A FAST WEIGHTED SUBBAND ADAPTIVE ALGORITHM

K. Mayyas

Department of Electrical Engg.
Jordan University of Science and Technology
Irbid, Jordan 221 10
mayyas@just.edu.jo

T. Aboulnasr

School of Information Technology and Engg.
University of Ottawa
Ottawa, Ontario, Canada K1N 6N5
aboulnas@eng.uottawa.ca

Abstract

The block algorithm in [1] has illustrated significant improvement in performance over the NLMS algorithm. However, it is known that block processing algorithms have lower tracking capabilities than their sample-by-sample counterparts. The Fast Affine Projection (FAP) algorithm [2] also outperforms the NLMS with a slight increase in complexity, but involves the fast calculation of the inverse of a covariance matrix of the input data that could undermine the performance of the algorithm. In this paper, we present a sample-by-sample version of the algorithm in [1] and develop a low complexity implementation of this algorithm using a similar approach to that in [2]. The new fast algorithm does not require matrix inversion thus alleviating the drawbacks of the FAP algorithm. A variable step size version of the proposed algorithm is also presented.

1 Introduction

With emerging applications requiring adaptive filter orders of several hundreds or thousands , the implementation complexity of fast versions of the RLS algorithm is still highly costly and beyond the capabilities of current DSP processors. Several algorithms have been proposed outperforming the NLMS algorithm in convergence speed with a reasonable increase in computational complexity [2], or with almost equivalent complexity to the NLMS as in [1,3] as a result of performing block adaptation.

The FAP algorithm [2] involves the computation of the inverse of an $N \times N$ input data covariance matrix, where L is the adaptive filter length and N is the value of the projection order of the Affine algorithm. The inverse is calculated using a sliding-window version of the Fast Transversal Filter (FTF), which is known to have numerical instability problems. In addition, when the input signal is highly correlated, the inversion process causes noise amplification resulting in larger misadjustment [4]. This problem appears also in [3] in both the block and sample-by-sample versions of the algorithm. The arithmetic complexity of the sample-by-sample version [3] is in the order of $12L$.

The block processing in [1,3] is advantageous in terms of complexity reduction, however it is known that block adaptation results in lower tracking capabilities compared to its sample-by-sample counterpart.

To solve the above problems, we propose a fast sample-by-sample version of the algorithm in [1]. The proposed algorithm combines the low complexity property of the FAP algorithm and the robust performance of the algorithm in [1]. The complexity of the original proposed algorithm is approximately $O(NL)$. Using the technique in [2], the complexity of our implementation is reduced to $O(2L)$ for the same algorithm. Moreover, we suggest an adaptive step size to the algorithm that enhances its performance by retaining the fast convergence characteristic of the original fast algorithm while providing lower misadjustment. Also, it allows better tracking of sudden changes in the adaptation process.

2 The Fast Weighted Subbands (FWS) Adaptive Algorithm

We start by deriving a fast sample-by-sample version of the algorithm in [1]. Fig.1 shows the structure of the proposed algorithm, which is similar to the one in [1] except that in [1] each subband error $e_i(n)$ is downsampled by N before being used in the update scheme. This follows from the block adaptation operation adopted in [1]. The analysis filters $\mathbf{H}_i$, $i = 1, 2, ..., N$ are assumed to form a perfect reconstruction filter bank. The principal advantage of the structure in Fig.1 is that it updates a full-band adaptive filter as opposed to the traditional subband adaptive filtering techniques that apply individual adaptive filter in each subband. The delay due to the filter bank is also moved to the adaptation loop out of the input signal path. The sample-by-sample algorithm attempts to minimize the instantaneous sum of the weighted square subbands errors, i.e., $\sum_{i=1}^{N} \lambda_i e_i^2(n)$, relative to $\mathbf{W}(n)$. Thus, the update recursion of the sample-by-sample algorithm is

$$\begin{aligned} \mathbf{W}(n+1) &= \mathbf{W}(n) - \frac{\mu}{2}\frac{\partial}{\partial \mathbf{W}(n)} \sum_{i=1}^{N} \lambda_i e_i^2(n) \\ &= \mathbf{W}(n) + \mu \sum_{i=1}^{N} \lambda_i e_i(n) \mathbf{X}_i(n) \quad (1) \end{aligned}$$

