A FAST RECURSIVE LEAST-SQUARES SECOND ORDER VOLTERRA FILTER

V. JOHN MATHEWS AND JUNGHSI LEE

Department of Electrical Engineering
University of Utah
Salt Lake City, Utah 84112

ABSTRACT

This paper presents a fast, recursive least-squares (RLS) adaptive nonlinear filter. The nonlinearity is modelled 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. This compares with $O(N^6)$ multiplications required for direct implementation. Simulation examples in which the filter is employed to identify nonlinear systems using noisy output observations are also presented. Further simplification to the structure through a simplified model is discussed very briefly in the paper.

I. INTRODUCTION

In this paper we present a fast, recursive least-squares (RLS) adaptive nonlinear filter. The nonlinearity employed is that of a second order Volterra series expansion where the input output relationship is given by [15]

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

where x(n) and (y)n are the input and output sequences, respectively, N is the number of delays involved and a_i; i=0, 1, ..., N-1, and $b_{i,j}$; i,j = 0, 1, ..., N-1 are possibly time-varying linear and quadratic coefficients of the nonlinear filter. We will assume without loss of generality that the quadratic coefficients are symmetric (i.e., $b_{i,j} = b_{j,i}$).

System analysis using smaller order Volterra series has several applications. Several researchers have used such nonlinear system representations for nonlinear channel equalization and noise cancellation [3,4], studying nonlinear distortion in electronic devices and communication systems [6,14], performance evaluation of data transmission systems [1,9,12], process control [17] and several other applications.

Possibly because of their high computational complexity, very little work has been done in adaptively tracking time-varying nonlinear system parameters. Many of the past work employ the least mean square (LMS) algorithm [3,4,7,8]. In many applications, the slow convergence of the LMS algorithm is unacceptable. Continuously adaptive RLS second order Volterra filters were studied in [5,18]. However, both the methods assume the structure derived for Gaussian input signals and consequently do not work well when the input probability distribution is non-Gaussian. The method presented here presents an exact, recursive solution to the least squares estimation problem and therefore will work well with any type of input signal.

The rest of the paper is organized as follows. The next section introduces the fast RLS second order Volterra filter. The ideas used for fast RLS multichannel filters are employed in our derivations and they result in a computational complexity that corresponds to $O(N^3)$ multiplications per data sample. The direct solution of the problem requires $O(N^6)$ multiplications. This is a tremendous savings in computations. Simulation examples in which the filter is employed to identify a nonlinear system are presented in Section III. The final section contains the concluding remarks where we briefly discuss some further simplifications to the structure through approximate techniques and also using simpler models.

This work was supported in part by the National Science Foundation under Grant MIP-8708970 and by a University of Utah Research Grant.

II. THE FAST RLS SECOND ORDER VOLTERRA FILTER

Let d(n) and x(n) represent the reference and primary inputs, respectively, to the adaptive filter. Then the problem considered in the paper is that of finding an exponentially windowed, fast RLS solution for the linear and quadratic coefficients of the adaptive filter that minimizes the cost function

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

at each time instant n. In Eq. 2, $\hat{d}_n(k)$ is obtained as

$$\hat{d}_n(k) = \sum_{i=0}^{N-1} \hat{a}_i(n)\, x(k-i) + \sum_{i=0}^{N-1}\sum_{j=i}^{N-1} \hat{b}_{i,j}(n)\, x(k-i)\, x(k-j) \tag{3}$$

Note that we have made use of the assumed symmetry of the quadratic coefficients. (Comparing the right-hand sides of Eqs. 1 and 3, we note that $\hat{a}_i(n)$ is an estimate of a_i while $\hat{b}_{i,j}(n)$ is an estimate of $2b_{i,j}$ if $i \neq j$ and of $b_{i,j}$ if i = j). Also $0 < \lambda \le 1$ is the parameter of the exponential window that controls the rate at which the adaptive filter tracks time-varying parameters.

