A Fast Recursive Least Squares Adaptive Second-Order Volterra Filter and Its Performance Analysis

Junghsi Lee, *Member, IEEE*, and V. John Mathews, *Senior Member, IEEE*

***Abstract*—This paper presents a fast, recursive least squares (RLS) adaptive nonlinear filter. The nonlinearity is modeled using a second-order Volterra series expansion. The structure presented in the paper makes use of the ideas of fast RLS multichannel filters and has a computational complexity of $O(N^3)$ multiplications per time instant where $N - 1$ represents the memory span in number of samples of the nonlinear system model. This compares with $O(N^6)$ multiplications required for direct implementation. A theoretical performance analysis of the steady-state behavior of the adaptive filter operating in both stationary and nonstationary environments is presented in the paper. The analysis shows that, when the input is zero mean, Gaussian distributed, and the adaptive filter is operating in a stationary environment, the steady-state excess mean-squared error due to the coefficient noise vector is independent of the statistics of the input signal. The results of several simulation experiments are included in the paper. These results show that the adaptive Volterra filter performs well in a variety of situations. Furthermore, the steady-state behavior predicted by the analysis is in very good agreement with the experimental results.**

I. Introduction

THE concept of linear filtering has had tremendous impact on the development of various techniques for processing stationary and nonstationary signals. The obvious advantage of linear filters is their inherent simplicity. However, in many situations, the performance of the linear filters may be totally unacceptable. A simple example is that of trying to relate two signals whose significant frequency components do not overlap in the frequency domain [24].

System analysis using nonlinear structures has several applications. High-speed communication channels usually need nonlinear equalizers for acceptable performance. For example, Lucky [29] has conjectured that error probability performance of data transmission systems operating at rates higher than 4800 b/s is almost entirely due to nonlinear distortion. In telephone transmission, nonlinearities arise mainly from inaccuracies in signal companding. In digital satellite links, the satellite amplifiers are usually driven to near the saturation point and exhibit highly nonlinear characteristics. Several researchers have used Volterra series representation [39], [41] of nonlinear systems to implement nonlinear channel equalizers [4], [5]. Other applications of nonlinear modeling and filtering in communication problems include echo cancellation [9], [43], [46], performance analysis of data transmission systems [31], noise cancellation [13], and detection of nonlinear functions of Gaussian processes [22]. Nonlinear filters have also been successfully employed in modeling biological phenomenon [19], modeling drift oscillations in random seas [24], myoelectric signal processing [20], image processing [37], characterization of semiconductor devices [21], [35], [36], and several other areas.

Unlike the case of linear systems which are completely characterized by the system's unit impulse response function, it is impossible to find a unified framework for describing arbitrary nonlinear systems. Consequently, the researchers working on nonlinear filters are forced to restrict themselves to certain nonlinear system models that are less general. In this paper, we will concentrate on system representations using a second-order Volterra series expansion. The Volterra system model is extremely popular in adaptive nonlinear filtering and has developed an identity of its own in the last few years.

In the Volterra series representation of systems, which is an extension of linear system theory, the output $y(n)$ of any causal, discrete-time, time-invariant nonlinear system can be represented as a function of the input sequence $x(n)$

$$y(n) = h_0 + \sum_{m_1=0}^{\infty} h_1(m_1)x(n-m_1) + \sum_{m_1=0}^{\infty}\sum_{m_2=0}^{\infty} h_2(m_1, m_2)x(n-m_1)x(n-m_2) + \cdots + \sum_{m_1=0}^{\infty}\cdots\sum_{m_p=0}^{\infty} h_p(m_1, m_2, \cdots, m_p) \cdot x(n-m_1)\cdots x(n-m_p) + \cdots \quad (1)$$

Manuscript received January 8, 1991; revised April 17, 1992. This work was supported in part by the National Science Foundation under Grants MIP-8708970 and MIP-8922146, and by a University of Utah Faculty Fellow Award. Part of this paper was presented at the IEEE International Conference on Acoustics, Speech, and Signal Processing, New York, April 1988.

J. Lee was with the Department of Electrical Engineering, University of Utah, Salt Lake City, UT 84112. He is now with the Advanced Technology Center, Computer and Communications Research Laboratories, Industrial Technology Research Institute, Chutung, Hsinchu, Taiwan 310.

V. J. Mathews is with the Department of Electrical Engineering, University of Utah, Salt Lake City, UT 84112.

IEEE Log Number 9206027.

1053-587X/93$03.00 © 1993 IEEE

where $h_p(m_1, m_2, \cdots, m_p)$ is the pth order Volterra kernel [39], [41], [42] of the system. Without any loss of generality, one can assume that the Volterra kernels are symmetric, i.e., $h_p(m_1, m_2, \cdots, m_p)$ is left unchanged for any of the p! permutations of the indices $m_1, m_2, \cdots, m_p$. One can think of the Volterra series expansion as a Taylor series expansion with memory.

Possibly because the extremely complex nature of the nonlinear filters, very little work has been done in adaptively tracking the time-varying coefficients of such filters. Sandberg [40] has shown the existence of locally convergent Volterra series expansions for time-varying nonlinear systems. Marmarelis [32] has studied the problem of identification of nonstationary nonlinear systems from a single time-limited record of the output function when the input is white Gaussian noise. Most of the work in adaptive Volterra filtering was very recent and many of them [4], [9], [13], [14], [24], [43] employed the least mean square (LMS) algorithm or its variations together with the transversal filter structure. Unfortunately, such filters have convergence rates that are too slow to be useful in many applications. An efficient VLSI implementation of gradient adaptive nonlinear filters can be found in [28]. Efficient implementations using distributed arithmetic were presented in [44] and [46]. One effort to speed up the convergence speed of adaptive Volterra filters equipped with stochastic gradient adaptation algorithms is [23], which introduced a second-order adaptive Volterra filter with lattice orthogonalization. However, the structure assumed Gaussian input signals, and experiments have shown that the rate of convergence can be very slow even for Gaussian signals. Other efforts to introduce adaptive nonlinear filters with good convergence properties have been through approximate least squares techniques [15], [48]. Both methods assumed Gaussian input signals and, consequently, do not work well when the probability distribution of the input signal is non-Gaussian.

This paper presents a fast, RLS adaptive second-order Volterra filter. The computational complexity of this filter is an order of magnitude lower than the most efficient previously available RLS techniques. The algorithm exhibits fast convergence and good tracking behavior. We also present a theoretical performance evaluation of the adaptive filter operating in stationary and nonstationary environments. The remainder of this paper is organized as follows. The following section describes the fast RLS adaptive second-order Volterra filter. Section III presents a theoretical performance analysis of the convergence properties of our filter operating in stationary and nonstationary environments. Section IV contains several simulation examples that reveal the good properties of our filter and also validate the performance analysis. A comparison that shows our filter performs better than two other methods available in literature for adaptive Volterra filtering is also given here. Finally, the concluding remarks are made in Section V.

II. The Fast RLS Second-Order Volterra Filter

Let $d(k)$ and $x(k)$ represent the desired response signal and the input signal, respectively, to the adaptive filter. The problem is then to find an exponentially windowed, fast RLS solution for the linear and quadratic coefficients of the adaptive filter that minimizes the cost function

$$J(n) = \sum_{k=0}^{n} \lambda^{n-k} |d(k) - d_n(k)|^2 \tag{2}$$

at each time instant n, where λ is the parameter of the exponential window that controls the rate at which the adaptive filter tracks time-varying parameters. The positive constant λ is less than, and usually close to, unity. The filter output $d_n(k)$ is a second-order Volterra series expansion in the input signal $x(k)$

$$d_n(k) = \sum_{i=0}^{N-1} a_i^*(n)x(k-i) + \sum_{i=0}^{N-1}\sum_{j=i}^{N-1} b_{i,j}^*(n)x(k-i)x(k-j) \tag{3}$$

where the asterisk denotes complex conjugation. In (3), $a_i(n)$ and $b_{i,j}(n)$ are the linear and quadratic coefficients at time n, respectively, of the adaptive filter and $N-1$ represents the order of memory in the adaptive filter.

For simplicity of representation, the matrix notation will be used. The input vector X_k at time k, which has $N(N+3)/2$ elements, is defined as

$$X_k^T = [x(k), x^2(k), x(k)x(k-1), \cdots, x(k) \cdot x(k-N+1), x(k-1), x^2(k-1), x(k-1) \cdot x(k-2), \cdots, x(k-N+1), x^2(k-N+1)] \tag{4}$$

where the superscript T denotes transposition. Also, the $N(N+3)/2 \times 1$ coefficient vector W_n at time n is defined as

$$W_n^T = [a_0(n), b_{0,0}(n), b_{0,1}(n), \cdots, b_{0,N-1}(n), a_1(n), b_{1,1}(n), b_{1,2}(n), \cdots, a_{N-1}(n), b_{N-1,N-1}(n)]. \tag{5}$$

Thus, the main concern of the exponentially weighted LS problem under consideration is to find, at each time n, the optimal coefficient vector W_n that would minimize the cost function

$$J(n) = \sum_{k=0}^{n} \lambda^{n-k} |d(k) - W_n^H X_k|^2 \tag{6}$$

where $(\cdot)^H$ represents the Hermitian transpose of $(\cdot)$. It is easy to show that the optimal solution W_n is given by

$$\hat{W}_n = \Omega_n^{-1} P_n \tag{7}$$

where

$$\Omega_n = \sum_{k=0}^{n} \lambda^{n-k} X_k X_k^H \tag{8}$$

and

$$P_n = \sum_{k=0}^{n} \lambda^{n-k} X_k d^*(k). \tag{9}$$

Here, Ω_n is the LS autocorrelation matrix of the input vector X_n, and P_n is the LS cross-correlation vector between the input vector X_n and the desired response $d(n)$. Direct evaluation of this solution requires $O(N^6)$ multiplications at each instant. Using the matrix inversion lemma, this complexity can be reduced to $O(N^4)$ multiplications per iteration. Previous attempts to further simplify the computational complexity have been made through approximate techniques [15], [48]. Our approach reduces the complexity to $O(N^3)$ multiplications per iteration without resorting to approximations.

The approach employed in the derivations is to consider the nonlinear filtering problem as a multichannel, but linear filtering problem. Note from the definition of the problem that, at time k, the $N + 1$ elements contained in

$$r_{k-1}^T = [x(k-1)x(k-N), x(k-2) \cdot x(k-N), \cdots, x(k-N), x^2(k-N)] \tag{10}$$

are discarded from the input vector X_{k-1} and another set of $N + 1$ elements contained in

$$v_k^T = [x(k), x^2(k), x(k)x(k-1), \cdots, x(k)x(k-N+1)] \tag{11}$$

are added to the remaining elements to form the input data vector X_k. Thus, the adaptive nonlinear filtering problem can be viewed as a multichannel filtering problem with $N + 1$ channels and $N - 1$ delays. However, the number of delay elements in each channel is different and therefore, the fast RLS multichannel adaptive filtering algorithms [8], [10] cannot be used directly in our case. However, the more general technique in [26] can be combined with the algorithms in [8], [10] to obtain a very efficient algorithm for adaptive Volterra filtering.