where μ is a step size factor, $e_i(n) = \mathbf{H}_i^T \mathbf{e}(n)$, $\mathbf{e}(n) = [e(n)\ e(n-1)\ \ldots\ e(n-K+1)]^T$, K is the length of the analysis filter $\mathbf{H}_i$, and $\mathbf{X}_i(n) = \Phi(n)\mathbf{H}_i$. $\Phi(n)$ is an $L \times K$ matrix defined as $\Phi(n) = [\mathbf{X}(n)\, \mathbf{X}(n-1)\ \ldots\ \mathbf{X}(n-K+1)]$, where $\mathbf{X}(n) = [x(n)\ x(n-1)\ \ldots\ x(n-L+1)]^T$, and L is the length of the adaptive filter. The weighting factor λ_i is chosen as $\lambda_i(n) = \frac{1}{||\mathbf{X}_i(n)||^2+\delta}$, $0 < \delta \ll 1$, to normalize the power in each subband to reduce the eigenvalues disparity of the input data autocorrelation matrix. Define the matrix $H = [\mathbf{H}_1\ \mathbf{H}_2\ \ldots\ \mathbf{H}_N]$, and the diagonal matrix $\Lambda^{-1}(n) = \text{diag}\{\mu\lambda_1(n),\ \mu\lambda_2(n), \ldots,\ \mu\lambda_N(n)\}$, then Eq.(1) can be expressed as

$$\mathbf{W}(n+1) = \mathbf{W}(n) + \Phi(n) H \Lambda^{-1}(n) H^T \mathbf{e}(n) \quad (2)$$

The update algorithm in Eq.(2) requires a total of $L(N+1) + N(2K+1)$ multiplications, and $L(N+1) + 2N(K-1) - 1$ additions excluding the overhead of calculating $\lambda_i(n)$, $i = 1, 2, ..., N$. It is clear that the complexity of the algorithm grows significantly as N increases. We will show now how to reduce the complexity of the sample-by-sample update algorithm in Eq.(2).

Define $\mathbf{P}(n) = H\Lambda^{-1}(n)H^T\mathbf{e}(n)$, where $\mathbf{P}(n) = [\rho_1(n)\ \rho_2(n)\ \ldots\ \rho_K(n)]^T$. The second term in Eq.(2) is given by $\Phi(n)\mathbf{P}(n)$ which can be calculated efficiently using similar approach employed in the FAP algorithm in [1]. Define the intermediate weight vector $\hat{\mathbf{W}}(n)$ such that

$$\hat{\mathbf{W}}(n) = \mathbf{W}(n) - \Phi_{K-1}(n)\mathbf{S}(n-1) \quad (3)$$

where $\Phi(n) = [\mathbf{X}(n)\ \ \Phi_{K-1}(n)]$ and

$$\mathbf{S}(n) = \begin{bmatrix} \rho_1(n) \\ \rho_2(n) + \rho_1(n-1) \\ \vdots \\ \rho_{K-1}(n) + \rho_{K-2}(n-1) + \ldots \\ \ldots + \rho_1(n-K+2) \end{bmatrix} \quad (4)$$

Using Eq.(2) in (3), $\hat{\mathbf{W}}(n)$ can be updated as follows

$$\hat{\mathbf{W}}(n+1) = \hat{\mathbf{W}}(n) + z(n)\mathbf{X}(n-K+1) \quad (5)$$

where $z(n) = \rho_K(n) + \rho_{K-1}(n-1) + \ldots + \rho_1(n-K+1)$. Notice that $\mathbf{S}(n)$ and $z(n)$ can be easily computed recursively as

$$\begin{bmatrix} \mathbf{S}(n) \\ z(n) \end{bmatrix} = \begin{bmatrix} 0 \\ \mathbf{S}(n-1) \end{bmatrix} + \mathbf{P}(n) \quad (6)$$

Since $\hat{\mathbf{W}}(n)$ is already available from Eq.(5), Eq(3) is used to calculate the error $e(n) = d(n) - \mathbf{X}^T(n)\mathbf{W}(n)$ as

$$e(n) = d(n) - \mathbf{X}^T(n)\hat{\mathbf{W}}(n) - \mathbf{V}^T(n)\mathbf{S}(n-1) \quad (7)$$

where $\mathbf{V}(n) = \Phi_{K-1}^T(n)\mathbf{X}(n)$. The quantity $\mathbf{V}(n)$ can be calculated efficiently by

$$\mathbf{V}(n) = \mathbf{V}(n-1) + x(n)\bar{\mathbf{X}}(n-1) - x(n-L)\bar{\mathbf{X}}(n-L-1) \quad (8)$$

