at the ith sampling interval. Using (2.1a), the value of $k_n(i)$ which minimizes the mean-squared forward prediction error $E[e_f^2(i|n)]$,

$$k_{n,\text{opt}}(i)=\frac{E\left[e_f(i|n-1)e_b(i-1|n-1)\right]}{E\left[e_b^2(i-1|n-1)\right]} \quad (2.2)$$

0096-3518/81/0600-0642$00.75

where the dependence upon i, the sampling interval, is due to the adaptation of $k_1(i),\cdots,k_{n-1}(i)$ (which cause $e_f(i|n-1)$ and $e_b(i-1|n-1)$ to be nonstationary). Note also that if the coefficients are fixed then

$$E\left[e_f^2(i|n)\right]=E\left[e_b^2(i-1|n)\right], \qquad 1\leqslant n\leqslant N \quad (2.3)$$

where N is the order of the filter.

The adaptive gradient algorithm for the lattice filter is obtained as follows [1]:

$$k_n(i+1)=k_n(i)-\beta_1\frac{\partial e_b^2(i|n)}{\partial k_n(i)}$$

where β_1 is a small adaptation constant. Evaluating the derivative gives

$$k_n(i+1)=\left[1-\beta_1 e_f^2(i|n-1)\right]k_n(i)+\beta_1 e_f(i|n-1)e_b(i-1|n-1). \quad (2.4)$$

A modification of this algorithm attempts to recursively estimate the numerator and denominator of (2.2):

$$C_n(i+1)=(1-\beta_2)C_n(i)+e_f(i|n-1)e_b(i-1|n-1) \quad (2.5a)$$

$$D_n(i+1)=(1-\beta_2)D_n(i)+e_f^2(i|n-1) \quad (2.5b)$$

and

$$k_n(i+1)=\frac{C_n(i+1)}{D_n(i+1)}. \quad (2.5c)$$

It is easily verified that (2.5) can be rewritten as

$$k_n(i+1)=\left[1-\frac{e_f^2(i|n-1)}{D_n(i+1)}\right]k_n(i)+\frac{e_f(i|n-1)e_b(i-1|n-1)}{D_n(i+1)} \quad (2.6)$$

and hence is a normalized version of (2.4). The "unnormalized" and "normalized" gradient algorithms given, respectively, by (2.4) and (2.6) are the only adaptive lattice algorithms considered in this paper. Other gradient types of algorithms have been proposed [2], [4], [5], and they can be analyzed by the same techniques we employ here. In addition the model of convergence presented in Section IV can be extended to apply to the lattice joint process estimator [2] and to recently developed recursive least squares lattice algorithms [10]–[12] (details will be reported in a future paper [17]).

The objective of the adaptive lattice algorithm is to rapidly drive the PARCOR coefficients to the set of values which minimizes the short term mean-squared output. In the case of a stationary input we therefore wish to drive the coefficients to their fixed optimal values as rapidly as possible. Unfortunately, because these algorithms use a noisy version of the error gradient to drive the coefficients, the convergence rate is somewhat slower than the "block data" method referred to in Section I and in addition each coefficient has some nonzero variance even after convergence which increases the resultant output MSE.

In Appendix A we calculate the optimal mean value of $k_n(i)$ given that it has some nonzero variance. As would be expected the optimal value of $E[k_n(i)]$ is very close (but not generally equal) to the optimal fixed coefficient value given by (2.2) assuming $e_f(i|n-1)$ and $e_b(i|n-1)$ are stationary.

From (2.4) if we assume β_1 is small enough so that convergence takes place,

$$E_\infty\left[k_n(i)e_f^2(i|n-1)\right]=E_\infty\left[e_f(i|n-1)e_b(i-1|n-1)\right]$$

where $E_\infty[X_i]$ denotes the asymptotic mean value of the sequence X_i. If $k_n(i)$ and $e_f^2(i|n-1)$ are approximately uncorrelated we have $E_\infty k_n(i)\approx k_{n,\mathrm{opt}}(\infty)$. Similarly, from (2.6)

$$E_\infty\left[\frac{k_n(i)e_f^2(i|n-1)}{D_n(i+1)}\right]=E_\infty\left[\frac{e_f(i|n-1)e_b(i-1|n-1)}{D_n(i+1)}\right].$$

In general if X and Y are two random variables,

$$E\left[\frac{X}{Y}\right]\neq\frac{E[X]}{E[Y]}$$

and hence the estimate of $k_{n,\mathrm{opt}}$ obtained using the normalized algorithm is biased; however, simulations indicate the bias to be generally very small.

III. Single-Stage Adaptation

In this section we investigate the convergence behavior of a single stage of a multistage lattice assuming that both inputs to that stage are stationary. This is equivalent to assuming that the previous stages have fixed coefficients, as would approximately be true if they have already adapted. Our motivation is to determine the important factors which affect the convergence of a single stage, deferring consideration of the effects of the previous adapting stages to Section IV.

As implied in Section II, there are two distinguishing features to the adaptation algorithms being discussed. These are: 1) the time required for the filter to adapt from some initial state to its final (stationary) state, and 2) the final mean-squared value of the output error signal after convergence is achieved.

A. Single-Stage Convergence Time

To characterize the convergence time for a single stage we first iterate (2.4) and take the expected value of both sides to get

$$E\left[k_n(i+1)\right]=k_n(0)E\left\{\prod_{j=0}^{i}\left[1-\beta_1 e_f^2(j|n-1)\right]\right\}+\beta_1 E\left\{\sum_{j=0}^{i-1}e_f(j|n-1)e_b(j-1|n-1)\cdot\prod_{l=j+1}^{i}\left[1-\beta_1 e_f^2(l|n-1)\right]\right\}+\beta_1 E\left[e_f(i|n-1)e_b(i-1|n-1)\right]. \quad (3.1)$$

In general, evaluation of the right-hand terms is nontrivial due to correlations present in both sequences $e_f(i|n-1)$ and $e_b(i|n-1)$.[1] To simplify the discussion we therefore, assume that both sequences are each independent. Note that this assumption improves as n increases since each stage of the lattice attempts to whiten its two input signals. In addition, simulations have shown this assumption does not introduce major inaccuracies in our final results. We can then rewrite (3.1) as

$$E[k_n(i+1)]-k_{n,\mathrm{opt}} \approx \{1-\beta_1 E[e_f^2(i|n-1)]\}^{i+1}(k_n(0)-k_{n,\mathrm{opt}}).$$

$E[k_n(i)]$ therefore, decays exponentially towards $k_{n,\mathrm{opt}}$ with time constant

$$\tau_n \approx -\frac{1}{\ln\{1-\beta_1 E[e_f^2(i|n-1)]\}} \approx \frac{1}{\beta_1 E[e_f^2(i|n-1)]}. \tag{3.2}$$

In particular, the first-stage time constant $\tau_1 \approx 1/\beta_1 R_0$ where R_m, $m=0,1,2,\cdots$, is the autocorrelation sequence of the input signal. A disadvantage of this unnormalized algorithm which is approximately eliminated by the normalized algorithm is the dependence of adaptation speed upon the input signal variance.

When using the normalized algorithm given by (2.5), the initial value of the denominator is commonly set at β_2^{-1} times an initial estimate of the input power [3] (note from (2.5b) that $E_\infty[D_n(i)]=(1/\beta_2)E[e_f^2(i|n-1)]$). If this is done successfully we have $\beta_2 E_\infty[D_n(i)] \approx E[e_f^2(i|n-1)]$. Using this initial value and (2.6) we rewrite (3.2) for the normalized algorithm as

$$\tau_n = -\frac{1}{\ln\left\{1-E\left[\frac{e_f^2(i|n-1)}{D_n(i)}\right]\right\}} \approx -\frac{1}{\ln\left\{1-\frac{E[e_f^2(i|n-1)]}{E[D_n(i)]}\right\}} \approx -\frac{1}{\ln(1-\beta_2)} \tag{3.3}$$

implying that in this case the single-stage time constant is dependent only upon β_2.