The "trick" used in all fast RLS adaptive filtering algorithms involves the successful exploitation of the relationships among the forward predictor, the backward predictor, and the gain vector to obtain the relevant update equations. The forward predictor estimates v_k as a linear combination of the elements of X_{k-1}. Similarly, the backward predictor estimates r_{k-1} using X_k. It should be noted that the structure of the predictors is exactly the same as that of the estimator for $d(k)$. Let A_n and B_n be the optimal coefficient matrices (in the LS sense over the observation interval $0 \le k \le n$) for the forward predictor using X_{k-1} and the backward predictor using X_k, respectively. The corresponding prediction error vectors at time k, denoted as $f_n(k)$ and $b_n(k)$, are then defined as

$$f_n(k) = v_k + A_n^H X_{k-1} \tag{12}$$

and

$$b_n(k) = r_{k-1} + B_n^H X_k. \tag{13}$$

Note that A_n and B_n are $(N^2 + 3N)/2 \times (N - 1)$ matrices and that we have used positive signs instead of negative signs in the definition of the prediction errors.

The $(N^2 + 3N)/2 \times 1$ gain vector C_n, defined as

$$C_n = \Omega_n^{-1} X_n \tag{14}$$

plays a crucial role in the development of the coefficient update equations for the adaptive filter coefficients. It is easy to see that C_n may be viewed as the optimal coefficient vector (in the LS sense over the observation interval $0 \le k \le n$) of a transversal filter that estimates the pinning sequence, defined as

$$\Pi_n(k) = \begin{cases} 1; & k = n \\ 0; & k = 0, 1, \cdots, n-1 \end{cases} \tag{15}$$

using X_k. The corresponding estimation error λ_n at time n is given by

$$\gamma_n = 1 - C_n^H X_n. \tag{16}$$

The estimation error, which is usually called the "likelihood variable," is a real-valued scalar and is bounded by zero and one; that is, $0 \le \gamma_n \le 1$.

The coefficient update equations are easily developed using the matrix inversion lemma (for example, see [18]) and the derivations are not repeated here. The relevant equations are

$$A_n = A_{n-1} - C_{n-1} f_{n-1}^H(n) \tag{17}$$

$$B_n = B_{n-1} - C_n b_{n-1}^H(n) \tag{18}$$

and

$$\hat{W}_n = \hat{W}_{n-1} + C_n e_{n-1}^*(n) \tag{19}$$

where

$$e_n(k) = d(k) - \hat{W}_n^H X_k. \tag{20}$$

It should be clear from the above equations that the problem of efficiently updating $\hat{W}_n$ boils down to that of efficiently updating the gain vector C_n.

Update C_n involves the computation of an $((N + 1) + (N^2 + 3N)/2) \times 1$ extended gain vector, $\overline{C}_n$, that is the LS estimator for the pinning vector using $\overline{X}_k$, where

$$\overline{X}_k = \begin{bmatrix} v_k \\ X_{k-1} \end{bmatrix}. \tag{21}$$

The corresponding estimation error at time n is denoted as $\overline{\gamma}_n$. The vector $\overline{X}_k$, which has the same size as $\overline{C}_n$, can be viewed as the augmented input vector for the forward predictor. Note that $[X_k^T, r_{k-1}^T]^T$ is the augmented input vector for the backward predictor.

Appendix A contains the derivation of the following recursion for the extended gain vector:

$$\overline{C}_n = \begin{bmatrix} \mathbf{0} \\ C_{n-1} \end{bmatrix} + \begin{bmatrix} \alpha_n^{-1} f_n(n) \\ A_n \alpha_n^{-1} f_n(n) \end{bmatrix} \tag{22}$$

where α_n is the LS autocorrelation matrix of the forward prediction error sequence $f_n(k)$, i.e.,

$$\alpha_n = \sum_{k=0}^{n} \lambda^{n-k} f_n(k) f_n^H(k). \tag{23}$$

It is also noted that α_n is Hermitian and can be recursively updated as

$$\alpha_n = \lambda \alpha_{n-1} + f_n(n) f_{n-1}^H(n). \tag{24}$$

The equation for updating α_n^{-1} can then be obtained by employing the matrix inversion lemma.

Our next step is to develop an update equation for the gain vector C_n. For most linear adaptive filtering algorithms, the recursion for updating the gain vector utilizes the extended gain vector directly. This is due to the fact that the augmented input vector for the forward predictor is generally the same as the augmented input vector for the backward predictor in those cases. Since $[v_n^T, X_{n-1}^T]^T$ differs from $[X_n^T, r_{n-1}^T]^T$, $\overline{C}_n$ cannot be employed directly for updating C_n. However, observe that the elements which constitute these two augmented input vectors are the same, and therefore, there exists a nonidentity permutation matrix L such that

$$L\overline{X}_k = \begin{bmatrix} X_k \\ r_{k-1} \end{bmatrix}. \tag{25}$$

By employing the idea proposed in [26], we can use $L\overline{C}_n$ for updating C_n. Let m_n and μ_n signify the top $(N^2 + 3N)/2$ and the remaining $N + 1$ elements, respectively, of $L\overline{C}_n$, i.e.,

$$\begin{bmatrix} m_n \\ \mu_n \end{bmatrix} = L\overline{C}_n. \tag{26}$$

A recursive equation for updating C_n can be obtained in the form

$$C_n = (1 - b_{n-1}^H(n)\mu_n)^{-1}[m_n - B_{n-1}\mu_n]. \tag{27}$$

The derivations of this gain vector and some other update equations needed by the algorithm are given in Appendix A. A complete set of recursions as well as the operations count and the size of the vectors and matrices are presented in Table I.

Remark 1: Exact initialization of the algorithm is possible. However, in all the examples presented later, the initialization was done as shown in Table I with $\delta = 0.001$. $\hat{W}_{-1}$ was assigned to be zero vector, although it can be chosen arbitrarily.

Remark 2: One of the major problems with fast RLS algorithms is their poor numerical properties. This algorithm is no exception. Several researchers [2], [11], [27] have investigated the numerical error propagation in RLS adaptive linear filters. Recently, some techniques of stabilizing [6], [45] or rescuing [25] fast RLS algorithms have been introduced. These techniques can also be extended to the nonlinear filter structures. We have employed the rescue device proposed in [25] to alleviate the numerical problems to a large extent. As noted earlier, $0 \leq \gamma_n \leq 1$. It is only because of numerical errors that γ_n can go out of this range. Experiments by several researchers have shown that γ_n usually goes outside the above range right before the onset of numerical instability. Thus, the algorithm is reinitialized whenever γ_n goes below zero or above one. The reinitialization is achieved by resetting all the variables of the adaptive filter to their initial values as given in Table I with the exception that $\hat{W}_n$ is left unchanged. For the simulation examples presented in Section IV, we did not observe problems due to numerical instability for the duration of the simulations. The algorithm is certain to diverge because of numerical problems if run for much longer durations without rescues or other stabilization techniques. In a large number of experiments, the results of which are not reported here, the rescue device of [25] performed adequately most of the time. However, this aspect of the algorithm requires further study.

Remark 3: It is possible to compute the *a priori* backward prediction error vector $b_{n-1}(n)$ much more efficiently in a manner similar to the approach used in [10]. However, our experiments have shown that the version of the algorithm presented in Table I has much better numerical properties.

Remark 4: Extension of our algorithm to higher order Volterra filter and some other nonlinear models is straightforward. In fact, Table I can also be used to represent a fast, RLS, pth order Volterra filter with a memory span of N samples if all the matrices and vectors are defined properly. In this case, the input vector will have $O(N^p)$ elements and $O(N^{p-1})$ elements are replaced at each time. Consequently, this filter requires $O(N^{2p-1})$ multiplications per sample.

III. Performance Analysis

The analysis relies heavily on the methods used in [16], [30] for studying the convergence and tracking ability of RLS linear adaptive filters. The following assumptions will be used to make the analysis tractable.

1) The adaptive filter is operating in the system identification mode, i.e., it is assumed that the desired output $d(n)$ is a noisy measurement of the output of a second-order Volterra system with the same structure as in (3) and the adaptive filter has at least the same number of linear and quadratic coefficients as the unknown system. Define the optimal coefficient vector $W_{\text{opt},n}$ as

$$\begin{aligned} W_{\text{opt},n}^T = [&a_0^o(n), b_{0,0}^o(n), b_{0,1}^o(n), \cdots, b_{0,N-1}^o(n), \\ &a_1^o(n), b_{1,1}^o(n), b_{1,2}^o(n), \cdots, a_{N-1}^o(n), \\ &b_{N-1,N-1}^o(n)] \end{aligned} \tag{28}$$

where $a_i^o(n)$ and $b_{i,j}^o(n)$ are the linear and quadratic coefficients at time n, respectively, of the unknown Volterra system. Assuming that the measurement noise $e_o(n)$ is additive, the desired response signal $d(n)$ can be expressed as

$$d(n) = W_{\text{opt},n}^H X_n + e_o(n). \tag{29}$$

TABLE I
FAST RLS SECOND-ORDER VOLTERRA FILTER

Initialization:

$\alpha_{-1}^{-1} = \delta^{-1} I_{N+1 \times N+1}$ $\quad A_{-1} = 0_{((N^2+3N)/2 \times N+1)}$ $\quad C_{-1} = 0_{((N^2+3N)/2 \times 1)}$

$\gamma_{-1} = 1$ $\quad B_{-1} = 0_{((N^2+3N)/2 \times N+1)}$ $\quad \hat{W}_{-1} = 0_{((N^2+3N)/2 \times 1)}$ †