Let us define the input vector X_n (of size N(N+3)/2) at time n as

$$X_n = \left[x(n), x^2(n), x(n)\, x(n-1), \ldots, x(n)\, x(n-N+1), x(n-1), x^2(n-1), \ldots, x(n+1-N), x^2(n-N+1)\right]^T \tag{4}$$

where $(\bullet)^T$ denotes the transpose of $(\bullet)$. Also, define the coefficient vector W_n at time n as

$$W_n = \left[\hat{a}_0(n), \hat{b}_{0,0}(n), \hat{b}_{0,1}(n), \ldots, \hat{b}_{0,N-1}(n), \hat{a}_1(n), \hat{b}_{1,1}(n)\, \hat{b}_{1,2}(n), \ldots, \hat{a}_{N-1}(n), \hat{b}_{N-1,N-1}(n)\right]. \tag{5}$$

Then, the least-squares problem under consideration is to find at each time n, the optimum coefficient vector W_n that would minimize the cost function

$$J(n) = \sum_{k=0}^{n} \lambda^{n-k}\left[d(k) - W_n^T X_k\right] \tag{6}$$

It is easy to show that the optimal solution to the problem is given by

$$W_{n,opt} = \Omega_n^{-1} P_n \tag{7}$$

where

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

and

CH2561-9/88/0000-1383 $1.00 © 1988 IEEE

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

Direct evaluation of this solution requires $O(N^6)$ multiplications at each instant. Even when the block Toeplitz nature of Ω is taken into account, this requires $O(N^4)$ multiplications per sample. Previous attempts at simplification of the computational complexity have been through approximate techniques [3,4,5,7,8,18]. The results that we present obtains the exact RLS solution using $O(N^3)$ multiplications.

Due to page limitations, only sketches of the derivation will be given. The derivations are based on modifications to fast RLS multichannel filtering algorithms of [2,11].

Note from the definition of the input vector X_k in Eq. 4 that at time n+1, N+1 elements of X_k are replaced by new entries. Let v_{n+1} denote the vector formed by the new elements that appear in X_{n+1} and let r_n be the vector formed by the elements that are removed from X_n at time n+1. Define

$$\overline{X}_{n+1} = \begin{bmatrix} v_{n+1} \\ X_n \end{bmatrix} \tag{10}$$

and let L be a permutation matrix consisting of only ones and zeroes such that

$$L\,\overline{X}_{n+1} = \begin{bmatrix} X_{n+1} \\ r_n \end{bmatrix} \tag{11}$$

Then, the algorithm in Table 1 constitutes the fast RLS second order Volterra filter.

In the algorithm, A_n and B_n are the "predictors" for v_n and r_{n-1} and are of order $(N+3)N/2 \times (N+1)$, $C_N(n)$ is the gain vector of size $(N+3)N/2$ and $C_{N+1}(n)$ is the augmented gain vector that has N+1 additional elements. A count of the multiplications involved will show that the complexity is $O(N^3)$ multiplications per time instant and therefore this algorithm represents a substantial improvement over direct implementations in terms of computational complexity.

III. EXPERIMENTAL RESULTS

The system identification set up in Fig. 1 was used in the experimental results presented here. The results presented are ensemble averages of 20 independent runs using 2,000 samples each. The performance index used here is the norm of tap error vector evaluated separately for the linear and quadratic coefficients. They are defined as

$$\|V_A(n)\| = 10 \log \frac{\sum_{i=0}^{N-1} |\hat{a}_i(n) - a_i|^2}{\sum_{i=0}^{N-1} |a_i|^2} \tag{12}$$

and

$$\|V_B(n)\| = 10 \log \frac{\sum_{i=0}^{N-1} \sum_{j=i}^{N-1} |\hat{b}_{i,j}(n) - (2 - \delta(i-j))\, b_{i,j}|^2}{\sum_{i=0}^{N-1} \sum_{j=0}^{N-1} |b_{i,j}|^2} \tag{13}$$

where $\delta(i-j)$ is the Dirac-delta function defined as

$$\delta(i-j) = \begin{cases} 1 & ; \; i=j \\ 0 & ; \; \text{otherwise} \end{cases} \tag{14}$$

In all our experiments, the filter was initialized using zero values for A_0, B_0, C_0 and W_0. Further $\gamma_n(0)$ was chosen to be 1, and $\alpha_N(0)$ was chosen to be μI where μ is a positive constant and I is the $(N+1) \times (N+1)$ order identity matrix. It is possible to use exact initialization of the adaptive filter, but we did not do so because of the numerical problems associated with exact initialization.