If we make no assumptions about the residual energies, we can use (2.5) to write

$$k_n(i)= \frac{(1-\beta_2)^i C_n(0)+\sum_{j=1}^{i}(1-\beta_2)^{i-j}e_f(j-1|n-1)e_b(j-2|n-1)}{(1-\beta_2)^i D_n(0)+\sum_{j=1}^{i}(1-\beta_2)^{i-j}e_f^2(j-1|n-1)}. \tag{3.4}$$

Multiplying through by the denominator, taking expected values of both sides, and assuming $k_n(i)$ is approximately uncorrelated with $e_f^2(j|n-1)$, $0 \leq j \leq i-1$, yields

$$E[k_n(i)] \approx \frac{\beta_2(1-\beta_2)^i C_n(0)+[1-(1-\beta_2)^i]\cdot E[e_f(i|n-1)e_b(i-1|n-1)]}{\beta_2(1-\beta_2)^i D_n(0)+[1-(1-\beta_2)^i]E[e_f^2(i|n-1)]}.$$

(Note that as β_2 decreases, $k_n(i)$ should fluctuate less, and hence should be less correlated with $e_f^2(j|n-1)$.) The trajectory of $E[k_n(i)]$ is not exponential; however, defining τ_n as the time it takes $E[k_n(i)]$ to reach the value $k_{n,\mathrm{opt}}+(k_n(0)-k_{n,\mathrm{opt}})\gamma$ where $0<\gamma<1$ we get

$$\tau_n \approx \frac{1}{\beta_2}\ln\left[\beta_2\left(\frac{1-\gamma}{\gamma}\right)\frac{D_n(0)}{E[e_f^2(i|n-1)]}+1\right]. \tag{3.5}$$

In this case the "time constant" τ_n depends upon the normalized input signal variance

$$\frac{E[e_f^2(i|n-1)]}{D_n(0)}.$$

B. Single-Stage Output MSE

To compute the asymptotic mean-squared output signal for the nth stage of the adaptive lattice after convergence is achieved we square (2.1a), let

$$k_n(i)=E[k_n(i)]+\tilde{k}_n(i) \tag{3.6}$$

where $\tilde{k}_n(i)$ represents the instantaneous fluctuations of $k_n(i)$ about its mean value, and take asymptotic expected values of both sides to get

$$\begin{aligned} E_\infty[e_f^2(i|n)] &= E_\infty[e_f^2(i|n-1)] \\ &+\{E_\infty[k_n(i)]\}^2 E_\infty[e_b^2(i-1|n-1)] \\ &+E_\infty[\tilde{k}_n^2(i)e_b^2(i-1|n-1)] \\ &-2E_\infty[\tilde{k}_n(i)e_f(i|n-1)e_b(i-1|n-1)] \\ &+2E_\infty[k_n(i)]E_\infty[\tilde{k}_n(i)e_b^2(i-1|n-1)] \\ &-2E_\infty[k_n(i)]E_\infty[e_f(i|n-1)e_b(i-1|n-1)]. \end{aligned} \tag{3.7}$$

Unfortunately, the "noise" term $\tilde{k}_n(i)$ is generally correlated with both $e_f(i|n-1)$ and $e_b(i-1|n-1)$. As a first-order approximation, however, we shall ignore this effect, assume that $E_\infty[k_n(i)] \approx k_{n,\mathrm{opt}}$, and use (2.3) to get

$$E_\infty[e_f^2(i|n)] \approx \{1-k_{n,\mathrm{opt}}^2+E_\infty[\tilde{k}_n^2(i)]\}E[e_f^2(i|n-1)] \tag{3.8}$$

where $E_\infty[\tilde{k}_n^2(i)] \equiv \mathrm{var}_\infty k_n \approx E[k_n^2(i)]-k_{n,\mathrm{opt}}^2$. The asymptotic variance of $k_n(i)$, therefore, contributes an asymptotic excess mean-squared error at the output of the nth stage approximated by $[\mathrm{var}_\infty k_n]E_\infty[e_f^2(i|n-1)]$.

Completion of the description, therefore, requires an expression for $\mathrm{var}_\infty k_n$ for each of the two algorithms. Squaring (2.4), taking expected values of both sides assuming $k_n(i)$ is independent of $e_f(i|n-1)$ and $e_b(i-1|n-1)$,

[1] A similar problem is discussed by Mazo in the context of adaptive equalization [18].

and rearranging gives

$$\operatorname{var}_\infty k_n \approx \beta_1 \frac{E_\infty\left[k_{n,\text{opt}} e_f^2(i|n-1) - e_f(i|n-1)e_b(i-1|n-1)\right]^2}{2E_\infty\left[e_f^2(i|n-1)\right] - \beta_1 E_\infty\left[e_f^2(i|n-1)\right]}. \tag{3.9}$$

As a first approximation, if we further assume $e_f(i|n-1)$ and $e_b(i|n-1)$ to be jointly Gaussian, (3.9) simplifies to

$$\operatorname{var}_\infty k_n \approx \beta_1 \frac{\left(1-k_{n,\text{opt}}^2\right)E_\infty\left[e_f^2(i|n-1)\right]}{2-3\beta_1 E_\infty\left[e_f^2(i|n-1)\right]}. \tag{3.10}$$

It is interesting to note that as $|k_{n,\text{opt}}| \to 1$, (3.10) predicts that $\operatorname{var}_\infty k_n \to 0$. This could prove advantageous when doing spectral estimation where in general the closer $|k_n|$ is to one, the more accuracy is needed to represent k_n in order to stay within a given maximum spectral deviation [16].

Similarly, using (2.6) and assuming $k_n(i)$ is independent of $e_b(i-1|n-1)$ and $e_f(j|n-1)$, $0 \leq j \leq i$,

$$\operatorname{var}_\infty k_n \approx \frac{E_\infty\left[k_{n,\text{opt}} \dfrac{e_f^2(i|n-1)}{D_n(i+1)} - \dfrac{e_f(i|n-1)e_b(i-1|n-1)}{D_n(i+1)}\right]^2}{2E_\infty\left[\dfrac{e_f^2(i|n-1)}{D_n(i+1)}\right] - E_\infty\left[\dfrac{e_f^4(i|n-1)}{D_n^2(i+1)}\right]}. \tag{3.11}$$

If β_2 is small enough so that

$$D_n(i) \approx \frac{1}{\beta_2} E_\infty\left[e_f^2(i|n-1)\right]$$

for large i we can rewrite (3.11) as

$$\operatorname{var}_\infty k_n \approx \frac{E_\infty\left[k_{n,\text{opt}} e_f^2(i|n-1) - e_f(i|n-1)e_b(i-1|n-1)\right]^2}{2\left\{\dfrac{E_\infty\left[e_f^2(i|n-1)\right]}{\beta_2}\right\}E_\infty\left[e_f^2(i|n-1)\right] - E_\infty\left[e_f^4(i|n-1)\right]}. \tag{3.12}$$

Note that coefficient variance produced by the unnormalized algorithm as given by (3.9) is identical to coefficient variance produced by the normalized algorithm as given by (3.12) provided that

$$\beta_1 = \frac{\beta_2}{E_\infty\left[e_f^2(i|n-1)\right]}.$$

Simulations have generally verified this result.

An alternative method for calculating coefficient variance using (2.5) is to assume $k_n(i)$ is approximately uncorrelated with $e_f^2(j|n-1)$, $0 \leq j \leq i-1$, so that

$$\operatorname{var}_\infty k_n \approx \frac{E_\infty\left[C_n^2(i)\right]}{E_\infty\left[D_n^2(i)\right]} - k_{n,\text{opt}}^2.$$

The right-hand side can be evaluated assuming $e_f(i|n-1)$ and $e_b(i-1|n-1)$ are jointly Gaussian to give

$$\operatorname{var}_\infty k_n \approx \frac{\beta_2\left\{1-k_{n,\text{opt}}^2 + 2\sum_{m=1}^{\infty}(1-\beta_2)^m\left[\rho_m^2\left(1-2k_{n,\text{opt}}^2\right) + \rho_{m+1}\rho_{m-1}\right]\right\}}{2+\beta_2\left[1+4\sum_{m=1}^{\infty}(1-\beta_2)^m\rho_m^2\right]} \tag{3.13}$$

where

$$\rho_m = \frac{E_\infty\left[e_f(i|n-1)e_f(i+m|n-1)\right]}{E_\infty\left[e_f^2(i|n-1)\right]}.$$

If $e_f(i|n-1)$ is an uncorrelated sequence this simplifies to

$$\operatorname{var}_\infty k_n \approx \frac{\beta_2}{2+\beta_2}\left(1-k_{n,\text{opt}}^2\right).$$

Simulations have shown that (3.8), (3.9), and (3.13) are accurate when an uncorrelated Gaussian noise sequence is used as the input to a multistage lattice. Unfortunately, the accuracy of these formulas becomes questionable when applied to arbitrary correlated inputs using an arbitrary step size. In general the correlations tend to make the coefficient variance and hence, output MSE somewhat smaller than the previous formulas predict.

Thus far only the asymptotic behavior of the MSE output of a filter stage has been investigated. In Appendix B we investigate how this MSE varies with time. In particular we use the same approach used in [19] to compute single-stage output MSE as a function of time and to find a step size sequence $\beta(i)$ which minimizes this output MSE at each sampling interval.