Equation	Relation	Dimension	Operation Count
(T.0)	X_n is defined as in (4)	$((N^2 + 3N)/2 \times 1)$	N
(T.1)	$f_{n-1}(n) = v_n + A_{n-1}^H X_{n-1}$	$(N + 1 \times 1)$	$(N^3 + 4N^2 + 3N)/2$
(T.2)	$f_n(n) = \gamma_{n-1} f_{n-1}(n)$	$(N + 1 \times 1)$	$N + 1$
(T.3)	$\alpha_n^{-1} = \lambda^{-1} \left[\alpha_{n-1}^{-1} - \dfrac{\alpha_{n-1}^{-1} f_{n-1}(n) f_{n-1}^H(n) \alpha_{n-1}^{-1}}{\dfrac{\lambda}{\gamma_{n-1}} + f_{n-1}^H(n) \alpha_{n-1}^{-1} f_{n-1}(n)} \right]$	$(N + 1 \times N + 1)$	$2N^2 + 7N + 6$
(T.4)	$\bar{\gamma}_n = \gamma_{n-1} - f_n^H(n) \alpha_n^{-1} f_n(n)$	(1×1)	$N^2 + 3N + 2$
(T.5)	$A_n = A_{n-1} - C_{n-1} f_{n-1}^H(n)$	$((N^2 + 3N)/2 \times N + 1)$	$(N^3 + 4N^2 + 3N)/2$
(T.6)	$\bar{C}_n = \begin{bmatrix} \underline{0} \\ C_{n-1} \end{bmatrix} + \begin{bmatrix} \alpha_n^{-1} f_n(n) \\ A_n \alpha_n^{-1} f_n(n) \end{bmatrix}$ ††	$((N + 1) + (N^2 + 3N)/2 \times 1)$	$(N^3 + 4N^2 + 3N)/2$
(T.7)	$\begin{bmatrix} m_n \\ \mu_n \end{bmatrix} = L\bar{C}_n$ †††	$((N^2 + 3N)/2 + (N + 1) \times 1)$	
(T.8)	$b_{n-1}(n) = r_{n-1} + B_{n-1}^H X_n$	$(N + 1 \times 1)$	$(N^3 + 4N^2 + 3N)/2$
(T.9)	$C_n = (1 - b_{n-1}^H(n) \mu_n)^{-1} [m_n - B_{n-1} \mu_n]$	$((N^2 + 3N)/2 \times 1)$	$(N^3 + 5N^2 + 8N + 4)/2$
(T.10)	$\gamma_n = (1 - b_{n-1}^H(n) \mu_n)^{-1} \bar{\gamma}_n$	(1×1)	1
(T.11)	$B_n = B_{n-1} - C_n b_{n-1}^H(n)$	$((N^2 + 3N)/2 \times N + 1)$	$(N^3 + 4N^2 + 3N)/2$
(T.12)	$e_{n-1}(n) = d(n) - \hat{W}_{n-1}^H X_n$	(1×1)	$(N^2 + 3N)/2$
(T.13)	$e_n(n) = \gamma_n e_{n-1}(n)$	(1×1)	1
(T.14)	$\hat{W}_n = \hat{W}_{n-1} + C_n e_{n-1}^*(n)$	$((N^2 + 3N)/2 \times 1)$	$(N^2 + 3N)/2$
			Total: $3N^3 + 16.5N^2 + 26.5N + 13$

† The filter coefficients can be initialized with arbitrary elements.
†† $\underline{0}$ is a vector of $N + 1$ zero elements.
††† m_n has $(N^2 + 3N)/2$ elements and μ_n has $N + 1$ elements.

2) The sole source of nonstationarity is the random behavior of the optimal coefficient vector $W_{\text{opt},n}$. The evolution of $W_{\text{opt},n}$ is described by

$$W_{\text{opt},n} = W_{\text{opt},n-1} + \nu(n-1) \tag{30}$$

where $\nu(n)$ belongs to a stationary, zero-mean, and white vector Gaussian process with covariance matrix $\sigma_\nu^2 I$.

3) The input signal $x(n)$ and the measurement noise $e_o(n)$ belong to zero-mean and jointly stationary random processes. Moreover, $e_o(n)$ is white and independent of $x(n)$.

4) The input pair $\{X_i, d(i)\}$ is independent of $\{X_j, d(j)\}$ for $i \neq j$. This is the commonly used independence assumption which is not true in practice. The justification for using this assumption is that this would make the otherwise cumbersome analysis mathematically tractable and that analyses using this assumption have provided fairly accurate characterizations and useful design rules for a wide variety of adaptive systems, even in applications where the independence assumption is grossly violated [16], [34].

The problem that we consider here is that of finding the steady-state *a priori* excess mean-squared estimation error (MSE). The *a priori* excess MSE $\epsilon(n)$ is defined as

$$\epsilon(n) = E\{|e_{n-1}(n) - e_o(n)|^2\}. \tag{31}$$

Define the coefficient error vector $\tilde{W}(n-1)$ at time n as

$$\tilde{W}(n-1) = W_{\text{opt},n} - \hat{W}_{n-1}. \tag{32}$$

Using (20), (29), and (32), we can express the *a priori* estimation error $e_{n-1}(n)$ as

$$e_{n-1}(n) = \tilde{W}^H(n-1)X_n + e_o(n). \tag{33}$$

One of the consequences of the independence assumption is that $\tilde{W}(n-1)$ and X_n are uncorrelated with each other.

It is easy to see from the above discussion that the *a priori* excess MSE $\epsilon(n)$ is given by

$$\epsilon(n) = E\{X_n^H \tilde{W}(n-1)\tilde{W}^H(n-1)X_n\} \tag{34}$$

$$= \text{tr}\,\{\Omega K(n-1)\} \tag{35}$$

where

$$\Omega = E\{X_n X_n^H\} \tag{36}$$

is the autocorrelation matrix of X_n,

$$K(n) = E\{\tilde{W}(n)\tilde{W}^H(n)\} \tag{37}$$

is a second moment matrix of the coefficient misalignment vector and tr $\{\cdot\}$ denotes trace of the matrix $\{\cdot\}$.

We can show using (14), (19), (20), and (29) that

$$\hat{W}_n = \hat{W}_{n-1} + \Omega_n^{-1} X_n X_n^H (W_{\text{opt},n} - \hat{W}_{n-1}) + \Omega_n^{-1} X_n e_o^*(n). \tag{38}$$

From (30), (32), and (38), we get

$$\tilde{W}_n = (I - \Omega_n^{-1} X_n X_n^H)\tilde{W}_{n-1} - \Omega_n^{-1} X_n e_o^*(n) + \nu(n). \tag{39}$$

Consider a decomposition of the coefficient misalignment vector into two components with the following evolution equations:

$$\tilde{W}_1(n) = (I - \Omega_n^{-1} X_n X_n^H)\tilde{W}_1(n-1) - \Omega_n^{-1} X_n e_o^*(n) \tag{40}$$

and

$$\tilde{W}_2(n) = (I - \Omega_n^{-1} X_n X_n^H)\tilde{W}_2(n-1) + \nu(n). \tag{41}$$

Note that

$$\tilde{W}(n) = \tilde{W}_1(n) + \tilde{W}_2(n). \tag{42}$$

Furthermore $\tilde{W}_1(n)$ does not depend on $\nu(n)$ and $\tilde{W}_2(n)$ does not depend on $e_o(n)$. Consequently, we can consider $\tilde{W}_1(n)$ as the component of the coefficient misalignment vector that is contributed by the measurement noise. Similarly, $\tilde{W}_2(n)$ is contributed by the lag noise which is due to the nonstationarity of the operating environments. On the basis of the above insights, we will initialize $\tilde{W}_2(n)$ to be a zero vector.

Similar to the arguments used by Macchi and Bershad [30], we can argue that $\tilde{W}_1(n)$ and $\tilde{W}_2(n)$ are uncorrelated with each other in the steady state. Furthermore, since the driving signals in both (40) and (41) are zero-mean vectors, both $\tilde{W}_1(n)$ and $\tilde{W}_2(n)$ will have zero-mean values in the steady state. Now it is easy to show that the excess MSE will satisfy the following relationship in the steady state:

$$\epsilon(n) = \text{tr}\,\{\Omega K_1(n)\} + \text{tr}\,\{\Omega K_2(n)\} \tag{43}$$

where

$$K_1(n) = E\{\tilde{W}_1(n)\tilde{W}_1^H(n)\} \tag{44}$$

and

$$K_2(n) = E\{\tilde{W}_2(n)\tilde{W}_2^H(n)\}. \tag{45}$$

In order to evaluate the steady-state values of $K_1(n)$ and $K_2(n)$, we need to make some approximations to the evolution equations (40) and (41). First of all, note that for large values of n

$$E\{\Omega_n\} \approx \frac{\Omega}{1-\lambda}. \tag{46}$$

Let ω_n be a fluctuation matrix with zero-mean elements such that

$$\Omega_n = \frac{\Omega}{1-\lambda} + \omega_n. \tag{47}$$

Similar to the arguments in [30] for the linear case, we can show that for $(1-\lambda)N^2 << 1$ and large values of n, Ω_n is almost deterministic, i.e., the entries of ω_n are very small compared to those of $\Omega/(1-\lambda)$. Then, at steady state,

$$\Omega_n \approx \Omega_{n-1}. \tag{48}$$

This implies that

$$\Omega_n^{-1}\Omega_{n-1} \approx I. \tag{49}$$

Note that Ω_n can be recursively updated as

$$\Omega_n = \lambda\Omega_{n-1} + X_n X_n^H. \tag{50}$$

Substituting $X_n X_n^H$ as in (50) into (40) and (41), respectively, and employing (49) in the resulting equations will give us the following simplified results:

$$\tilde{W}_1(n) \approx \lambda\tilde{W}_1(n-1) - \Omega_n^{-1} X_n e_o^*(n) \tag{51}$$

and

$$\tilde{W}_2(n) \approx \lambda\tilde{W}_2(n-1) + \nu(n). \tag{52}$$

A. Evaluation of the Steady-State Value of tr $\{\Omega K_1(n)\}$

Postmultiplying both sides of (51) with their respective Hermitian transposes and evaluating the statistical expectation of the products result in

$$K_1(n) \approx \lambda^2 K_1(n-1) + E\{\Omega_n^{-1} X_n X_n^H \Omega_n^{-1}\} E\{|e_o(n)|^2\}. \tag{53}$$

Note that we have used the property that $e_o(n)$ and $\tilde{W}_1(n-1)$ are mutually uncorrelated with each other. Assuming that the evolution in (53) converges, the limiting value of $K_1(n)$ can then be written in the form

$$K_1(n) \approx \frac{E\{\Omega_n^{-1} X_n X_n^H \Omega_n^{-1}\}\xi_o}{1-\lambda_2} \tag{54}$$

where

$$\xi_o = E\{|e_o(n)|^2\}. \tag{55}$$

Since Ω_n is almost deterministic, it follows that Ω_n and X_n can be considered to be almost uncorrelated with each other. Thus, we get

$$\text{tr}\,\{\Omega K_1(n)\} \approx \frac{\xi_o \,\text{tr}\,\{E\{\Omega\Omega_n^{-1}\Omega\Omega_n^{-1}\}\}}{1-\lambda^2}. \tag{56}$$

The above formula is further simplified in Appendix B by using an approximate expression for $\Omega\Omega_n^{-1}$. By so doing, we obtain

$$\mathrm{tr}\,\{\Omega K_1(n)\} \approx \frac{1-\lambda}{1+\lambda}\,\xi_o\left(N' + \frac{1-\lambda}{1+\lambda}\cdot\left(E\{(X_n^H\Omega^{-1}X_n)^2\} - N'\right)\right) \quad (57)$$

where N' equals to $(N+3)N/2$. Note that the excess MSE can be explicitly determined for a given input signal statistics.