where $\bar{\mathbf{X}}(n) = [x(n)\ x(n-1)\ \ldots\ x(n-K+2)]^T$. Table 1 lists the equations along with the number of operations needed by the FWS algorithm at each time step. Note that the FWS algorithm needs the quantities $\mathbf{X}_i(n)$, $i = 1, 2, \ldots, N$ to be available to evaluate $\Lambda^{-1}(n)$. The complexity of the FWS algorithm , excluding that of calculating $\Lambda^{-1}(n)$ (the overhead of computing $\mathbf{X}_i(n)$, $i = 1, 2, \ldots, N$ is included), is $2L+3K(1+N)+N-3$ multiplications and $2L + 3K(1 + N) - 2N - 2$ additions. For example, for $L = 1024$, $N = 16$, $K = 32$, the complexity of the original sample-by-sample algorithm is 18448 multiplies and 18400 adds, while that of the FWS algorithm is 3693 multiplies and 3646 adds. On the other hand, the FAP algorithm requires 2368 multiplies.

Table 1
Computational organization of the fast weighted subbands adaptive algorithm

Equations	Number of mults.	Number of adds.
1. Eq.(8)	$2K-2$	$2K$
2. Eq.(7)	$L+K-1$	$K+L-1$
3. $\mathbf{X}_i(n) = \Phi(n)\mathbf{H}_i$, $i = 1, 2, ..., N$	NK	$N(K-1)$
4. $\mathbf{P}(n) = H\Lambda^{-1}(n)$. $H^T\mathbf{e}(n)$	$2NK+N$	$2NK-N$ $-K$
5. Eq.(6)		$K-1$
6. Eq.(5)	L	L
Total	$2L$ $+3K(1+N)$ $+N-3$	$2L$ $+3K(N+1)$ $-2N-2$

3 A Variable Step Size FWS (VFWS) Algorithm

We suggest here a time-varying step size for the FWS algorithm that follows that of the variable step size algorithm in [5]. The step size control criterion used in [5] was shown to adjust effectively the step size according to the adaptation state while not being affected by independent noise disturbance. The step size update equation is given by

$$\mu(n+1) = \alpha\mu(n) + \gamma p^2(n) \qquad (9)$$

and

$$p(n) = \beta p(n-1) + (1-\beta)e(n)e(n-1) \qquad (10)$$

where limits on $\mu(n+1)$, α, β, and γ are the same as those in [5]. Simulations will show that this technique improves the performance of the FWS algorithm with negligible increase in complexity.

4 Simulations

We examine here the performance of the FWS, VFWS, FAP, and NLMS algorithm. The unknown system to be identified is a 50-coefficient FIR filter, which is a truncation of a 200-tap impulse response of an anechoic room, measured at 8kHz sampling rate. Perfect modeling of the unknown system is assumed, i.e., $L = 50$. The input signal is a highly correlated one generated by passing a zero-mean Gaussian signal with unity variance through the filter $H(z) = \frac{1}{1-1.58z^{-1}+0.8z^{-2}}$. A white zero-mean Gaussian noise is added to the desired signal such that SNR=30dB. Results are obtained by averaging over 100 independent runs.

In this example, the FWS is compared with the FAP and NLMS algorithm. Both the FAP and FWS have $N = 8$. The step sizes used are $\mu_{FWS} = 0.07$, $\mu_{FAP} = 0.09$, and $\mu_{NLMS} = 1$, which are chosen to achieve the same steady state excess MSE of the NLMS algorithm. The analysis filters are cosine modulated perfect reconstruction filter banks with $K = 32$. It is clear from Fig.2 that the FWS operates as well as the FAP algorithm, and outperforms the NLMS.

Fig.3 compares the behavior of the FWS and VFWS for the same above example but with an abrupt change in the unknown system, i.e., all coefficients are multiplied by -1 at iteration 10 000. The VFWS is used with $\beta = 0.99$, $\alpha = 0.97$ [5], $\gamma = 0.1$, $\mu_{max} = 0.08$, and $\mu_{min} = 0.005$. For the FWS, $\mu_{FWS} = 0.007$. The parameters are selected to achieve a steady-state excess MSE of approximately 40dB. As expected, the VFWS algorithm gives the fastest speed of convergence while retaining the same small level of misadjustment, and also maintains the ability to respond fast to changes in the system.