In this section we have explored the convergence behavior of a single stage of an adaptive lattice filter assuming that the inputs are stationary. The results presented can be used to gain insight into the relationships between convergence time, output MSE, the step size β, and the input signal variance. The time constants also give us some idea of the speed of convergence of the filter as a whole, if we make the worst-case assumption that the first $(n-1)$ stages have to adapt before the nth stage can begin its adaptation. Intuitively, however, it is clear that this assumption is very pessimistic and hence, out interest in a simple model for the adaptation of a multistage filter in the next section.

IV. Multistage Adaptation Model

Intuitively we expect that the nth stage of a lattice filter will start adapting in the direction of its asymptotic optimum value before the first $(n-1)$ stages have completed their adaptation. In fact, in Appendix C it is shown that

$$\left.\frac{\partial}{\partial k_j} k_{n,\text{opt}}\right|_* = 0, \qquad 1 \leq j \leq n-1 \tag{4.1}$$

where "*" refers to the condition

$$k_m = k_{m,\text{opt}}, \qquad 1 \leq m \leq n-1 \tag{4.2}$$

which indicates that $k_{n,\text{opt}}$ is to first-order insensitive to k_1

through k_{n-1} when the latter are in the region of their optimum values.

The previous time constant calculations are, therefore, unsatisfactory for predicting the behavior of the multistage adaptation, since the affect of previous stages, adaptation must be taken into account. Our interest is in the trajectories of the mean values of the N PARCOR coefficients versus time, which could be obtained by averaging the results of multiple simulations of the algorithms. However, since multiple simulations are expensive and do not provide much insight, we develop in this section a simpler model for the multistage adaptation, and then demonstrate its accuracy through comparison to simulation results.

Our model for the adaptation of the nth stage is to simply ignore the statistical fluctuation of $k_1(i)$ through $k_{n-1}(i)$ about their mean values, since those fluctuations should have little effect on the *mean value* of $k_n(i)$. We can then assume that $k_1(i)$ through $k_{n-1}(i)$ are following their deterministic mean value trajectories. These plus the input statistics provide a set of second-order statistics for $e_f(i|n-1)$ and $e_b(i|n-1)$ versus time, which can be used to predict the mean value trajectory of $k_n(i)$. Proceeding one stage at a time, we can thereby predict the mean value trajectories of all the PARCOR coefficients. The resulting model, which is unfortunately represented by a computer program rather than analytically, is nevertheless much simpler and less expensive than a simulation. Further, by plotting quantities such as $k_{n,\mathrm{opt}}$ versus time, much insight can be gained. Simulation results indicate that the model is fairly accurate, even for relatively large n (i.e., $n=10$).

In order to describe our model in more detail, we first note that $e_f(i|n)$ and $e_b(i|n)$ are linear combinations of $y_i, y_{i-1}, \cdots, y_{i-n}$, i.e.,

$$e_f(i|n)=y_i-\sum_{j=1}^{n} f_{j|n}(i)y_{i-j}=\boldsymbol{F}^T(i|n)\boldsymbol{Y}_i \tag{4.3}$$

and

$$e_b(i|n)=y_{i-n}-\sum_{j=0}^{n-1} b_{j|n}(i)y_{i-j}=\boldsymbol{B}^T(i|n)\boldsymbol{Y}_i \tag{4.4}$$

where

$$\boldsymbol{F}^T(i|n)=[1 \quad -f_{1|n}(i) \quad \cdots \quad -f_{n|n}(i) \quad 0 \quad \cdots \quad 0]$$
$$\boldsymbol{B}^T(i|n)=[-b_{1|n}(i) \quad \cdots \quad -b_{n|n}(i) \quad 1 \quad 0 \quad \cdots \quad 0]$$
$$\boldsymbol{Y}_i^T=[y_i \quad y_{i-1} \quad \cdots \quad y_{i-n} \quad \cdots \quad y_{i-N-1}]$$

and all vectors have dimension $N+2$. We can, therefore, advantageously represent $e_f(i|n)$ and $e_b(i|n)$ by the coefficient vectors $\boldsymbol{F}(i|n)$ and $\boldsymbol{B}(i|n)$. The lattice recursions (2.1), with $k_n(i)$ replaced by its mean value trajectory, can then be reformulated as

$$\boldsymbol{F}(i|n)=\boldsymbol{F}(i|n-1)-E[k_n(i)][z^{-1}\boldsymbol{B}(i-1|n-1)] \tag{4.5}$$

and

$$\boldsymbol{B}(i|n)=z^{-1}\boldsymbol{B}(i-1|n-1)-E[k_n(i)]\boldsymbol{F}(i|n-1) \tag{4.6}$$

where $z^{-1}\boldsymbol{B}(i-1|n-1)$ represents $\boldsymbol{B}(i-1|n-1)$ shifted "down" one element, i.e.,

$$[z^{-1}\boldsymbol{B}(i-1|n)]_j=[\boldsymbol{B}(i-1|n)]_{j-1},$$
$$\text{for } 2\leq j\leq N+2 \text{ and } [z^{-1}\boldsymbol{B}(i-1|n)]_1=0.$$

$z^{-1}\boldsymbol{B}(i-1|n-1)$ must be used instead of $\boldsymbol{B}(i-1|n-1)$ since $e_b(i-1|n-1)$ is a linear combination of $y_{i-1}, y_{i-2}, \cdots, y_{i-n}$. Once we have $\boldsymbol{F}$ and $\boldsymbol{B}$ trajectories the second-order statistics of e_f and e_b can be estimated as

$$E[e_f^2(i|n)]=E\left[y_i-\sum_{j=1}^{n} f_{j|n}(i)y_{i-j}\right]^2$$
$$\approx\sum_{j=1}^{n+1}\sum_{m=1}^{n+1}[\boldsymbol{F}(i|n)]_j[\boldsymbol{F}(i|n)]_m R_{j-m} \tag{4.7}$$

and

$$E[e_f(i|n)e_b(i-1|n)]$$
$$=E\left\{y_i-\sum_{j=1}^{n} f_{j|n}(i)y_{i-j}\right\}\left\{y_{i-n-1}-\sum_{j=1}^{n} b_{j|n}(i-1)y_{i-j}\right\}$$
$$\approx\sum_{j=1}^{n+1}\sum_{m=2}^{n+2}[\boldsymbol{F}(i|n)]_j[\boldsymbol{B}(i-1|n-1)]_{m-1}R_{j-m} \tag{4.8}$$

where $R_j=E[y_iy_{i+j}]$. Finally, the trajectories of the PARCOR coefficient means follow from (2.4) or (2.5) where each random element is replaced by its mean, i.e., in the case of the normalized algorithm we approximate

$$E[k_n(i)]\approx\frac{E[C_n(i)]}{E[D_n(i)]}.$$

It should be noted that since reducing the step size β will reduce the statistical fluctuations present in each coefficient, the accuracy of the model should improve as β decreases.

To illustrate the accuracy of the model we show graphs of $E[k_n(i)]$, $n=1$, 2, 4, and 10, for a particular case of input statistics in Fig. 2 (unnormalized algorithm) and in Fig. 3 (normalized algorithm). Shown in addition to $E[k_n(i)]$ as generated by the model is $k_{n,\mathrm{opt}}(i)$, the value towards which $E[k_n(i)]$ is converging at time i. The change in direction of $E[k_n(i)]$ whenever it crosses $k_{n,\mathrm{opt}}$ is evident. Also shown is $E[k_n(i)]$, $n=1$, 2, 4, and 10, as obtained by averaging the results of 200 simulations of each algorithm (the input was obtained by passing white Gaussian noise through a ten pole filter). Figs. 4 and 5 show output MSE versus time as obtained from the model and from averaging 200 simulations.