It is shown in Appendix C that the expectation $E\{(X_n^H\Omega^{-1}X_n)^2\}$ in (57) does not depend on the input signal statistics when the input is a Gaussian process with zero mean. When $x(n)$ belongs to a zero-mean and real Gaussian process, we can show using some lengthy but straightforward calculations that

$$E\{(X_n^T\Omega^{-1}X_n)^2\} = (N^5 + 16N^4 + 77N^3 + 130N^2 + 72N)/(4(N+2)). \quad (58)$$

The corresponding steady-state value of the excess MSE is now

$$\mathrm{tr}\,\{\Omega K_1(n)\} \approx \frac{1-\lambda}{1+\lambda}\,\xi_o\left(\left(\frac{N^2+3N}{2}\right) + \frac{1-\lambda}{4(1+\lambda)}\cdot\left((N^4 + 14N^3 + 47N^2 + 26N + 8) - \frac{16}{N+2}\right)\right). \quad (59)$$

Closed form expression for the excess MSE can be obtained in a similar manner when the input is a zero-mean and complex Gaussian process. It can be seen that the second term within the parenthesis in (59) will be very small compared to the first term for the case that $(1-\lambda)N^2 \ll 1$. Thus, we can approximate $\mathrm{tr}\,\{\Omega K_1(n)\}$ as

$$\mathrm{tr}\,\{\Omega K_1(n)\} \approx \frac{1-\lambda}{1+\lambda}\,\xi_o\left(\frac{N^2+3N}{2}\right) \quad (60)$$

which is similar to the simplified result obtained in [16]. When λ is very close to one, we can see from the above that the steady-state excess MSE is directly proportional to $(1-\lambda)$, the minimum mean-squared estimation error and the number of coefficients. Note that this result is very similar to those available for adaptive linear filters [16].

B. *Evaluation of the Steady-State Value of* $\mathrm{tr}\,\{\Omega K_2(n)\}$

It is now straightforward to show, in a manner similar to the evaluation of $K_1(n)$, that

$$K_2(n) \approx \frac{\sigma_v^2 I}{1-\lambda^2} \quad (61)$$

and that

$$\mathrm{tr}\,\{\Omega K_2(n)\} \approx \frac{\sigma_v^2\,\mathrm{tr}\,\{\Omega\}}{1-\lambda^2} \quad (62)$$

in the steady state. Again, note the similarity between the results for the nonlinear and linear cases [16].

The total steady-state value of the excess MSE may be obtained by summing (59) with (62). As a specific example, when the input signal is a zero-mean real Gaussian process, we have

$$\epsilon(n) \approx \frac{1-\lambda}{1+\lambda}\,\xi_o\left(\left(\frac{N^2+3N}{2}\right) + \frac{1-\lambda}{4(1+\lambda)}\cdot\left((N^4 + 14N^3 + 47N^2 + 26N + 8) - \frac{16}{N+2}\right)\right) + \frac{\sigma_v^2\,\mathrm{tr}\,\{\Omega\}}{1-\lambda^2}. \quad (63)$$

Another interesting and common case is that when λ is very close to 1 and $(1-\lambda)N^2 \ll 1$; we may approximate the total steady-state value of the average excess MSE as

$$\epsilon(n) \approx \frac{1-\lambda}{2}\,\xi_o\left(\frac{N^2+3N}{2}\right) + \frac{\sigma_v^2\,\mathrm{tr}\,\{\Omega\}}{2(1-\lambda)}. \quad (64)$$

It should be noted that the excess MSE contributed by the measurement noise is directly proportioned to $(1-\lambda)$ while the excess MSE contributed by the lag noise is inversely proportioned to $(1-\lambda)$.

Although done here only for a second-order Volterra system model, this analysis can be easily extended to the higher order Volterra system models also.

IV. Experimental Results

In this section, we present the results of several experiments that demonstrate the good properties of the fast RLS adaptive Volterra filter as well as verify the theoretical analysis presented earlier. The adaptive filter was used in the system identification mode in the first two examples. In these examples, the adaptive filter was run with the same structure and the same number of coefficients as that of the system that was to be identified. The last example involves experimenting with the adaptive filter under conditions of model mismatch. All the random processes involved in this section are real. The measurement noise in each example belonged to a pseudorandom, white, Gaussian process with zero mean. Also, all of the results presented are ensemble averages over 50 independent runs.

A. *Example 1*

The main purpose of this example is to validate the performance analysis presented in the previous section. The problem considered here is that of identifying a second-order Volterra system with $N = 10$, in both stationary and nonstationary environments. The coefficients of the unknown Volterra system are given in Table II. Several cases were studied: three forgetting factors (0.9955, 0.9975, and 0.9995) were employed and two types of input signals (white Gaussian and colored Gaussian) were used. The white Gaussian process is of zero mean and variance 0.0577, while the colored Gaussian signal was generated

TABLE II
LINEAR AND QUADRATIC COEFFICIENTS OF THE UNKNOWN SYSTEM USED IN EXAMPLE 1

	a_i^o		$b_{i,j}^o$									
i		$j =$	0	1	2	3	4	5	6	7	8	9
0	−0.052		1.020	−1.812	−0.593	−1.454	2.779	4.258	−1.509	0.538	0.698	0.396
1	0.723			−0.570	−0.145	1.820	−2.608	−0.728	0.812	−2.093	0.753	−0.363
2	0.435				−0.483	−4.023	−1.487	−0.264	1.285	−0.775	−3.498	−2.402
3	−0.196					2.282	−1.383	2.890	1.581	−3.314	0.461	1.647
4	−0.143						−1.155	−1.270	−1.800	−0.349	−1.975	−3.465
5	0.812							−0.234	0.748	0.273	3.941	−1.334
6	0.354								−1.044	0.022	2.926	−3.071
7	0.077									1.459	−0.508	1.720
8	−1.379										0.824	0.648
9	2.251											0.305

TABLE III
COMPARISON OF EXPERIMENTAL AND THEORETICAL VALUES OF EXCESS MEAN-SQUARED ERROR FOR EXAMPLE 1 (INPUT SIGNAL IS WHITE GAUSSIAN)

	Experiment			Experiment 2			Experiment 3†	
λ	ξ_o	Theoretical	Measured	σ_v^2	Theoretical	Measured	Theoretical	Measured
	10^{-1}	1.83×10^{-2}	1.61×10^{-2}	4.0×10^{-4}	3.68×10^{-2}	4.45×10^{-2}	5.51×10^{-2}	6.02×10^{-2}
0.9955	10^{-2}	1.83×10^{-3}	1.60×10^{-3}	2.0×10^{-5}	1.84×10^{-3}	2.22×10^{-3}	3.67×10^{-3}	3.82×10^{-3}
	10^{-3}	1.83×10^{-4}	1.61×10^{-4}	1.0×10^{-6}	9.20×10^{-5}	1.11×10^{-4}	2.75×10^{-4}	2.71×10^{-4}
	10^{-1}	9.26×10^{-3}	8.62×10^{-3}	1.8×10^{-4}	2.97×10^{-2}	3.22×10^{-2}	3.90×10^{-2}	4.07×10^{-2}
0.9975	10^{-2}	9.26×10^{-4}	8.52×10^{-4}	0.6×10^{-5}	9.90×10^{-4}	1.07×10^{-3}	1.92×10^{-3}	1.93×10^{-3}
	10^{-3}	9.26×10^{-5}	8.61×10^{-5}	0.2×10^{-6}	3.30×10^{-5}	3.58×10^{-5}	1.26×10^{-4}	1.22×10^{-4}
	10^{-1}	1.67×10^{-3}	1.66×10^{-3}	1.0×10^{-6}	8.26×10^{-4}	7.62×10^{-4}	2.50×10^{-3}	2.44×10^{-3}
0.9995	10^{-2}	1.67×10^{-4}	1.64×10^{-4}	2.0×10^{-7}	1.65×10^{-4}	1.52×10^{-4}	3.32×10^{-4}	3.16×10^{-4}
	10^{-3}	1.67×10^{-5}	1.65×10^{-5}	4.0×10^{-8}	3.30×10^{-5}	3.05×10^{-5}	4.97×10^{-5}	4.73×10^{-5}

†ξ_o is as in Experiment 1. σ_v^2 is as in Experiment 2.

by processing a pseudorandom, white, Gaussian noise of zero mean and variance 0.0248 with a linear filter whose impulse response h_n was given by

$$h_n = \begin{Bmatrix} 0.9045 & n = 0 \\ 1.0 & n = 1 \\ 0.9045 & n = 2 \\ 0, & \text{otherwise} \end{Bmatrix}. \tag{65}$$

This setup gave an almost unit output power for the unknown system in the stationary cases. The experimental values of the steady-state excess MSE were measured by time averaging the ensemble average of the *a priori* excess MSE over 3000 time samples in the range [17001, 20000]. These are then compared to the theoretical values obtained from (59), (62), or (63), depending on the situation.

Experiment 1: In this experiment, the unknown system was time invariant. The measured excess MSE for several cases (different values of forgetting factor λ and additive noise level ξ_o) are tabulated along with the theoretical values generated from (59) in Tables III and IV for white and colored inputs signals, respectively. Note that the analytical and empirical results show reasonably good match in all cases. The analysis becomes more accurate when λ is very close to one. This is so because many of the approximations become more accurate when λ is close to one.

Experiment 2: This experiment evaluate the performance of the adaptive filter operating in nonstationary environments when there is no measurement noise. The sole source of nonstationarity is the random behavior of the optimal coefficient vector $W_{\text{opt},n}$. The evolution of $W_{\text{opt},n}$ is described by (30), and the coefficients in Table II were used as the initial values of the optimal coefficient process. We have chosen the values for σ_v^2 so that the excess MSE contributed by the measurement noise in Experiment 1 and the excess MSE contributed by the lag noise in this experiment are at least within a factor of ten of each other. The measured excess MSE of several cases (different values of λ and σ_v^2) are compared to the theoretical values generated from (63) in Tables III and IV. Conclusions similar to those of Experiment 1 can be made here also.

Experiment 3: This experiment considered the effect of nonstationarity as well as measurement noise on the performance of the adaptive filter. The coefficient vectors were the same as in Experiment 2. Comparisons of the empirical and analytical excess MSE for several values of λ, ξ_o, and σ_v^2 are made in Tables III and IV. Once again note the close agreement of the experimental and theoretical values. Note that the sum of the excess MSE in Experiments 1 and 2 is approximately equal to the excess MSE in Experiment 3 for all cases. This observation validates our assumption that the contributions to the excess MSE from adaptation and nonstationarity are additive.

TABLE IV
COMPARISON OF EXPERIMENTAL AND THEORETICAL VALUES OF EXCESS MEAN-SQUARED ERROR FOR EXAMPLE 1 (INPUT SIGNAL IS COLORED GAUSSIAN)

	Experimental 1			Experiment 2			Experiment 3†	
λ	ξ_o	Theoretical	Measured	σ_e^2	Theoretical	Measured	Theoretical	Measured
0.9955	10^{-1}	1.83×10^{-2}	1.59×10^{-2}	3.2×10^{-4}	3.62×10^{-2}	4.37×10^{-2}	5.45×10^{-2}	5.94×10^{-2}
	10^{-2}	1.83×10^{-3}	1.58×10^{-3}	1.6×10^{-5}	1.81×10^{-3}	2.19×10^{-3}	3.64×10^{-3}	3.75×10^{-3}
	10^{-3}	1.83×10^{-4}	1.60×10^{-4}	0.8×10^{-6}	9.04×10^{-5}	1.09×10^{-4}	2.73×10^{-4}	2.71×10^{-4}
0.9975	10^{-1}	9.26×10^{-3}	8.57×10^{-3}	1.35×10^{-4}	2.75×10^{-2}	3.00×10^{-2}	3.68×10^{-2}	3.83×10^{-2}
	10^{-2}	9.26×10^{-4}	8.42×10^{-4}	4.5×10^{-6}	9.18×10^{-4}	1.00×10^{-3}	1.84×10^{-3}	1.83×10^{-3}
	10^{-3}	9.26×10^{-5}	8.53×10^{-5}	1.5×10^{-7}	3.06×10^{-5}	3.34×10^{-5}	1.23×10^{-4}	1.20×10^{-4}
0.9995	10^{-1}	1.67×10^{-3}	1.67×10^{-3}	0.8×10^{-6}	8.14×10^{-4}	7.65×10^{-4}	2.48×10^{-3}	2.43×10^{-3}
	10^{-2}	1.67×10^{-4}	1.66×10^{-4}	1.6×10^{-7}	1.63×10^{-4}	1.53×10^{-4}	3.30×10^{-4}	3.13×10^{-4}
	10^{-3}	1.67×10^{-5}	1.64×10^{-5}	3.2×10^{-8}	3.25×10^{-5}	3.06×10^{-5}	4.92×10^{-5}	4.77×10^{-5}

†ξ_o is as in Experiment 1. σ_e^2 is as in Experiment 2.