5 Conclusions

In this paper, we have presented a fast weighted subbands adaptive algorithm that leads to considerable improvement over the NLMS algorithm with a reasonable level of complexity. The new algorithm also performs as well as the FAP algorithm for the same N. The arithmetic complexity of the fast algorithm amounts to $2L + 3K(1 + N) + N$ multiplications and $2L + 3K(1 + N) - 2N$ additions. A time-varying step size can easily be incorporated in the fast new algorithm. The performance of the proposed algorithm was compared to that of the FAP and NLMS algorithms. Simulation results confirmed its effectiveness for both fixed and variable step sizes.

6 References

[1] M. Courville, and P. Duhamel, "Adaptive Filtering In Subbands Using a Weighted Criterion", *Proc. IEEE Int. Conf. Acoustic, Speech and Signal Processing*, pp. 985-988, 1995.

[2] S. Gay, and S. Tavathia, "The Fast Affine Projection Algorithm", *Proc. IEEE Int. Conf. Acoustic, Speech and Signal Processing*, pp. 3023-3026, 1995.

[3] M. Montazeri, and P. Duhamel, "A Set of Algorithms Linking NLMS and Block RLS Algorithms", *IEEE Trans. on Signal Processing*, Vol. 43, No. 2, pp. 444-453, Feb. 1995.

[4] K. Maouche, and D. T. Slock, "A Fast Instrumental Variable Affine Projection Algorithm", *Proc. IEEE Int. Conf. Acoustic, Speech and Signal Processing*, pp. 1481-1484, 1998.

[5] T. Aboulnasr and K. Mayyas, "A Robust Variable Step-Size LMS-Type Algorithm: Analysis and Simulations", *IEEE Trans. Signal Processing*, Vol. 45, No. 3, pp. 631-639, March 1997.

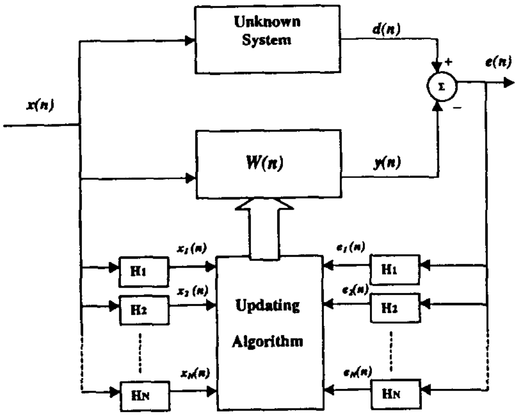

Fig.1 The structure of the sample-by-sample weighted subband adaptive algorithm.

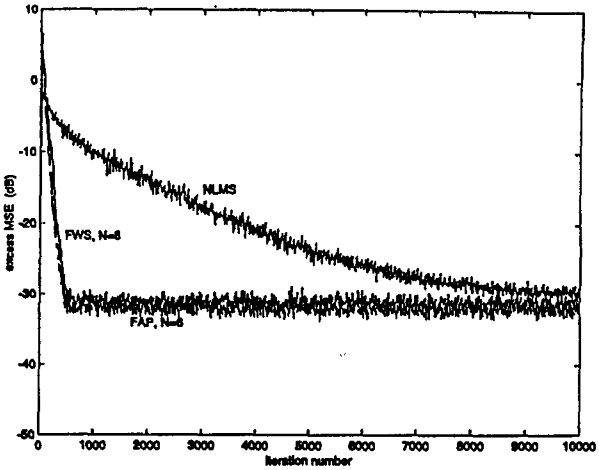

Fig.2 Comparison of excess MSE between the NLMS, FWS, and FAP algorithm for correlated input signal SNR=30 dB.

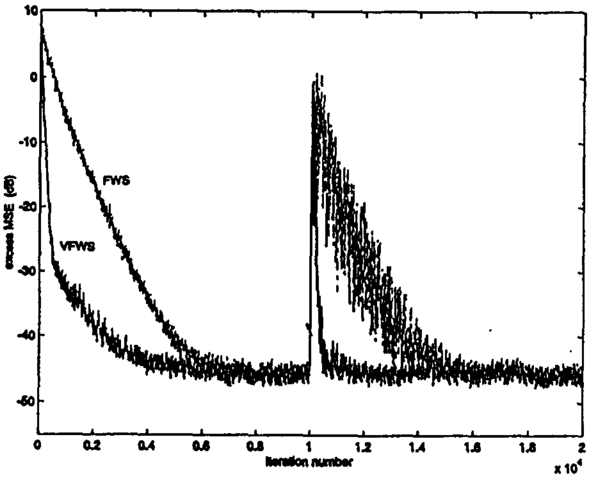

Fig.3 Comparison of excess MSE between the FWS, and VFWS algorithm with $N = 8$ for an abrupt change in the unknown system parameters.