Much of the difference between the model and simulated curves can be attributed to the different asymptotic values towards which they converge. The value of $E[k_n(i)]$ as generated by the model converges to the optimum value of k_n for the given input statistics under the assumption that $k_1, \cdots, k_{n-1}$ are fixed at their optimal values. In reality, however, k_1 through k_{n-1} have some nonzero variance not

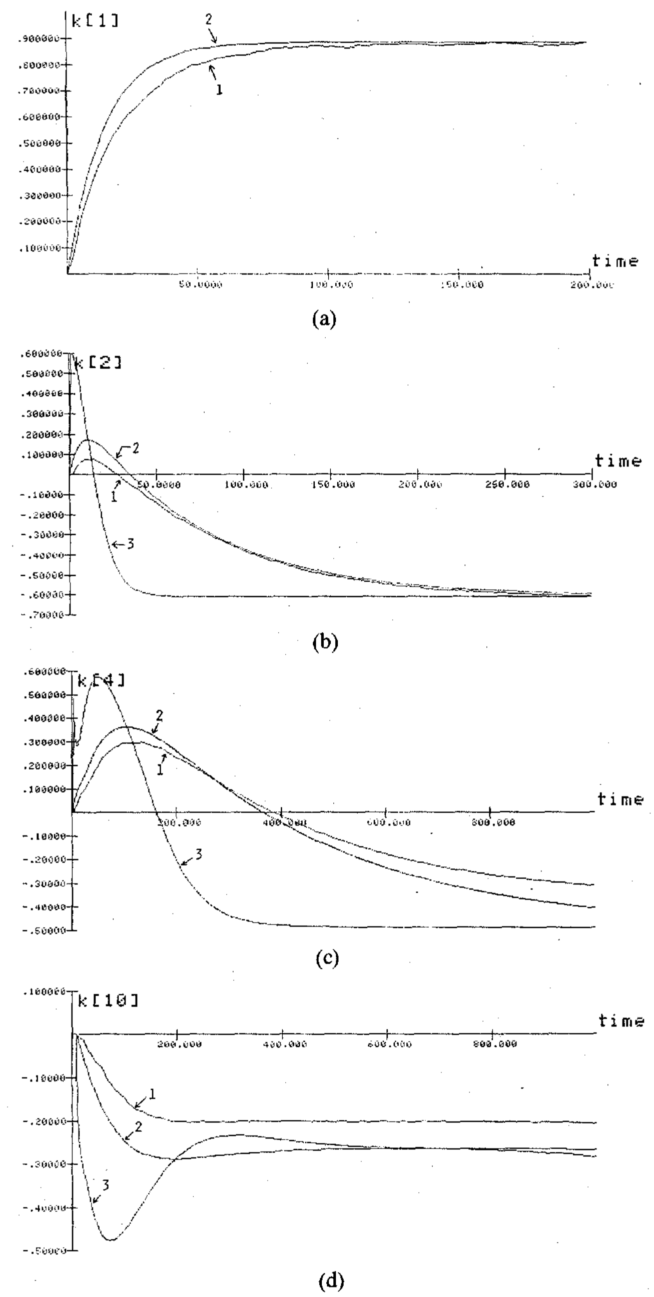


Fig. 2. Mean value trajectories of PARCOR coefficients 1, 2, 4, and 10 in a tenth-order filter using the unnormalized gradient algorithm by (Curve 1) simulation and (Curve 2) from the model in Section IV. Curve 3 shows the trajectory of $k_{n,\mathrm{opt}}(i)$ given by the model.

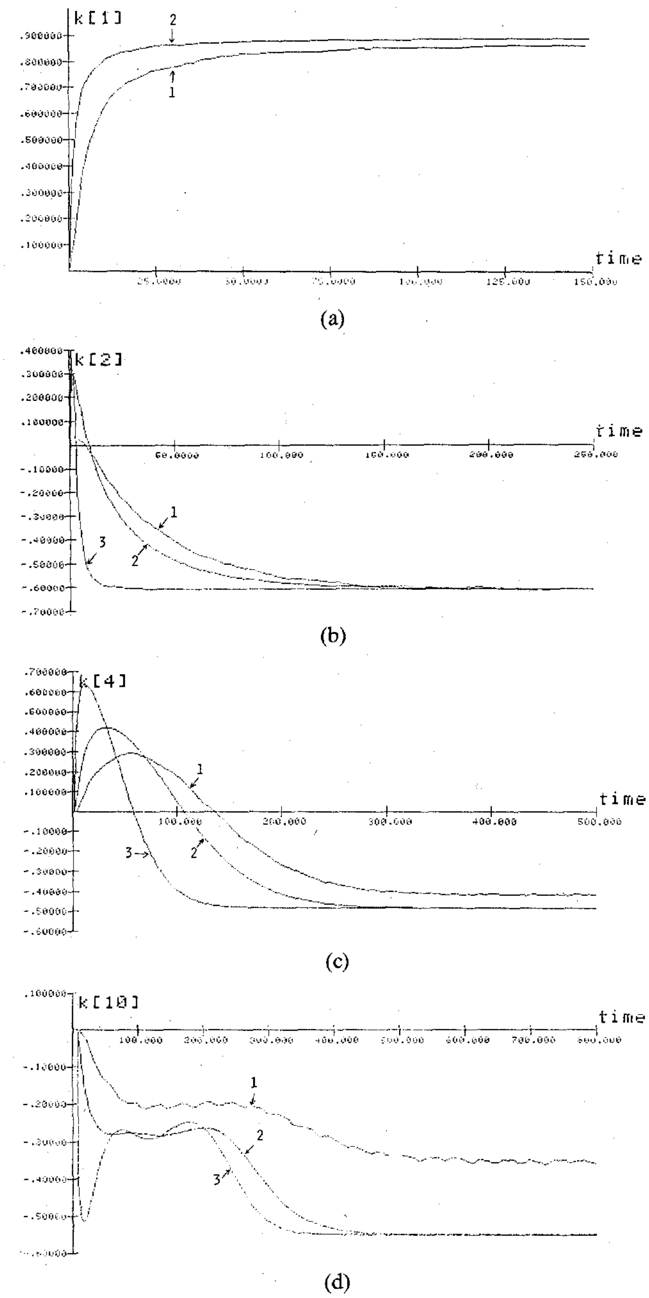


Fig. 3. Mean value trajectories of the PARCOR coefficients, 1, 2, 4, and 10 in a tenth-order filter using the normalized gradient algorithm by (Curve 1) simulation and (Curve 2) from the model in Section IV. Curve 3 shows the trajectory of $k_{n,\mathrm{opt}}(i)$ given by the model.

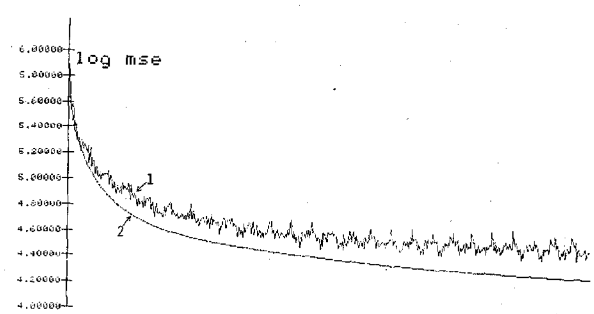


Fig. 4. Output MSE of a tenth-order lattice unnormalized gradient algorithm by (Curve 1) simulation and (Curve 2) from the model in Section IV.

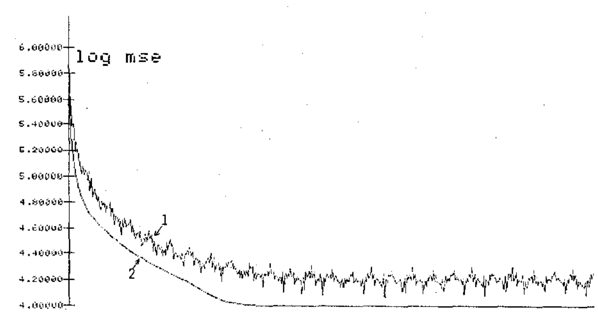


Fig. 5. Output MSE of a tenth-order lattice normalized gradient algorithm by (Curve 1) simulation and (Curve 2) from the model in Section IV.