It is well known that fast recursive least-squares algorithms suffer from numerical instability. The algorithm presented in this paper is no exception. In the experimental works, we reinitialized the algorithm every time $\gamma_N(n)$ became negative [2,10]. We have found that this approach works satisfactorily in many situations. Several other techniques for improving the numerical stability are also being studied now.

Example 1: In this example the input signal to the experimental setup was obtained by processing a psuedo random, zero mean, Gaussian sequence with unit variance with a low pass filter. The resulting primary input sequence to the adaptive filter had an eigenvalue spread that was greater than 400. The system to be identified was a second order Volterra system with nine delays (10 linear coefficients a_i; i = 0,1, ...9 and 100 quadratic coefficients $b_{i,j}$; i,j = 0,1,...,9). The norms of the tap error vector are plotted in Figs. 2a and 2b for signal to noise ratios of 10, 20 and 30 dB. The forgetting factor λ used in this experiment was 0.989. We can notice that the adaptive filter performs very well in this application even for those cases where the SNR is relatively low.

Example 2: In this example we consider identifying the coefficients of a second order Volterra system with three delays. The primary input sequence to the adaptive filter is a sum of two sinusoids. There are several applications in which it is required to predict the output of a nonlinear system when the inputs are sinusoidal. The results presented in Figs. 3a and 3b indicate that the fast RLS second order Volterra filter works well even when the spectral contents of the primary input signal exist only at discrete frequencies.

IV. CONCLUDING REMARKS

In this paper, we presented a fast, RLS second order Volterra filter. Exploiting the ideas used for developing fast RLS multichannel linear filters, we were able to obtain an adaptive filter structure that requires $O(N^3)$ multiplications per sample. This complexity represents a substantial saving over direct implementations. The experimental results presented showed that the algorithm works well for different types of input signals. Further, by appropriate initialization of the algorithm and also reinitializing the filter every time $\gamma_N(n)$ became negative, we were able to mitigate the effects of numerical instability associated with the algorithm. Further studies on the properties of the algorithm are required and are being done by the authors.

Several simplifications to the structure are also being studied now. One of them include using a simplified model consisting of a squarer followed by a linear system for the nonlinearity. This is a simple case of the problem discussed in [13] and requires only O(4N) multiplications per data sample. Preliminary results have been very encouraging. Use of approximate predictors in the filter structure is another topic that is being investigated.

REFERENCES

1. S. Benedetto, E. Biglieri and R. Daffara, "Performance of multilevel baseband digital systems in a nonlinear environment," *IEEE Transactions on Communications*, Vol. COM-24, No. 10, pp. 1166-1175, October 1976.,
2. J. M. Cioffi and T. Kailath, "Fast, recursive least-squares transversal filters for adaptive filtering," *IEEE Transactions on Acoustics, Speech, Signal Processing*, Vol. ASSP-32, No. 2, pp. 304-313, April 1984.
3. M. J. Coker and D. N. Simkins, "A nonlinear adaptive noise canceller," *Proceedings 1980 IEEE International Conference on Acoustics, Speech and Signal Processing*, pp. 470-473, 1980.
4. C. F. N. Cowan and P. F. Adams, "Nonlinear system modelling: Concept and application," *Proc. ICASSP*, pp. 456.1-4, San Diego, California, April 1984.
5. C. E. Davila, A. J. Welch and H. G. Rylander, III, "A second-order adaptive Volterra filter with rapid convergency," *IEEE Transactions Acoustics, Speech, Signal Processing*, Vol. ASSP-35, No. 9, pp. 1259-1263, September 1987.