We have also used the squared norm of the linear and quadratic coefficient error vectors to evaluate the performance of our algorithm. These norms are defined as

$$\|V_L(n)\| = 10 \log \frac{\sum_{i=0}^{N-1} |\hat{a}_i(n) - a_i^o(n)|^2}{\sum_{i=0}^{N-1} |a_i^o(n)|^2} \tag{66}$$

and

$$\|V_Q(n)\| = 10 \log \frac{\sum_{i=0}^{N-1} \sum_{j=i}^{N-1} |\hat{b}_{i,j}(n) - b_{i,j}^o(n)|^2}{\sum_{i=0}^{N-1} \sum_{j=i}^{N-1} |b_{i,j}^o(n)|^2}. \tag{67}$$

Figs. 1 and 2 show the evolution of the squared norms of the linear and quadratic coefficient error vectors generated from Experiments 1 and 3, respectively, when $\lambda = 0.9975$ and the input signal is colored. It is noted that our algorithm exhibits fast convergence behavior and excellent tracking properties as well, even though the adaptive filter used a fairly large number of coefficients in this example.

B. Example 2

This example compares the performance of our filter to two other alternatives, i.e., and LMS filter [24] and a fast Kalman filter algorithm proposed in [15]. We have considered system identification problem in both stationary and nonstationary environments. The system to be identified was a second-order Volterra system described by

$$\begin{aligned} y(n) = &-0.78x(n) - 1.48x(n-1) + 1.39x(n-2) \\ &+ 0.04x(n-3) + 0.54x^2(n) \\ &+ 3.72x(n)x(n-1) \\ &+ 1.86x(n)x(n-2) - 0.76x(n)x(n-3) \\ &- 1.62x^2(n-1) + 0.76x(n-1)x(n-2) \\ &- 0.12x(n-1)x(n-3) + 1.41x^2(n-2) \\ &- 1.52x(n-2)x(n-3) - 0.13x^2(n-3). \end{aligned} \tag{68}$$

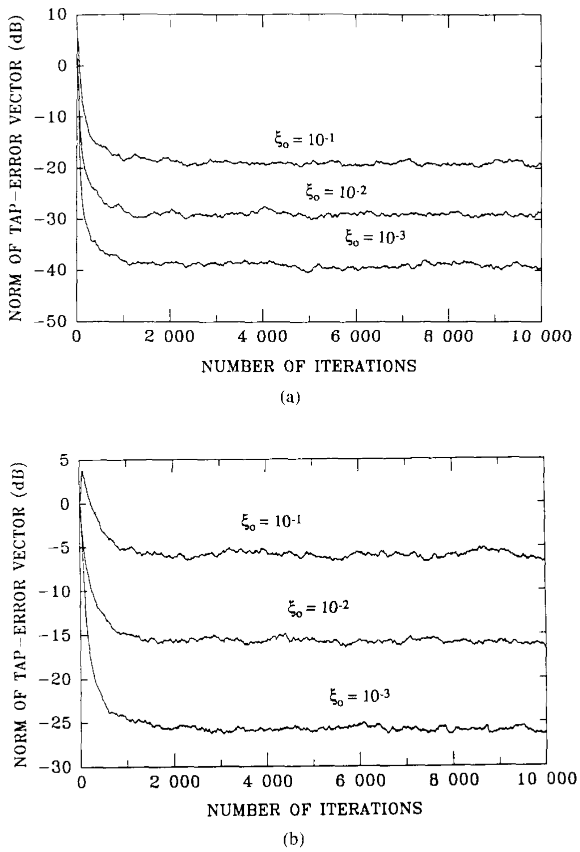

Fig. 1. Norm of coefficient error vector for Experiment 1, $\lambda = 0.9975$, input signal is colored. (a) $\|V_L(n)\|$. (b) $\|V_Q(n)\|$.

We first considered a colored input signal $x(n)$, which was obtained by processing a pseudorandom, white, Gaussian process of zero mean and variance 0.05 with a linear filter given by (65). With this setup, the power of $y(n)$ was about 1. Three different output signal-to-measurement noise ratio (SNR) were considered: 10, 20, and 30 dB. The forgetting factor employed was 0.9966 and the step size of the LMS filter was set to 0.055 so that the theoretical excess MSE's of all the algorithms were about the

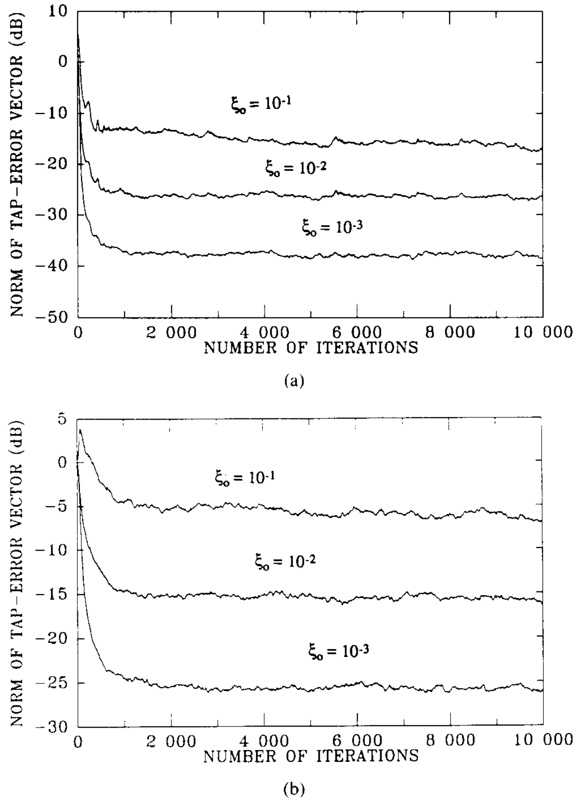

Fig. 2. Norm of coefficient error vector for Experiment 3, $\lambda = 0.9975$, input signal is colored. (a) $\|V_L(n)\|$. (b) $\|V_Q(n)\|$.

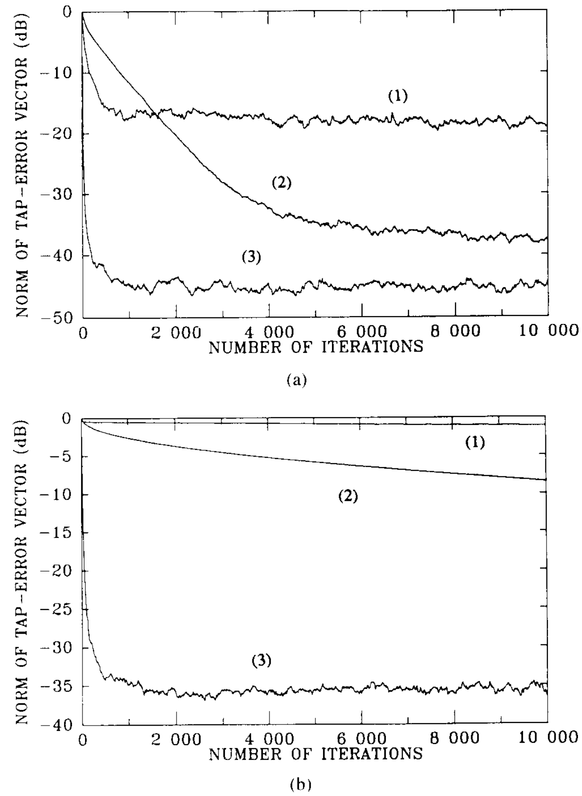

Fig. 3. Norm of coefficient error vector for Experiment 2. Stationary case. (1) Fast Kalman algorithm of [15], (2) LMS algorithm, and (3) fast RLS algorithm of this paper. (a) $\|V_l(n)\|$. (b) $\|V_Q(n)\|$.

same. This experiment was repeated using a white Gaussian input signal. We have again used the squared norm of the coefficient vectors to evaluate the performance of the algorithms. Fig. 3 illustrates the evolution of the squared norms of the linear and quadratic coefficient error vectors corresponding to the three algorithms for the case that the input signal is colored Gaussian and SNR = 30 dB.

The results show that our method performs much better than the other two in terms of convergence speed. The fast Kalman algorithm of [15] has very slow convergence behavior in all cases considered in this example. The convergence speed of this structure can be improved with a small λ. However, results of other experiments that are not included here have shown that small values of λ would result in very severe numerical problems. The LMS algorithms, being a stochastic-gradient technique, has a convergence behavior that is signal dependent: the convergence speed did improve a lot when the input signal is white. The method we have developed, on the other hand, does not have such signal dependence. We have also observed very poor performance of the fast Kalman algorithm of [15] when larger system orders were employed. Hence, we did not include comparison results for larger system orders.

We also evaluate the performance of the adaptive filters operating in nonstationary environments in a way similar to that done in Experiment 3 of Example 1. Fig. 4 illustrates the evolution of the squared norms of the linear and quadratic coefficient error vectors corresponding to the three algorithms for the case that the input signal is colored Gaussian, SNR = 30 dB, and $\sigma_v^2 = 2.0 \times 10^{-7}$. Conclusions similar to those for the stationary case can be made here also.