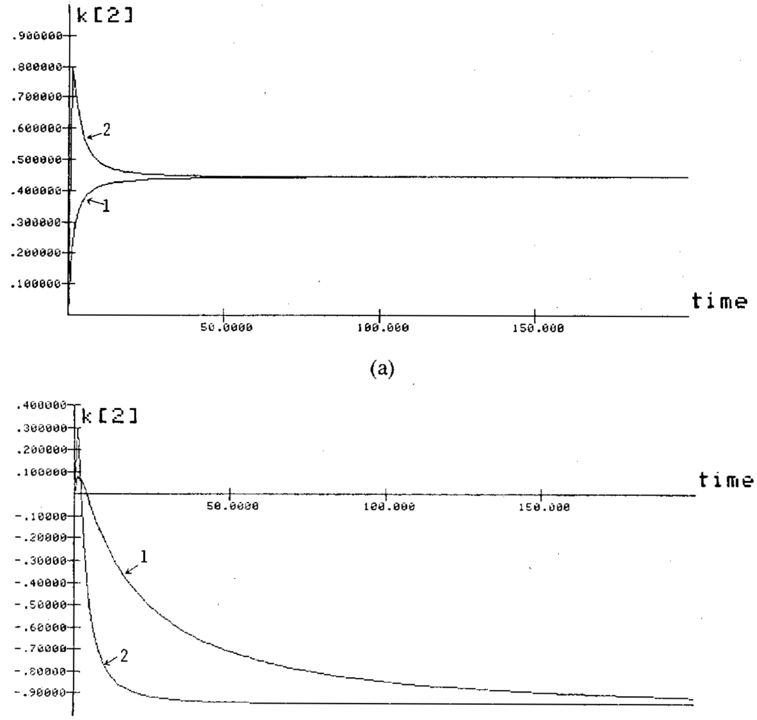


Fig. 6. Mean value trajectories (Curve 1) of the second PARCOR coefficient in a two-stage adaptive lattice (given by the model in Section IV) using different input statistics. Curve 2 shows the trajectory of $k_{2,\mathrm{opt}}(i)$. Fig. 6(b) represents a "fast mode" of the filter, Fig. 6(b) represents a "slow mode."

accounted for in the model. The offset shown in the simulated curve is therefore primarily due to the fact that the variances of k_1 through k_{n-1} perturb the statistics of $e_f(i|n-1)$ and $e_b(i|n-1)$, producing an offset between the simulated and calculated versions of $k_{n,\mathrm{opt}}$. (Note also that the asymptotic MSE predicted by the model is the minimum attainable by a fixed coefficient filter.)

In the next section we use the results obtained thus far to discuss the dependence of the lattice filter's convergence speed upon the input statistics and to compare this behavior with that exhibited by the adaptive transversal filter.

V. Two-Stage Lattice-Transversal Comparison

Two common claims concerning the adaptive lattice filter are: 1) the convergence speed of the adaptive lattice is approximately independent of the input statistics (eigenvalue spread of the input autocorrelation matrix), and 2) the adaptive lattice filter will generally converge faster to a given (stationary) input than the adaptive transversal filter assuming that both filters have already converged to some different set of input statistics. In this section we show that these claims are not strictly true by presenting two-stage counterexamples. A detailed discussion of the general n-stage case appears to be considerably more complicated, although the basic ideas used to derive our two-stage counterexamples should carry through.

We begin by considering the first claim when the normalized algorithm is used along with the assumptions used to derive (3.3). In particular if we assume each stage (coefficient) does not start to converge until its inputs are stationary (i.e., when $k_{n,\mathrm{opt}}(i)$ reaches its asymptotic value), the adaptive lattice will converge stage by stage. From (3.3) we know that the time constants for each stage are at least approximately dependent only upon the step size β and hence are independent of the input signal statistics. We have already seen, however, that the convergence speed of the nth stage is significantly influenced by the behavior of the first $(n-1)$ stages. In particular, $E[k_n(i)]$ is continually moving towards $k_{n,\mathrm{opt}}(i)$ which *does* depend upon the input signal statistics. The trajectory of $k_{n,\mathrm{opt}}(i)$ before it reaches its asymptotic value will, therefore, significantly influence the trajectory of $E[k_n(i)]$ and hence the convergence time of the filter.

In general the trajectories of $k_{n,\mathrm{opt}}$, $n>2$, are quite complicated (as Figs. 2 and 3 will testify) so that it is quite difficult to analytically determine the dependence of $k_{n,\mathrm{opt}}(i)$, $n>2$, upon the input statistics. For $n=2$; however, the problem simplifies considerably. To illustrate the previous discussion we, therefore, consider a two-stage lattice for which $k_1(0)=k_2(0)=0$ and examine the dependence of the trajectories of $E[k_1(i)]$ and $E[k_2(i)]$ upon the input statistics (i.e. R_0, R_1, and R_2). Since we assume a stationary input $E[k_1(i)]$ will follow an approximately exponential path with time constant given by (3.3) independent of the input statistics. We, therefore, concentrate on the behavior of $E[k_2(i)]$. Using our standard assump-

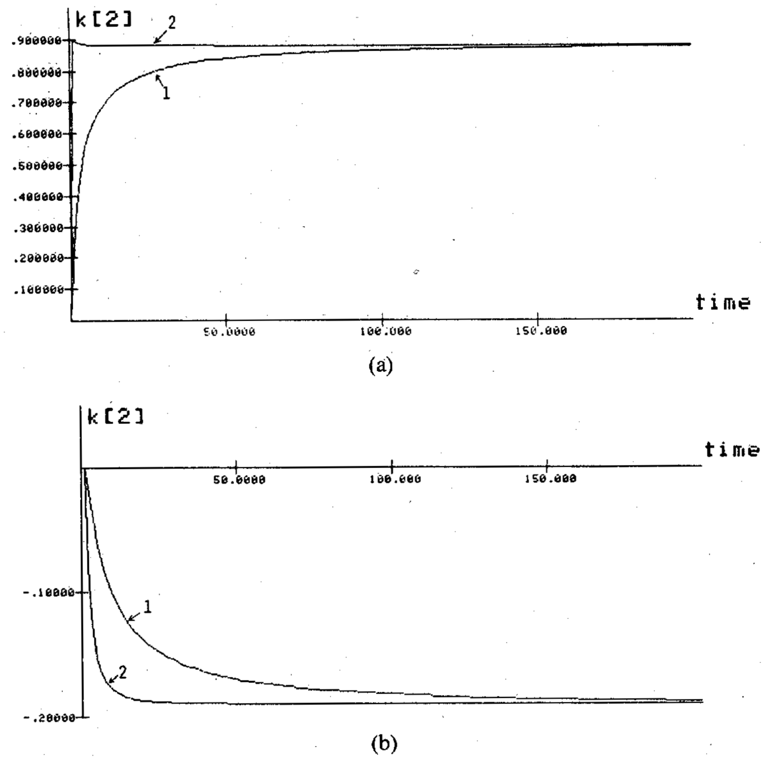


Fig. 7. Trajectories of (Curve 1) $E[k_2(i)]$ and (Curve 2) $k_{2,\mathrm{opt}}(i)$ for a two-stage adaptive lattice using different input statistics. The value of R_1 used in this case is significantly less than that used to generate the curves in Fig. 6.

tions we have

$$k_{2,\mathrm{opt}}(i)=\frac{E\left[e_f(i|1)e_b(i-1|1)\right]}{E\left[e_f^2(i|1)\right]}$$

$$=\frac{E\left[(y_1-k_1(i)y_{i-1})(y_{i-2}-k_1(i)y_{i-1})\right]}{E\left[y_i-k_1(i)y_{i-1}\right]^2}$$

$$\approx\frac{R_2-2E\left[k_i(i)\right]R_1+E\left[k_1^2(i)\right]R_0}{R_0-2E\left[k_1(i)\right]R_1+E\left[k_1^2(i)\right]R_0}$$

which can be rewritten as

$$k_{2,\mathrm{opt}}(i)=1-\frac{R_2-R_0}{E\left[e_f^2(i|1)\right]}.$$

This implies that

$$k_{2,\mathrm{opt}}(i)-k_{2,\mathrm{opt}}(\infty)$$

$$=(R_0-R_2)\left[\frac{1}{E_\infty\left[e_f^2(i|1)\right]}-\frac{1}{E\left[e_f^2(i|1)\right]}\right]\geqslant 0$$

since

$$E_\infty\left[e_f^2(i|1)\right]\leqslant E\left[e_f^2(i|1)\right]$$

and hence

$$k_{2,\mathrm{opt}}(i)\geqslant k_{2,\mathrm{opt}}(\infty)=E_\infty\left[k_2(i)\right] \tag{5.1}$$

for all i, independent of the input signal statistics. Now $k_{2,\mathrm{opt}}(i)$ is monotonically decreasing; hence, by changing the value of $k_{2,\mathrm{opt}}(0)=R_2/R_0$, as shown in Fig. 6, we can significantly alter the convergence time of $E[k_2(i)]$, and hence the convergence time for the filter. Fig. 7 shows the same type of behavior for a smaller value of R_1/R_0. Figs. 6(a) and 7(a) represent a fast "mode" of the filter which corresponds to $k_2(0)<k_{2,\mathrm{opt}}$ while Figs. 6(b) and 7(b) represent a slower "mode" which corresponds to $k_2(0)>k_{2,\mathrm{opt}}$.