6. A. Javad, B. A. Syrett and P. A. Goud, "Intermodulation distortion analysis of reflection-type IMPATT amplifiers using Volterra series representation," *IEEE Transactions on Microwave Theory and Techniques*, Vol. MTT-25, No. 9, pp. 729-733, September 1977.
7. T. Koh and E. J. Powers, "Second order Volterra filtering and its application to nonlinear system identification," *IEEE Transactions on Acoustics, Speech, Signal Processing*, Vol. ASSP-33, No. 6, pp. 1445-1455, December 1985.
8. T. Koh and E. J. Powers, "An adaptive nonlinear digital filter with lattice orthogonalization," *Proceedings of ICASSP 1983*, pp. 37-40, Boston, 1983.
9. T. L. Lim and J. K. Omura, "Error rate estimates in digital communication over a nonlinear channel with memory," *IEEE Transactions on Communications*, Vol. COM-31, No. 3, pp. 407-412, March 1983.
10. D. W. Lin, "On digital implementation of the fast Kalman algorithm," *IEEE Transactions on Acoustics, Speech, Signal Processing*, Vol. ASSP-32, No. 5, pp. 998-1005, October 1984.
11. L. Ljung, M. Morf and D. Falconer, "Fast calculation of gain matrices for recursive estimation schemes," *International Journal of Control*, Vol. 27, pp. 1-19, January 1978.
12. M. Maqusi, "Performance of baseband digital data transmission in nonlinear channels with memory," *IEEE Transactions on Communications*, Vol. COM-33, No. 7, pp. 715-719, July 1985.
13. V.J. Mathews, "A fast recursive least-squares adaptive nonlinear filter," *Proceedings of 21st Annual Asilomar Conference on Signals, Systems and Computers*, Pacific Grove, California, November 1987.
14. W. Reiss, "Nonlinear distortion analysis of P-I-N diode attenuators using Volterra series representation," *IEEE Transactions on Circuits and Systems*, Vol. CAS-31, No. 6, pp. 535-542, June 1984.
15. M. Schetzen, *The Volterra and Weiner Theory of the Nonlinear Systems*, Wiley, New York, 1980.
16. J. C. Stapleton and S. C. Bass, "Adaptive noise cancellation for a class of nonlinear, dynamic reference channels," *IEEE Transactions on Circuits and Systems*, Vol. CAS-32, No. 2, pp. 143-150, February 1985.
17. R. Takahashi and A. Arakawa, "Real-time application of simultaneous estimation theory for parameters and state to one-dimensional heat conduction," *Nuclear Engineering and Design*, Vol. 81, No. 3, pp. 437-449, 1984.
18. D. H. Youn, K. K. Yu, and V. J. Mathews, "Adaptive nonlinear digital filter with sequential regression algorithm, *Proc. 22nd Annual Allerton Conference on Control Communication and Computing*, pp. 152-161, Urbana, Illinois, October3-5, 1984.

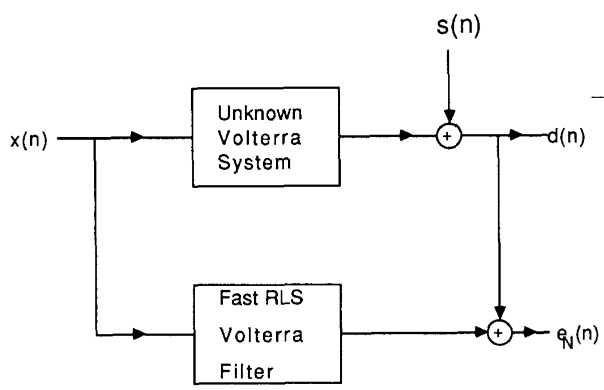

Figure 1. Block diagram for experimental set up

Table 1. Fast RLS Second Order Volterra Filter

$$\eta_N(n) = v_n + A^T_{n-1} X_{n-1} \quad (1.1)$$

$$f_N(n) = \gamma_N(n-1)\,\eta_N(n) \quad (1.2)$$

$$\alpha_N^{-1}(n) = \frac{1}{\lambda}\left[\alpha_n^{-1}(n-1) - \frac{\alpha_N^{-1}(n-1)\,\eta_N(n)\,f_N^T(n)\,\alpha_N^{-1}(n-1)}{\lambda + f_N^T(n)\,\alpha_N^{-1}(n-1)\,\eta_N(n)}\right] \quad (1.3)$$

$$\gamma_{N+1}(n) = \gamma_N(n-1) - f_N^T(n)\,\alpha_N^{-1}(n)\,f_N(n) \quad (1.4)$$