C. Example 3

This example evaluates the performance of our algorithm when the underlying system model is different from the second-order Volterra system model used in the development of the adaptive filter. The problem considered here is that of identifying a nonlinear channel using the adaptive second-order Volterra filter illustrated in Fig. 5. The nonlinear channel is a simplified model of a digital satellite transmission system [5]. Satellite digital transmission represents one of the most important cases of a digital communication system employing a nonlinear channel. The memoryless nonlinear device is an AM/AM converter whose characteristics are shown in Fig. 6. The transfer functions of the fourth-order low-pass Butterworth and Chebyshev filters, denoted as $H_B(z)$ and $H_C(z)$,

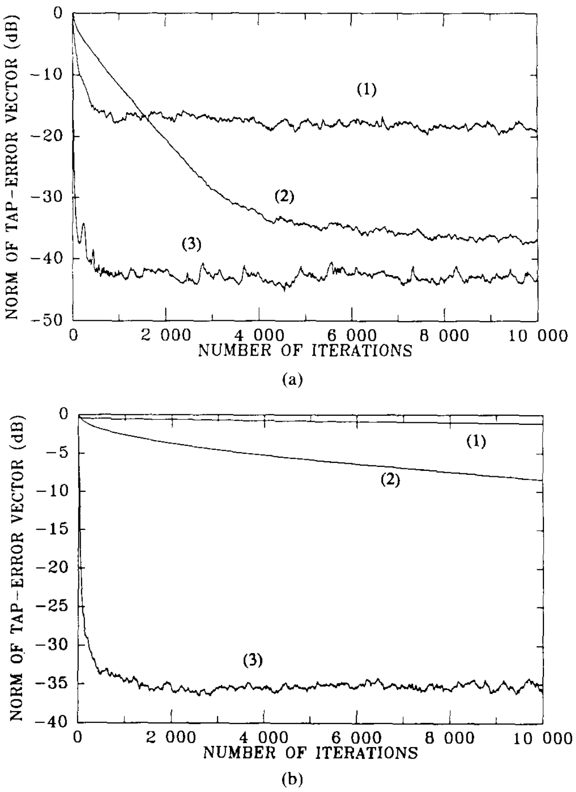

Fig. 4. Norm of coefficient error vector for Experiment 2. Nonstationary case. (1) Fast Kalman algorithm of [15], (2) LMS algorithm, and (3) fast RLS algorithm of this paper. (a) $\|V_L(n)\|$. (b) $\|V_Q(n)\|$.

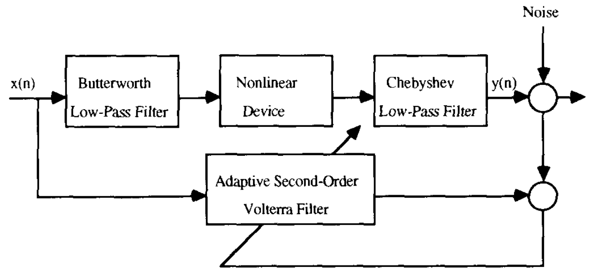

Fig. 5. Application of adaptive filter to identify a nonlinear transmission system.

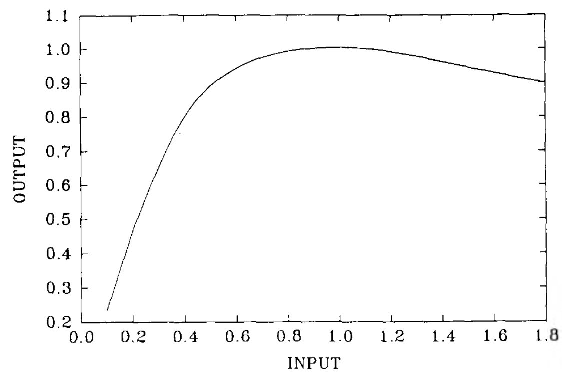

Fig. 6. Input-output characteristics of the AM/AM converter used in Example 3.

are given by

$$H_B(z) = \frac{(0.078 + 0.1559z^{-1} + 0.078z^{-2})\,(0.0619 + 0.1238z^{-1} + 0.0619z^{-2})}{(1.0 - 1.3209z^{-1} + 0.6327z^{-2})\,(1.0 - 1.0486z^{-1} + 0.2961z^{-2})} \tag{69}$$

and

$$H_C(z) = \frac{(0.4638 - 0.4942z^{-1} + 0.4638z^{-2})\,(0.183 + 0.1024z^{-1} + 0.183z^{-2})}{(1.0 - 1.2556z^{-1} + 0.6891z^{-2})\,(1.0 - 0.7204z^{-1} + 0.1888z^{-2})} \tag{70}$$

respectively. Both filters have a cutoff frequency 0.1 cycles/sample. The input signal $x(n)$ was uniformly distributed on the interval [0.12, 1.78] so that the AM/AM converter was operated at saturation region most of the time. With this setup, the power of $y(n)$ was about 1. We have used 0.9985 for λ. The adaptive filter used 11 delays ($N = 12$) in the experiments. This choice was a compromise between performance and complexity. Fig. 7 shows the learning curves associated with the *a priori* MSE over the first 2000 samples. The steady-state MSE's were obtained by time averaging the ensemble averages in the range [9000, 10000] and are given in Table V. It appears that our algorithm works well in this situation even though the structure of the adaptive filter is completely different from that of the system model.

V. Concluding Remarks

This paper presented a fast RLS adaptive second-order Volterra filtering algorithm. The algorithm was derived by exploiting the ideas employed for developing fast RLS adaptive multichannel linear filters. It should be noted that several researchers have studied the multichannel linear filtering problems with arbitrary order in the past few years [1], [17]. However we believe that ours is the first attempt to extend fast, linear filtering algorithms to nonlinear filtering applications. Our adaptive nonlinear filter requires $O(N^3)$ multiplications per sample, which represents a substantial saving over direct implementations. We also presented an analysis of the performance of the algorithm

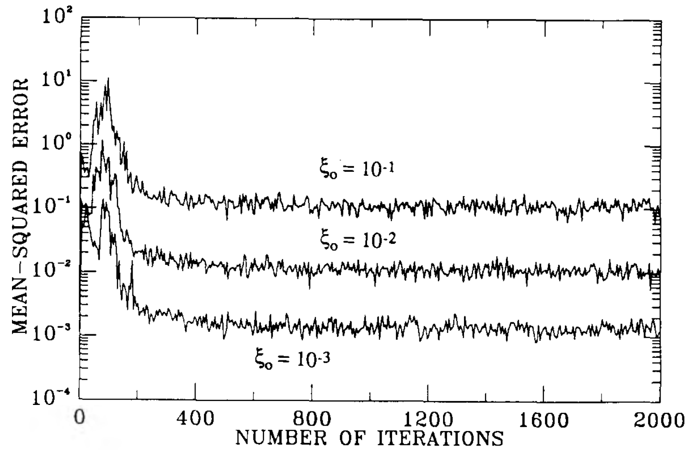

Fig. 7. Learning curves associated with the *a priori* MSE over the first 2000 samples.

TABLE V
STEADY-STATE MEAN-SQUARED ERROR FOR EXAMPLE 3

	$\xi_o = 10^{-3}$	$\xi_o = 10^{-2}$	$\xi_o = 10^{-1}$
$N = 12$	0.00131	0.01099	0.10623

operating in stationary and nonstationary environments. Comparison of analytical and empirical results demonstrated reasonably good match. The experimental results presented showed that the algorithm works well in several different situations. It also works better than two other competing techniques available in the literature.

In the past few years, several researchers have developed neural networks for modeling nonlinear systems. We have not attempted a comparison of the performance of our algorithm with that of such approaches. Such comparisons, as well as the development of adaptive filters that utilize techniques available in both fields would be of great interest.

There are several other issues that require further study. One major task is to develop numerically stable algorithms. Recent work on fast *QR* algorithms [12] have led to the development of what appears to be numerically stable, fast RLS adaptive linear filters. These results have been recently extended to the nonlinear case [47]. While the indications are that the lattice Volterra filter of [47] is numerically stable, such systems do not directly identify the Volterra system coefficients in an efficient manner. Slock and Kailath have recently proposed a numerically stable fast RLS algorithm [45]. Their algorithm can be extended to the nonlinear case, and we expect that such filters will be numerically stable at least for a limited range of values of λ. The estimation of the order of memory as well as the order of nonlinearity must be investigated. The effects of model mismatch should be further studied. Implementation issues, such as those involving exploitation of parallelisms and modularities in the structure of the adaptive filter, have not been addressed in this paper.

As previously mentioned, extension of the results in this paper to higher order Volterra systems is straightforward. A fast, RLS, Pth order adaptive Volterra filter was shown to require $O(N^{2P-1})$ multiplications per sample. This can be an extremely complex task even for moderately large values of N and P. Consequently, it is important to realize that the usefulness of the algorithm described in this paper is mainly in situations involving mild nonlinearities. Much work investigating possibilities of further simplifications to the fast RLS Volterra filters still needs to be done. Several simplifications are being studied now. One of the methods uses approximate predictors in the RLS filters. Another approach is to use simplified models such as cascades of linear and static nonlinear structures. This has recently led to the development of a fast RLS Hammerstein filter [33], in which the nonlinearity is modeled as a static nonlinearity followed by a linear system. Another promising approach is to use nonlinear models with feedback [3].

APPENDIX A

In this Appendix we will derive the update equations for $\bar{C}_n$, C_n, and some other quantities required for the development of the fast RLS Volterra filter. We first consider some useful interpretations of γ_n, the likelihood variable. Using (12), (16), an (17), $f_n(n)$ and $f_{n-1}(n)$ can be shown to be related as follows:

$$\begin{aligned} f_n(n) &= f_{n-1}(n) + (A_n - A_{n-1})^H X_{n-1} \\ &= f_{n-1}(n) - f_{n-1}(n) C_{n-1}^H X_{n-1} \\ &= f_{n-1}(n)\gamma_{n-1}. \end{aligned} \tag{A.1}$$

Similarly, it is easy to show that

$$b_n(n) = b_{n-1}(n)\gamma_n \tag{A.2}$$

and

$$e_n(n) = e_{n-1}(n)\gamma_n. \tag{A.3}$$

The likelihood variable also plays an important role in the development of the recursions for the gain vectors. Let us first derive an update equation for $\bar{C}_n$. By definition,

$$\bar{\gamma}_n = 1 - \bar{C}_n^H \bar{X}_n \quad (A.4)$$

$$= 1 - \bar{C}_n^H \begin{bmatrix} v_n \\ X_{n-1} \end{bmatrix}. \quad (A.5)$$

Substituting v_n from (12) into (A.5) yields

$$\bar{\gamma}_n = 1 - \bar{C}_n^H \begin{bmatrix} f_n(n) - A_n^H X_{n-1} \\ X_{n-1} \end{bmatrix}. \quad (A.6)$$

Note that the second term of the right side of (A.6) is the LS estimate of $\Pi_n(k)$ using $\bar{X}_k$. Geometrically, this LS estimate is the projection of $\Pi_n(k)$ on the sum of two orthogonal subspaces: one which gives $C_{n-1}^H X_{n-1}$ as the LS estimate of $\Pi_{n-1}(k)$ using X_{k-1} and the other one is spanned by the vector sequence $\{f_n(k),\ k = 0, 1, \cdots, n\}$. Applying the well-known projection theorem to this problem we may write (A.6) as

$$\bar{\gamma}_n = 1 - C_{n-1}^H X_{n-1} - [\alpha_n^{-1} f_n(n)]^H f_n(n) \quad (A.7)$$

$$= \gamma_{n-1} - f_n^H(n)\alpha_n^{-1} f_n(n). \quad (A.8)$$

By comparing the right-hand sides of (A.6) and (A.7), we have a recursion for the extended gain vector $\bar{C}_n$

$$\bar{C}_n = \begin{bmatrix} \mathbf{0} \\ C_{n-1} \end{bmatrix} + \begin{bmatrix} \alpha_n^{-1} f_n(n) \\ A_n \alpha_n^{-1} f_n(n) \end{bmatrix}. \quad (A.9)$$