It is instructive to compare this behavior with that exhibited by the two-stage adaptive transversal filter. Specifically, the transversal algorithm considered is the familiar LMS algorithm

$$f_{j|n}(i+1)=f_{j|n}(i)+\beta y_{i-j}e_f(i|n) \tag{5.2}$$

where $f_{j|n}(i)$ is the jth tap coefficient for an nth-order filter at time i. (Note that this is an unnormalized algorithm analogous to (2.4). A normalized version of (5.2) could also be considered.) As discussed in [20], for $n=2$ the mean value of the tap vector converges towards its optimal value according to fast and slow normal modes. The time constant associated with each normal mode is $\tau_i\approx 1/\beta\lambda_i$, $i=1,2$, where λ_i is the ith eigenvalue of the 2×2 autocorrelation matrix. In particular we have $\lambda_1=R_0+R_1$ and $\lambda_2=R_0-R_1$. If $f_1(0)$ and $f_2(0)$ are fixed, as discussed in Appendix D we can excite each mode to different degrees by changing the value of R_2/R_0. This is similar to the two-stage lattice behavior just discussed. Furthermore, for both the two-stage lattice and two-stage transversal filters as $|R_1|$ (and hence the eigenvalue spread) increases the difference between convergence times associated with each "mode" becomes greater.

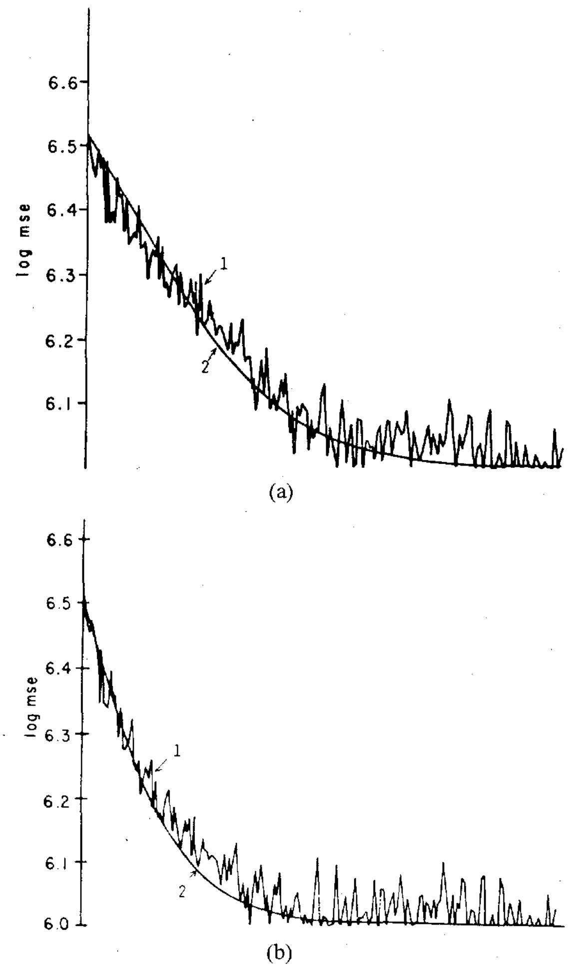


Fig. 8. Output MSE curves by (Curve 1) simulation and (Curve 2) from the model in Section IV for (a) a two-stage adaptive lattice, and (b) a two-stage adaptive transversal filter using equivalent initial conditions and input statistics.

We, therefore, conclude that lattice convergence speed is not independent of the input signal statistics. In fact the two-stage lattice exhibits a similar type of dependence upon the input signal statistics as the two-stage transversal filter. On the other hand the difference between slow and fast "modes" as shown in Figs. 6 and 7 is less than the difference between normal modes in a two-stage transversal filter using the same value of R_1.

We now examine the second claim stated at the beginning of this section by considering a two-stage adaptive transversal filter and a two-stage adaptive lattice which have converged to an initial set of input statistics, and at time $i=0$ apply a stationary input with different statistics. In Appendix D we use the previous discussion to derive necessary conditions for which the two-stage transversal filter will converge faster than the two-stage lattice. As an example, Fig. 8 shows output MSE by both simulation and as generated by the model in Section IV for two-stage adaptive lattice and transversal filters, respectively, using initial conditions and input signal statistics derived in Appendix D. (The asymptotic output MSE for both filters were approximately equal.) It is clear that in this case the transversal filter converges faster than the lattice.

We, therefore, conclude that the adaptive lattice filter does not always converge faster than its transversal counterpart. On the other hand in Appendix D we show that if the first autocorrelation coefficient of the input sequence is near unity, as is often the case in speech, the two-stage transversal filter can converge faster than the two-stage lattice only if the initial misadjustment is slight. (Note that this is a necessary but not sufficient conditions.) In speech processing applications the (two-stage) adaptive lattice would therefore appear to have a significant advantage over the (two-stage) adaptive transversal filter.

VI. Conclusions

This paper has begun the task of quantitatively characterizing the convergence properties of the adaptive lattice filter. While the adaptation process displays an unfortunate nonlinear interaction between a given stage and all those before it, by making approximations we have succeeded in obtaining results. The model presented in Section IV represents a first step towards predicting the behavior of the multistage adaptive lattice filter for an arbitrary stationary input, and achieves reasonable accuracy and at the same time is simple and inexpensive to compute.

Due to the different statistical behavior of different PARCOR coefficients when processing speech, it is likely that the convergence properties of the adaptive lattice (using gradient algorithms) can be improved by using different step sizes for the different stages. Another open issue is the convergence properties of appropriately modified versions of the algorithms given in this paper in comparison to those exhibited by the algorithms given in [10] and [11] which obtain an exact least squares solution at each time instant. These issues are currently being investigated.

Appendix A

Given that each coefficient has some nonzero variance after convergence is achieved we ask whether the optimal mean value of each coefficient is equal to its optimal fixed coefficient value. To answer this question we examine (3.7) which gives $E_\infty[e_f^2(i|n)$ as a function of $E_\infty[k_n(i)]$, $\tilde{k}_n(i)$, and the input signals to stage n. Minimizing $E_\infty[e_f^2(i|n)]$ with respect to $E_\infty[k_n(i)]$ gives

$$\{E_\infty[k_n(i)]\}_{\text{opt}} = \frac{E_\infty[e_f(i|n-1)e_b(i-1|n-1)]}{E_\infty[e_b^2(i|n-1)]} - \frac{E_\infty[\tilde{k}_n(i)e_b^2(i-1|n-1)]}{E_\infty[e_b^2(i|n-1)]}. \quad (A.1)$$

In general $\tilde{k}_n(i)$ is correlated with $e_b^2(i-1|n-1)$; however, simulations have shown the offset term to be generally negligible, and hence

$$\{E_\infty[k_n(i)]\}_{\text{opt}} \approx k_{n,\text{opt}}.$$

We note here that the same technique can also be used to investigate the nature of the coefficient bias referred to in Section 4. If we assume $k_{n+1}(i)$ converges to $k_{n+1,\text{ opt}}$ given by (2.2), this bias is caused by the fluctuations of

$k_1(i)$ through $k_n(i)$ about their mean values which cause $k_{n+1,\,opt}$ to differ from the value of $k_{n+1,\,opt}$ calculated by assuming $k_1(i)$ through $k_n(i)$ are fixed at their optimal values. The effect of $k_n(i)$ upon $E_\infty[k_{n+1}(i)]$ can, therefore, be estimated by using (3.6), (2.1), and (2.2), and assuming that $k_n(i)$ is statistically independent of the input signals to stage n. Unfortunately, while the resulting relation illustrates the nature of the bias, simulations have shown it to be an inaccurate estimate of coefficient bias. The effects of correlations must, therefore, be taken into account, making the problem considerably more difficult. We have empirically observed, however, that in general the simulated value of $|E_\infty[k_n]|$ is less than $|E_\infty[k_n]|$ as generated by the model (which is intuitively satisfying if we view the effect of previous coefficient variations as partially whitening the input to the current stage).

Appendix B

We investigate single-stage output MSE as a function of time considering, for simplicity, only the unnormalized algorithm. Squaring (2.1a), substituting (2.4) for $k_n(i)$, assuming $e_f(i|n-1)$ and $e_b(i|n-1)$ are jointly Gaussian and independent of $k_n(i)$, and using (2.3), we get (after some manipulation) the following recursive equation for $E[e_f^2(i|n)]$:

$$E\left[e_f^2(i|n)\right] \approx \left[\alpha(3\alpha-2)+1\right] E\left[e_f^2(i-1|n)\right] + 2\alpha(1-\alpha)\epsilon_{min} \tag{B.1}$$

where

$$\alpha = \beta_1 E\left[e_f^2(i|n-1)\right]$$

and

$$\epsilon_{min} = \left(1-k_{n,opt}^2\right) E\left[e_f^2(i|n-1)\right].$$

(Note that

$$E_\infty\left[e_f^2(i|n)\right] \approx \frac{2(1-\alpha)}{2-3\alpha}\epsilon_{min}$$

which agrees with (3.8) and (3.10).) For stability we require that

$$0 < \alpha(3\alpha-2)+1 < 1$$

or equivalently

$$0 < \alpha < \frac{2}{3}. \tag{B.2}$$

Note that this stability requirement for output MSE is stricter than the stability requirement for coefficient mean values ($0<\alpha<1$) obtained from (2.4). (An analogous result holds for the LMS transversal algorithm (5.2) [21].) The fastest convergence rate occurs when $3\alpha^2-2\alpha+1$ is minimized, i.e., when $\alpha=1/3$. In this case the asymptotic output MSE$\approx 4/3\epsilon_{min}$.