$$A_n = A_{n-1} - C_N(n-1)\,\eta_N^T(n) \quad (1.5)$$

$$C_{N+1}(n) = \begin{bmatrix} 0 \\ C_N(n-1) \end{bmatrix} + \begin{bmatrix} \alpha_N^{-1}(n)\,f_N(n) \\ A_n\alpha_N^{-1}(n)\,f_N(n) \end{bmatrix} \quad (1.6)$$

$$L\,C_{N+1}(n) = \begin{bmatrix} m_n \\ ---- \\ \mu_n \end{bmatrix} \begin{matrix} \}\,N(N+3)/2 \text{ vector} \\ \\ \}\,N+1 \text{ vector} \end{matrix} \quad (1.7)$$

$$\psi_N(n) = r_{n-1} + B^T_{n-1} X_n \quad (1.8)$$

$$\gamma_N(n) = \left[1 - \psi_N^T(n)\,\mu_n\right]^{-1} \gamma_{N+1}(n) \quad (1.9)$$

$$C_N(n) = \left[1 - \psi_N^T(n)\,\mu_n\right]^{-1} \left\{m_n - B_{n-1}\,\mu_v\right\} \quad (1.10)$$

$$B_n = B_{n-1} - C_N(n)\,\psi_N^T(n) \quad (1.11)$$

$$\varepsilon_N(n) = d(n) - W^T_{n-1} X_n \quad (1.12)$$

$$e_N(n) = \gamma_N(n)\,e_N(n) \quad (1.13)$$

$$W_n = W_{n-1} + C_N(n)\,\varepsilon_N(n) \quad (1.14)$$

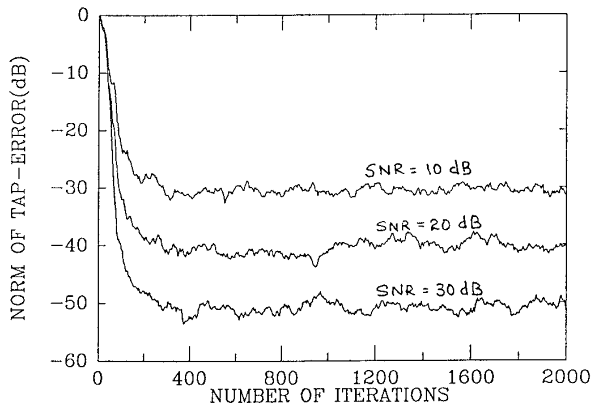

Figure 2a. Norm of linear coefficients error vector for example 1

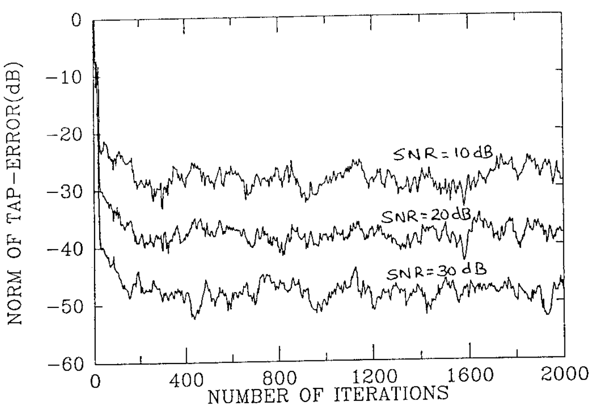

Figure 3a. Norm of linear coefficients error vector for example 2

Figure 2b. Norm of quadratic coefficients error vector for example 1

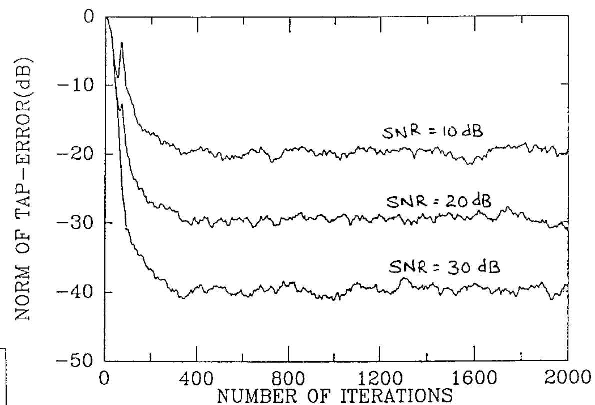

Figure 3b. Norm of quadratic coefficients error vector for example 2