We want now to derive an update equation for C_n. Since the permutation matrix L is orthogonal, (A.4) can be expressed in the form

$$\bar{\gamma}_n = 1 - \bar{C}_n^H L^H L \bar{X}_n. \quad (A.10)$$

From (25) and (26), we can rewrite (A.10) as

$$\bar{\gamma}_n = 1 - \begin{bmatrix} m_n \\ \mu_n \end{bmatrix}^H \begin{bmatrix} X_n \\ r_{n-1} \end{bmatrix}. \quad (A.11)$$

Substituting r_{n-1} as in (13) into (A.11) gives

$$\bar{\gamma}_n = 1 - \begin{bmatrix} m_n \\ \mu_n \end{bmatrix}^H \begin{bmatrix} X_n \\ b_n(n) - B_n^H X_n \end{bmatrix}. \quad (A.12)$$

Similar to the development of (A.7), we can show that

$$\bar{\gamma}_n = 1 - C_n^H X_n - [\beta_n^{-1} b_n(n)]^H b_n(n) \quad (A.13)$$

where β_n is the LS autocorrelation matrix of the backward prediction error sequence $b_n(k)$, i.e.,

$$\beta_n = \sum_{k=0}^{n} \lambda^{n-k} b_n(k) b_n^H(k). \quad (A.14)$$

Comparing the right-hand sides of (A.12) and (A.13) yields

$$C_n = m_n - B_n \mu_n \quad (A.15)$$

and

$$\mu_n = \beta_n^{-1} b_n(n). \quad (A.16)$$

Substituting B_n as in (18) into (A.15) and solving for C_n yield

$$C_n = (1 - b_{n-1}^H(n)\mu_n)^{-1} [m_n - B_{n-1}\mu_n]. \quad (A.17)$$

From (A.2), (A.13), and (A.16) it follows that

$$\gamma_n = (1 - b_{n-1}^H(n)\mu_n)^{-1} \bar{\gamma}_n. \quad (A.18)$$

The derivation of the algorithm is now complete.

Appendix B

Derivation of (57): It can be seen from (47) that

$$\Omega\Omega_n^{-1} = (1 - \lambda)(I + (1 - \lambda)\,\omega_n \Omega^{-1})^{-1}. \quad (B.1)$$

Since Ω_n is almost deterministic,

$$\|(1 - \lambda)\omega_n \Omega^{-1}\| \ll 1 \quad (B.2)$$

for any consistent matrix norm $\|(\cdot)\|$ and therefore we can approximate $\Omega\Omega_n^{-1}$ as

$$\Omega\Omega_n^{-1} \approx (1 - \lambda)(I - (1 - \lambda)\omega_n \Omega^{-1}). \quad (B.3)$$

This allows one to rewrite (56) as

$$\text{tr}\,\{\Omega K_1(n)\} \approx \frac{1 - \lambda}{1 + \lambda} \xi_o \,\text{tr}\,\{I + (1 - \lambda)^2 E\{\Theta(n)\}\} \quad (B.4)$$

where

$$\Theta(n) = \omega_n \Omega^{-1} \omega_n \Omega^{-1}. \quad (B.5)$$

From (47) and (50), we may write

$$\omega_n = \lambda \omega_{n-1} + X_n X_n^H - \Omega. \quad (B.6)$$

Postmultiplying both sides by Ω^{-1}, we get

$$\omega_n \Omega^{-1} = \lambda \omega_{n-1} \Omega^{-1} + X_n X_n^H \Omega^{-1} - I. \quad (B.7)$$

An expression for $\Theta(n)$ can be obtained by multiplying both sides of (B.7) with themselves. Assuming that the system is at its steady state and recognizing that ω_n is a fluctuation matrix, the quantity $E\{\Theta(n)\}$ can be put in the form

$$E\{\Theta(n)\} \approx \frac{\bar{O}}{1 - \lambda^2} \quad (B.8)$$

where

$$\bar{O} = E\{X_n X_n^H \Omega^{-1} X_n X_n^H \Omega^{-1}\} - I. \quad (B.9)$$

Finally, from (B.4), (B.8), and (B.9), we have

$$\text{tr}\,\{\Omega K_1(n)\} \approx \frac{1 - \lambda}{1 + \lambda} \xi_o \,\text{tr}\,\{I + \frac{1 - \lambda}{1 + \lambda} \bar{O}\}$$

$$\approx \frac{1 - \lambda}{1 + \lambda} \xi_o \Big(N' + \frac{1 - \lambda}{1 + \lambda} \cdot (E\{(X_n^H \Omega^{-1} X_n)^2\} - N')\Big) \quad (B.10)$$

where $N' = (N + 3)N/2$.

Appendix C

In this Appendix, we will show that when $x(n)$ is zero-mean and Gaussian distributed, $E\{(X_n^H \Omega^{-1} X_n)^2\}$ does not depend on the statistics of $x(n)$. Define the $N \times 1$ vector $X_{L,n}$ as

$$X_{L,n}^T = [x(n), x(n-1) \cdots, x(n-N+1)]. \quad (C.1)$$

Let Ω_L denote the statistical autocorrelation matrix of $X_{L,n}$, i.e.,

$$\Omega_L = E\{X_{L,n} X_{L,n}^H\}. \quad (C.2)$$

Note that Ω_L is nonsingular in our case. Let R be a nonsingular upper triangular matrix such that

$$\Omega_L^{-1} = RR^H \quad (C.3)$$

defines the Cholesky decomposition of Ω_L^{-1}. Finally, let us define U_n as

$$U_n = [u_1(n), u_2(n), \cdots, u_N(n)]^T = R^H X_{L,n}. \quad (C.4)$$

It is now easy to show that $\{u_1(n), u_2(n), \cdots, u_N(n)\}$ is a white Gaussian process with zero mean and unit variance. The following derivations rely on this fact and some properties of Kronecker products.

The Kronecker product of a $p \times q$ matrix E and an $m \times n$ matrix F is a $pm \times qn$ matrix defined as

$$E \otimes F = \begin{bmatrix} e_{11}F & \cdots & e_{1q}F \\ \vdots & & \vdots \\ e_{p1}F & \cdots & e_{pq}F \end{bmatrix} \quad (C.5)$$

where e_{ij} is the (i, j)th element of E. Two key properties of Kronecker products that are very useful in our derivations are [7]

$$1) \quad (D \otimes E)(F \otimes G) = DF \otimes EG \quad (C.6)$$

provided the matrices involved satisfy the requirements for multiplication, and

$$2) \quad (D \otimes E)^{-1} = D^{-1} \otimes E^{-1} \quad (C.7)$$

for D and E square and nonsingular.

Using (C.4) and (C.6), we may easily write $U_n \otimes U_n$ as

$$U_n \otimes U_n = (R^H \otimes R^H)(X_{L,n} \otimes X_{L,n}). \quad (C.8)$$

We can combine (C.4) and (C.7) to get

$$\begin{bmatrix} U_n \\ U_n \otimes U_n \end{bmatrix} = \begin{bmatrix} R^H & O^T \\ O & R^H \otimes R^H \end{bmatrix} \begin{bmatrix} X_{L,n} \\ X_{L,n} \otimes X_{L,n} \end{bmatrix} \quad (C.9)$$

where $\mathbf{O}$ is a zero matrix with dimension $N^2 \times N$. Since R is nonsingular, the above transformation is invertible. Notice that some of the elements in $U_n \otimes U_n$ are identical, i.e., both $u_i(n)u_j(n)$ and $u_j(n)u_i(n)$ appear in $U_n \otimes U_n$ whenever $i \neq j$. This also occurs in $X_{L,n} \otimes X_{L,n}$. Now, let $U_{\otimes,n}$ represent the vector obtained by removing those repetitive elements from the left-hand side of (C.9). It is easy to see that there exists an invertible linear transformation Γ such that

$$U_{\otimes,n} = \Gamma X_n. \quad (C.10)$$

This also implies that the linear span of the elements of $U_{\otimes,n}$ is the same as the span of the elements of X_n. Let

$$\Omega_{uu} = E\{U_{\otimes,n} U_{\otimes,n}^H\} \quad (C.11)$$

denote the statistical autocorrelation matrix of $U_{\otimes,n}$. The calculation of Ω_{uu} involves fourth-order moments of $\{u_i(n)\}$. These higher order moments may be evaluated by using the following identities [37]:

$$1) \quad E\{z_1 z_2^* z_3 z_4^*\} = E\{z_1 z_2^*\} E\{z_3 z_4^*\} + E\{z_1 z_4^*\} E\{z_3 z_2^*\} \quad (C.12)$$

when z_1, z_2, z_3, and z_4 are samples of a complex random process with zero-mean and Gaussian envelope, and

$$2) \quad E\{z_1 z_2 z_3 z_4\} = E\{z_1 z_2\} E\{z_3 z_4\} + \{z_1 z_3\} E\{z_2 z_4\} + E\{z_1 z_4\} E\{z_2 z_3\} \quad (C.13)$$

when z_1, z_2, z_3, and z_4 are samples of a real random process with zero mean. Now let Y be an $(N^2 + 3N)/2 \times 1$ vector and D be an $(N^2 + 3N)/2 \times (N^2 + 3N)/2$ diagonal matrix such that the ith element of Y and the (i, i)th element of D are one when the ith entry of $U_{\otimes,n}$ is $u_k^2(n)$ for some k and are zero otherwise. It is straightforward to show that

$$\Omega_{uu} = I + D \quad (C.14)$$

when U_n is complex and

$$\Omega_{uu} = I + D + YY^T \quad (C.15)$$

when U_n is real.

From (C.10) and (C.11), it is easy to show that

$$X_n^H \Omega^{-1} X_n = U_{\otimes,n}^H \Omega_{uu}^{-1} U_{\otimes,n}. \quad (C.16)$$

Hence, we have

$$E\{(X_n^H \Omega^{-1} X_n)^2\} = E\{(U_{\otimes,n}^H \Omega_{uu}^{-1} U_{\otimes,n})^2\}. \quad (C.17)$$

Note that the basic elements that form the vector $U_{\otimes,n}$ are $u_1(n)$, $u_2(n)$, $\cdots$, $u_N(n)$, which are Gaussian distributed, uncorrelated with each other, and have unit variance and zero-mean value. This implies that $E\{(X_n^H \Omega^{-1} X_n)^2\}$ does not depend on the statistics of the input signal.

Acknowledgment

The authors thank Dr. R. T. Short for his help with Example 3.

References

[1] S. H. Ardalan and L. J. Faber, "A fast ARMA transversal RLS filter algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 36, no. 3, pp. 349–358, Mar. 1988.

[2] S. H. Ardalan and S. T. Alexander, "Fixed-point roundoff error analysis of the exponentialy windowed RLS algorithm for time-varying systems," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, no. 6, pp. 770–783, June 1987.