As in [19] we can find a time varying sequence of step sizes $\alpha(i)$ which minimizes the output MSE at each iteration. Following the same procedures outlined in [19] we get the following iterative formula for the optimal step size sequence:

$$\alpha_{opt}(i+1) = \alpha_{opt}(i)\frac{\alpha_{opt}(i)-1}{3\alpha_{opt}^2(i)-1}. \tag{B.3}$$

If

$$\frac{\alpha(i)-1}{3\alpha^2(i)-1} < 1$$

then

$$\alpha_{opt}(i) \underset{i\to\infty}{\to} 0$$

and

$$E_\infty\left[e_f^2(i|n)\right] = \epsilon_{min}.$$

This result is similar to the optimal step size sequence for the LMS transversal filter obtained in [19]. Interestingly, in contrast to the transversal filter, for the multistage lattice the optimal step sizes for different stages will be unequal. (Although the general n-stage solution has not been attempted, intuition would suggest that the optimal step size sequence for the nth stage might be obtained by keeping $\alpha_n(i)-\beta_{1n}(i)E[e_f^2(i|n-1)]=1/3$ until the input to that stage is approximately stationary (i.e., when

$$E\left[e_f^2(i|n-1)\right] \approx E_\infty\left[e_f^2(i|n-1)\right])$$

and then using (B.3).)

Appendix C

We wish to establish (4.1). We first note that

$$e_f(i|n-1) = y_i - \sum_{j=1}^{n-1} f_{j|n-1} y_{i-j} \tag{C.1}$$

and

$$e_b(i-1|n-1) = y_{i-n} - \sum_{j=1}^{n-1} b_{j|n-1} y_{i-j} \tag{C.2}$$

where the forward and backward prediction coefficient satisfy

$$\frac{\partial}{\partial f_{j|n-1}}\left\{E\left[e_f^2(i|n-1)\right]\right\} = \frac{\partial}{\partial b_{j|n-1}}\left\{E\left[e_b^2(i|n-1)\right]\right\} = 0, \quad i \leqslant j \leqslant n-1.$$

Now

$$k_{n,opt} = \frac{E\left[e_f(i|n-1)e_b(i-1|n-1)\right]}{E\left[e_f^2(i|n-1)\right]}$$

and hence

$$\frac{\partial k_{n,opt}}{\partial k_j} = \frac{E\left[e_f^2(i)\right]\frac{\partial}{\partial k_j}E\left[e_f(i)e_b(i-1)\right] - E\left[e_f(i)e_b(i-1)\right]\frac{\partial}{\partial k_j}E\left[e_f^2(i)\right]}{\left\{E\left[e_f^2(i)\right]\right\}^2} \tag{C.3}$$

where all prediction errors have order $(n-1)$. Given (4.2) it follows that $E[e_f^2(i|n-1)]$ is at its minimum value, and hence

$$\frac{\partial}{\partial k_j}\left\{E\left[e_f^2(i|n-1)\right]\right\}\Big|_{*}=0. \tag{C.4}$$

Also,

$$\frac{\partial}{\partial k_j}E\left[e_f(i|n-1)e_b(i-1|n-1)\right]$$

$$=E\left[e_f(i|n-1)\frac{\partial}{\partial k_j}e_b(i-1|n-1)\right]$$

$$+E\left[e_b(i-1|n-1)\frac{\partial}{\partial k_j}e_f(i|n-1)\right]$$

$$=-\sum_{m=1}^{n-1}\frac{\partial b_{m|n-1}}{\partial k_j}E\left[y_{i-m}e_f(i|n-1)\right]$$

$$-\sum_{m=1}^{n-1}\frac{\partial f_{m|n-1}}{\partial k_j}E\left[y_{i-m}e_b(i-1|n-1)\right] \tag{C.5}$$

from (C.1) and (C.2). From the principle of orthogonality it follows that this term is also zero, and hence (4.1) follows from (C.3)–(C.5).

Note that this result does not hold for prediction coefficient, i.e., for a two-stage transversal filter,

$$E\left[e_f^2(i|2)\right]=R_0\left(1+f_{1|2}^2+f_{2|2}^2\right)+2f_{1|2}\left(f_{2|2}-1\right)R_1-2f_{2|2}R_2$$

from which it follows that

$$f_{2|2,\text{opt}}=r_2-f_{1|2}r_1$$

$$\frac{\partial f_{2|2,\text{opt}}}{\partial f_{1|2}}=-r_1$$

where $r_i=R_i/R_0$, for all $f_{1|2}$.

Appendix D

Following the discussion presented in Section V we examine the conditions under which a two-stage transversal filter will converge faster than the analogous two-stage lattice. For simplicity we use (2.4) and (5.2) as our adaptive algorithms, although the following discussion can be reinterpreted for the respective normalized versions.

We first observe the following inequalities:

$$\frac{1}{\beta R_0(1+|r_1|)}\leqslant\frac{1}{\beta R_0}\leqslant\frac{1}{\beta R_0\left(1-r_1^2\right)}\leqslant\frac{1}{\beta R_0(1-|r_1|)}$$

where $r_i=R_i/R_0$ which can be rewritten as

$$\tau_f^{(t)}\leqslant\tau_1^{(l)}\leqslant\tau_2^{(l)}\leqslant\tau_s^{(t)} \tag{D.1}$$

where $\tau_f^{(t)}$ and $\tau_s^{(t)}$ represent, respectively, the time constants associated with the two-stage transversal fast and slow normal modes and $\tau_i^{(l)}, i=1,2$, is the ith-stage lattice time constant obtained from (3.2). If we, therefore, select the initial conditions and input statistics to (1) excite only the fast normal mode of the transversal filter and (2) cause the lattice filter to exhibit the type of behavior shown in Figs. 6(b) and 7(b), i.e., excite its slower "mode", the adaptive transversal filter should then converge faster than the adaptive lattice.

To derive an example satisfying both of these conditions consider a second-order all-pole input sequence,

$$y_i=\eta_i+ay_{i-1}+by_{i-2}$$

where η_i is a stationary zero-mean independent sequence. We have

$$r_1=\frac{a}{1-b} \tag{D.2a}$$

$$r_2=\frac{a^2}{1-b}+b. \tag{D.2b}$$

The mean values of the tap coefficients are given by

$$E[f_1(i)]\approx\frac{r_1(1-r_2)}{1-r_1^2}$$

$$+1/2\left[f_2(0)+f_1(0)-\frac{r_1+r_2}{1+r_1}\right]\left[1-\beta(R_0+R_1)\right]^i$$

$$-1/2\left[f_2(0)-f_1(0)+\frac{r_1-r_2}{1-r_1}\right]\left[1-\beta(R_0+R_1)\right]^i$$

$$E[f_2(i)]\approx\frac{r_2-r_1^2}{1-r_1^2}$$

$$+1/2\left[f_2(0)+f_1(0)-\frac{r_1+r_2}{1+r_2}\right]\left[1-\beta(R_0+R_1)\right]^i$$

$$-1/2\left[f_2(0)-f_1(0)+\frac{r_1-r_2}{1-r_1}\right]\left[1-\beta(R_0-R_1)\right]^i.$$