[3] H. K. Baik and V. J. Mathews, "Adaptive lattice bilinear filters," *IEEE Trans. Signal Processing*, to be published 1993.
[4] M. Bellafemina and S. Benedetto, "Identification and equalization of nonlinear channels for digital transmission," in *Proc. of the 1985 IEEE Int. Symp. Circuits Syst.*, Kyoto, Japan, June 1985, pp. 1477-1480.
[5] S. Benedetto, E. Biglieri, and V. Castellini, *Digital Transmission Theory*. Englewood Cliffs, NJ: Prentice-Hall, 1987.
[6] J. L. Botto and G. V. Moustakides, "Stabilizing the fast Kalman algorithms," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 37, no. 9, pp. 1342-1348, Sept. 1989.
[7] J. W. Brewer, "Kronecker products and matrix calculus in system theory," *IEEE Trans. Circuits Syst.*, vol. CAS-25, no. 9, pp. 772-781, Sept. 1978.
[8] G. Carayannis, D. Manolakis, and N. Kalouptsidis, "A unified view of parametric processing algorithms for prewindowed signals," *Signal Processing*, vol. 10, pp. 335-368, 1986.
[9] J. R. Casar-Corredera, M. Garcia-Otero, and A. R. Figueiras-Vidal, "Data echo nonlinear cancellation," in *Proc. 1985 IEEE Int. Conf. Acoust., Speech, Signal Processing*, Tampa, FL, Mar. 1985, pp. 32.4.1-4.
[10] J. M. Cioffi and T. Kailath, "Fast, recursive least squares transversal filters for adaptive filtering," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, no. 2, pp. 304-337, Apr. 1984.
[11] J. M. Cioffi, "Limited-precision effects in adaptive filtering," *IEEE Trans. Circuits Syst.*, vol. CAS-34, no. 7, pp. 821-833, July 1987.
[12] J. M. Cioffi, "The fast adaptive ROTOR's RLS algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. 38, no. 4, pp. 631-653, Apr. 1990.
[13] M. J. Coker and D. N. Simkins, "A nonlinear adaptive noise canceller," in *Proc. 1980 IEEE Int. Conf. Acoust., Speech, Signal Processing*, 1980, pp. 470-473.
[14] C. F. N. Cowan and P. F. Adams, "Nonlinear system modeling: Concept and application," in *Proc. 1984 IEEE Int. Conf. Acoust., Speech, Signal Processing*, San Diego, CA, Apr. 1984, pp. 45.6.1-4.
[15] C. E. Davila, A. J. Welch, and H. G. Rylander, III, "A second-order adaptive Volterra filter with rapid convergence," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-35, no. 9, pp. 1259-1263, Sept. 1987.
[16] E. Eleftheriou and D. D. Falconer, "Tracking properties and steady-state performance of RLS adaptive filter algorithms," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, no. 5, pp. 1097-1109, Oct. 1986.
[17] G. O. Glentis and N. Kalouptsidis, "Fast adaptive algorithms for multichannel LS filtering," in *Proc. 1990 IEEE Int. Symp. Circuits Syst.*, New Orleans, LA, May 1990, pp. 801-804.
[18] S. Haykin, *Adaptive Filter Theory*, second ed. Englewood Cliffs, NJ: Prentice-Hall, 1991.
[19] I. W. Hunter and M. J. Korenberg, "The identification of nonlinear biological systems: Wiener and Hammerstein cascade models," *Biolog. Cybern.*, vol. 55, pp. 135-144, 1986.
[20] S. C. Jacobsen, S. G. Meek, and R. R. Fullmer, "An adaptive myoelectric filter," presented at the *6th IEEE Conf. Eng. Medicine Biology Soc.*, 1984.
[21] A. Javed, B. A. Syrett, and P. A. Goud, "Intermodulation distortion analysis of reflection-type IMPATT amplifiers using Volterra series representation," *IEEE Trans. Microwave Theory Techniques*, vol. MTT-25, no. 9, pp. 729-733, Sept. 1977.
[22] R. J. Kenefic and D. D. Weiner, "Application of the Volterra functional expansion in the detection of nonlinear functions of Gaussian processes," *IEEE Trans. Commun.*, vol. COM-33, no. 3, pp. 276-279, Mar. 1985.
[23] T. Koh and E. J. Powers, "An adaptive nonlinear digital filter with lattice orthogonalization," in *Proc. 1983 IEEE Int. Conf. Acoust., Speech, Signal Processing*, Boston, MA, Apr. 1983, pp. 37-40.
[24] T. Koh and E. J. Powers, "Second-order Volterra filtering and its application to nonlinear system identification," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-33, no. 6, pp. 1445-1455, Dec. 1985.
[25] D. W. Lin, "On digital implementation of the fast Kalman algorithm," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-32, no. 5, pp. 998-1005, Oct. 1984.
[26] L. Ljung, M. Morf, and D. D. Falconer, "Fast calculation of gain matrices for recursive estimation schemes," *Int. J. Contr.*, vol. 27, pp. 1-19, Jan. 1978.
[27] S. Ljung and L. Ljung, "Error propagation properties of recursive least-squares adaptation algorithms," *Automatica*, vol. 21, no. 2, pp. 157-167, 1985.
[28] Y. Lou, C. L. Nikias, and A. N. Venetsanopoulos, "Efficient VLSI array processing structures for adaptive quadratic digital filters," *Circuits, Syst. Signal Processing*, vol. 7, no. 2, pp. 253-273, 1988.
[29] R. W. Lucky, "Modulation and detection for data transmission on the telephone channel," in *New Directions in Signal Processing in Communication and Control*, J. K. Skwirzynski, Ed. Leiden, Holland: Noordhoff, 1975.
[30] O. M. Macchi and N. J. Bershad, "Adaptive recovery of a chirped sinusoid in noise, Part 1: Performance of the RLS algorithm," *IEEE Trans. Signal Processing*, vol. 39, no. 3, pp. 583-594, Mar. 1991.
[31] M. Maqusi, "Performance of baseband digital data transmission in nonlinear channels with memory," *IEEE Trans. Commun.*, vol. COM-33, no. 7, pp. 715-719, July 1985.
[32] V. Z. Marmarelis, "Practicable identification of nonstationary nonlinear systems," *Proc. Inst. Elec. Eng.*, vol. 128, Pt. D, no. 5, pp. 211-214, Sept. 1981.
[33] V. J. Mathews, "A fast recursive least-squares adaptive nonlinear filter," in *Proc. 21st Ann. Asilomar Conf. Syst. Signals, Comput.*, Pacific Grove, CA, Nov. 1987, pp. 156-160.
[34] J. E. Mazo, "On the independence theory of equalizer convergence," *Bell Syst. Tech. J.*, vol. 58, no. 5, pp. 963-993, May-June 1979.
[35] S. Narayanan, "Transistor distortion analysis using Volterra series representation," *Bell Syst. Tech. J.*, vol. 46, pp. 991-1024, May-June 1967.
[36] A. Prochazka and R. Neumann, "High-frequency distortion analysis of a semi-conductor diode for CATV applications," *IEEE Trans. Consumer Electron.*, vol. CE-21, no. 2, pp. 120-129, May 1975.
[37] G. Ramponi and G. L. Sicuranza, "Decision-directed nonlinear filters for image processing," *Electron. Lett.*, vol. 23, no. 23, pp. 1218-1219, Nov. 5, 1987.
[38] I. S. Reed, "On a moment theorem for complex Gaussian processes," *IRE Trans. Inform. Theory*, vol. IT-8, no. 3, pp. 194-195, Apr. 1962.
[39] W. J. Rugh, *Nonlinear System Theory, The Volterra-Wiener Approach*. Baltimore, MD: Johns Hopkins University Press, 1981.
[40] I. W. Sandberg, "On Volterra expansions for time-varying nonlinear systems," *IEEE Trans. Circuits Syst.*, vol. CAS-30, no. 2, pp. 61-67, Feb. 1983.
[41] M. Schetzen, *The Volterra and Wiener Theory of the Nonlinear Systems*. New York: Wiley, 1980.
[42] M. Schetzen, "Nonlinear system modeling based on the Wiener theory," *Proc. IEEE*, vol. 69, no. 12, pp. 1557-1573, Dec. 1981.
[43] G. L. Sicuranza, A. Bucconi, and P. Mitri, "Adaptive echo cancellation with nonlinear digital filters," in *Proc. 1984 IEEE Int. Conf. Acoust., Speech, Signal Processing*, San Diego, CA, pp. 3.10.1-4, Mar. 1984.
[44] G. L. Sicuranza and G. Ramponi, "Adaptive nonlinear digital filters using distributed arithmetic," *IEEE Trans. Acoust., Speech, Signal Processing*, vol. ASSP-34, no. 3, pp. 518-526, June 1986.
[45] D. T. M. Slock and T. Kailath, "Numerically stable fast transversal filters for recursive least-squares adaptive filtering," *IEEE Trans. Signal Processing*, vol. 39, no. 1, pp. 92-114, Jan. 1991.
[46] M. J. Smith, C. F. N. Cowan, and P. F. Adams, "Nonlinear echo cancellers based on transpose distributed arithmetic," *IEEE Trans. Circuits Syst.*, vol. CAS-35, no. 1, pp. 6-18, Jan. 1988.
[47] M. A. Syed and V. J. Mathews, "Lattice and *QR* decomposition-based algorithms for recursive least squares adaptive nonlinear filters," in *Proc. 1990 IEEE Int. Symp. Circuits Syst.*, New Orleans, LA, May 1990, pp. 262-265.
[48] D. H. Youn, K. K. Yu, and V. J. Mathews, "Adaptive nonlinear digital filter with sequential regression algorithm," in *Proc. 22nd Annu. Allerton Conf. Contr. Commun. Computing*, Urbana-Champaign, IL, Oct. 3-5, 1984, pp. 152-161.

Junghsi Lee (M'93) was born in Taiwan, Republic of China, in 1959. He received the B.S. degree in control engineering from the National Chiao-Tung University in 1981, and the M.S. and Ph.D. degrees in electrical engineering from the University of Utah in 1987 and 1992, respectively.

He joined the Industrial Technology Research Institute, Chutung, Hsinchu, Taiwan, in 1992 as a Researcher. His research interests include digital signal processing, nonlinear adaptive filtering algorithms and their applications.

V. John Mathews (S'82–M'84–SM'90) was born in Nedungadappally, Kerala, India, in 1958. He received the B.E. (Hons.) degree in electronics and communication engineering from the University of Madras, India, in 1980, and the M.S. and Ph.D. degrees in electrical and computer engineering from the University of Iowa, Iowa City, in 1981 and 1984, respectively.

From 1980 to 1984 he held a Teaching Research Fellowship at the University of Iowa, where he also worked as a Visiting Assistant Professor with the Department of Electrical and Computer Engineering from 1984 to 1985. Since 1985 he has been with the Department of Electrical Engineering, University of Utah, Salt Lake City, where he is now an Associate Professor. His research interests include adaptive filtering, spectrum estimation, and data compression.

Dr. Mathews is a past Associate Editor of the IEEE TRANSACTIONS ON SIGNAL PROCESSING. He is a member of the Digital Signal Processing Technical Committee of the IEEE Signal Processing Society.