Without loss of generality we can assume $R_1>0$. To excite only the fast normal mode we, therefore, require

$$f_2(0)-f_1(0)+\frac{r_1-r_2}{1-r_1}=0$$

or

$$f_2(0)-f_1(0)=b-a. \tag{D.3}$$

To ensure that the actual "time constant" for the second stage of the lattice is greater than or equal to $\tau_2^{(l)}$, the discussion in Section V tells us

$$f_2(0)=k_2(0)>k_{2,\text{opt}}. \tag{D.4}$$

Now

$$|b|=|k_{2,\text{opt}}|<1 \tag{D.6}$$

for stability, hence from (D.2a) and (D.5)

$$a>0.$$

From (D.3)–(D.5),

$$f_1(0)-a=f_2(0)-b>0$$

so that $f_1(0)>a>0$. For stability we also require

$$|k_1(0)|=\left|\frac{f_1(0)}{1-f_2(0)}\right|\leqslant1$$

or

$$f_2(0)+f_1(0)\leqslant1. \tag{D.6}$$

Combining (D.3)–(D.6) gives

$$0<a<f_1(0)\leqslant\frac{1-(b-a)}{2} \qquad \text{(D.7a)}$$

$$-1<b<f_2(0)\leqslant\frac{1+(b-a)}{2}. \qquad \text{(D.7b)}$$

To make our example as dramatic as possible we wish to strengthen the inequalities given by (D.1). To do this we must make r_1 relative large. Finally, to make the example meaningful we would like the initial MSE to be significantly greater than the asymptotic MSE. This implies making

$$d\equiv f_1(0)-a=f_2(0)-b$$

relatively large. From (D.7) we, therefore, will select

$$f_1(0)=\frac{1-(b-a)}{2} \text{ and } f_2(0)=\frac{1+(b-a)}{2}$$

so that

$$d=\frac{1-(b-a)}{2}-a=1/2[1-(b+a)] \qquad \text{(D.8)}$$

and attempt to make $(b+a)$ as negative as possible. To illustrate the previous discussion suppose we select $r_1=0.9$. From (D.1)

$$\tau_f^{(t)}\approx\frac{0.526}{\beta R_0}<\frac{1}{\beta R_0}\approx\tau_1^{(l)}.$$

Also, from (D.2a)

$$a+b=0.9+0.1b$$

which is most negative when $b=-1$. From (D.8)

$$d=1/2[1-(0.9-0.1)]=0.1.$$

For this case we, therefore, see that the transversal filter coefficients will converge nearly twice as fast as the lattice coefficients, however, the (maximum) initial misadjustment is slight. In order to find an example with a larger initial misadjustment we must decrease the value of r_1. The example shown in Figs. 8 and 9 was obtained by setting $d=0.7$, $a=0.3$, and $b=-0.7$. This in turn gives $r_1=0.176$, and $r_2=0.647$. From (D.1)

$$\tau_f^{(t)}=\frac{0.85}{\beta R_0}<\frac{1}{\beta R_0}=\tau_1^{(l)}.$$

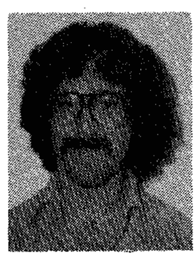
From (D.7) and (D.8), $f_1(0)=1$, $k_2(0)=f_2(0)=0$, and $k_1(0)=f_1(0)/(1-f_2(0))=1$. Also, $f_{1,\text{opt}}=a$, $k_{2,\text{opt}}=f_{2,\text{opt}}=b$, and $k_{1,\text{opt}}=r_1$. The same step size was used for both algorithms since the asymptotic MSE was approximately the same in both cases.

Acknowledgment

The authors wish to thank the reviewers for their helpful comments and suggestions.

References

[1] L. J. Griffiths, "A continuously-adaptive filter implemented as a lattice structure," in *Proc. IEEE ICASSP*, (Hartford, CT), pp. 683–686, May 1977.
[2] J. Makhoul, "A class of all-zero digital filters: Properties and applications," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-26, pp. 304–314, Aug. 1978.
[3] E. H. Satorius and S. T. Alexander, "Channel equalization using adaptive lattice algorithms," *IEEE Trans. Comm.*, vol. COM-27, pp. 899–905, June 1979.
[4] C. J. Gibson and S. Haykins, "A comparison of algorithms for the calculation of adaptive lattice filters," in *Proc. 1980 ICASSP*, (Denver, CO), pp. 978–983, Apr. 1980.
[5] S. Horvath, Jr., "Lattice form adaptive recursive digital filters: Algorithms and applications," in *Proc. 1980 Conf. Circuits Syst.*, (Houston, TX), pp. 128–133.
[6] D. Parikh, N. Ahmed, and S. D. Stearns, "An adaptive lattice algorithm for recursive filters," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-28, pp. 110–111, Feb. 1980.
[7] D. G. Messerschmitt, "A class of generalized lattice filters," *IEEE Trans. Acoust. Signal, Speech Processing*, vol. ASSP-28, pp. 198–204, Apr. 1980.
[8] J. Makhoul, "Stable and efficient lattice methods for linear prediction," *IEEE Trans. Acoust. Speech, Signal Processing*, vol. ASSP-25, pp. 423–428, Oct. 1977.
[9] M. Morf, D. T. Lee, J. R. Nickolls, and A. Vieira, "A classification of algorithms for ARMA models and ladder realizations," in *Proc. 1977 IEEE Conf. Acoust. Signal, Speech Processing*, (Hartford, CT), pp. 13–19, Apr. 1977.
[10] M. Morf and D. Lee, "Recursive least squares ladder forms for fast parameter tracking," in *Proc. 1978 IEEE Conf. D&C*, (San Diego, CA), pp. 1326–1367, Jan. 12, 1979.
[11] D. T. L. Lee and M. Morf, "Recursive square-root ladder estimation algorithms," in *Proc. 1980 ICASSP*, (Denver, CO), Apr. 1980.
[12] E. Satorius and J. Pack, "Application of least squares lattice algorithms to adaptive equalization," preprint, May 1979.
[13] A. H. Gray and J. D. Markel, *Linear Prediction of Speech*. Berlin, Springer-Verlag, 1976.
[14] L. R. Rabiner and R. W. Schafer, *Digital Processing of Speech Signals*. Englewood Cliffs, NJ: Prentice-Hall, 1978.
[15] P. L. Chu and D. G. Messerschmitt, "Zero-sensitivity analysis of the digital lattice filter," in *Proc. 1980 ICASSP*, (Denver, CO), Apr. 1980.
[16] A. H. Gray and J. D. Markel, "Quantization and bit allocation in speech processing," *IEEE Trans. Acoust. Signal, Speech Processing*, vol. ASSP-24, pp. 459–473, Dec. 1976.
[17] M. Honig and D. G. Messerschmitt, "Convergence models for adaptive gradient and least squares algorithms," in *Proc. 1981 ICASSP*, 1981.
[18] J. E. Mazo, "On the independence theory of equalizer convergence," *Bell Syst. Tech. J.*, vol. 58, pp. 963–993, May–June 1979.
[19] R. D. Gitlin, J. E. Mazo, and M. G. Taylor, "On the design of gradient algorithms for digitally implemented adjustment filters," *IEEE Trans. Circuit Theory*, vol. CT-20, pp. 125–136, Mar. 1973.
[20] B. Widrow, "Adaptive filters," in *Aspects of Network and System Theory*, R. Kalman and N. De Claris, Eds., New York: Holt, Rinehart, and Winston, 1971, pp. 563–587.
[21] G. Ungerboeck, "Theory on the speed of convergence in adaptive equalizers for digital communication," *IBM J. Res. and Dev.*, pp. 546–555, Nov. 1972.



Michael L. Honig was born in Phoenix, AZ, on December 23, 1955. He received the B.S. degree from Stanford University, Stanford, CA, in 1977, and the M.S. degree from the University of California, Berkeley, in 1978, both in electrical engineering. He is currently pursuing the Ph.D. degree at the University of California, Berkeley.

His research interests include adaptive filtering and its application to speech processing.

Mr. Honig is a member of Tau Beta Pi and Phi Beta Kappa.

David G. Messerschmitt (S'65–M'78–SM'78) received the B.S. degree from the University of Colorado in 1967, and the M.S. and Ph.D. degrees from the University of Michigan in 1968 and 1971, respectively.

He is an Associate Professor of Electrical Engineering and Computer Sciences at the University of California at Berkeley. From 1968 to 1974 he was a Member of Technical Staff and from 1974 to 1977 a Supervisor at Bell Laboratories, Holmdel NJ, where he did systems engineering, development, and research on digital transmission lines and terminals, digital speech interpolation, and digital signal processing, particularly as it relates to both low and high rate encoding of speech. His current research interests are analog and digital signal processing, with applications to voice encoding; digital transmission; phase-locked loops; multiprocessor approaches to signal processing and circuit simulation; and adaptive filtering. He has published 35 papers and has 8 patents issued or pending in these fields. Since 1977, he has served as a consultant to industry, including among other companies TRW Vidar, Hughes Aircraft, Acumenics, IBM, Intelsat, and Intel.

Dr. Messerschmitt is a member of Eta Kappa Nu, Tau Beta Pi, and Sigma Xi. He is currently serving as Editor for Transmission for the IEEE Transactions on Communications.