On the Use of A Modified Fast Affine Projection Algorithm in Subbands for Acoustic Echo Cancelation

*Q. G. Liu *, B. Champagne * and K. C. Ho ***

*INRS-Télécommunications, **Nortel
16 Place du Commerce, Verdun, Québec, Canada H3E 1H6
qingliu@inrs-telecom.uquebec.ca

ABSTRACT

The fast affine projection (FAP) has been proposed recently as a new adaptive filtering algorithm which possesses LMS like complexity but may achieve RLS like convergence. In this work, we investigate the use of FAP algorithms in subbands for acoustic echo cancelation (AEC). To this end, some modifications are first made on the FAP based on the consideration of the numerical stability. The modified FAP algorithm is then used in a complex adaptive subband filter structure for AEC. Advantages of the resulting subband FAP over the full-band NLMS scheme in terms of both complexity and performance are demonstrated by simulations.

1. INTRODUCTION

Subband adaptive filtering has been an attractive approach for AEC in recent years. The well-known normalized LMS (NLMS) algorithm has been widely used for the subband adaptive algorithm because of its simplicity. However, in the AEC context, the subband NLMS algorithm exhibits some performance limits due to its slow asymptotic convergence [1].

RLS-based algorithms have been also used in subband AEC to improve the convergence [2]. However, even though fast RLS algorithms are employed, their computational requirements are still too expensive to implement in typical AEC applications where a large number of filter taps are needed. Furthermore, the problem of numerical instability for fast RLS algorithms will become more crucial for parallel subband filters since higher risks will be brought in when multiple algorithms are run synchronously.

The affine projection algorithm (APA) [3] was first proposed as a generalization of the NLMS algorithm. In APA, the weight vector update is obtained from a projection on an affine subspace with dimension $L-N$, where L is the length of the filter and N is an integer. By increasing the value of N to some extent, the convergence speed of the weight vector will be improved. Recently, a fast version of APA, the fast affine projection (FAP), was presented [4]. The complexity of FAP is about $2L+21N$ multiplications per sample, where N can be chosen much smaller than L to achieve considerablely faster convergence than NLMS. However, since a fast RLS algorithm is employed in the FAP, instability will result in finite precision numerical computations.

In this paper, we investigate the use of the FAP algorithm in subbands for the purpose of AEC. We first make some modifications on the original FAP algorithm so as to improve its numerical stability. This modified FAP algorithm is then used in a multirate subband structure for adaptive filtering. The resulting subband echo canceler shows excellent properties in terms of both complexity and convergence performance.

2. THE MODIFIED FAP

2.1. The affine projection algorithm

The affine projection algorithm (APA) and the fast APA algorithm (FAP) were originally derived for real signals. Since the subband analysis filter bank used in this study produces complex outputs, it is necessary to formulate these algorithms in their complex versions.

Let $\mathbf{h}_n \in \mathcal{C}^L$ be the L-dimensional complex filter weight vector at time n. The input signal $x(n)$ is filtered by $\mathbf{h}_{n-1}$ and subtracted from the echo disturbed signal $s(n)$ to get the residual error $e(n)$:

$$e(n) = s(n) - \mathbf{h}_{n-1}^H \mathbf{x}_n \quad (1)$$

where $\mathbf{x}_n = [x(n), x(n-1), ..., x(n-L+1)]^T$ and the superscripts ^{T} and ^{H} denote transposition and complex conjugate transposition, respectively. Further define an $L \times N$ signal matrix X_n as:

$$X_n = [\mathbf{x}_n, \mathbf{x}_{n-1}, ..., \mathbf{x}_{n-(N-1)}], \quad (2)$$

and let $\mathbf{s}_n = [s(n), s(n-1), ..., s(n-N+1)]^T$. The complex version of the APA can be expressed as:

$$\mathbf{e}_n = \mathbf{s}_n - X_n^T \mathbf{h}_{n-1}^* \quad (3)$$

$$P_n = [X_n^H X_n + \delta I]^{-1} \quad (4)$$

$$\epsilon_n = \mu P_n \mathbf{e}_n^* \quad (5)$$

$$\mathbf{h}_n = \mathbf{h}_{n-1} + X_n \epsilon_n \quad (6)$$

where δ serves as the regularization parameter in computing the inverse of the signal autocorrelation matrix

(4), μ is a scalar called the relaxation factor and the superscript * denotes the complex conjugate. The weight vector $\mathbf{h}_n$ will converge when $0 \leq \mu < 2$. Note that when $N = 1$, $\mathbf{e}_n$ reduces to $e(n)$ and the above algorithm becomes exactly the NLMS algorithm.

The complexity of the APA is $2LN + O(N^2)$ multiplications per iteration. In [4], a (computationally) fast algorithm for APA, the FAP, is developed that reduces this complexity to $2L + 21N$. Some of the most important ideas used in the derivation of the FAP algorithm are summarized below.

The first one is to update $\mathbf{e}_n$ via the following approximation:

$$\mathbf{e}_n = \begin{bmatrix} e(n) \\ (1-\mu)\bar{\mathbf{e}}_{n-1} \end{bmatrix} \tag{7}$$

where $\bar{\mathbf{e}}_{n-1}$ consists of the upper $N-1$ elements of $\mathbf{e}_{n-1}$. This approximation is valid for reasonable choices of the value of δ.

The second distinguishing feature of the FAP algorithm is the use of the sliding window fast RLS algorithm to calculate the forward and backward linear predictors associated with the inverse of the matrix P_n (4), such that ϵ_n (5) can be calculated efficiently.

Another interesting idea in the FAP algorithm is to avoid computing the weight vector $\mathbf{h}_n$ (6), which is not the main concern in the AEC application. Instead, another vector $\hat{\mathbf{h}}_n$ is introduced which is connected to $\mathbf{h}_n$ through the following equation:

$$\mathbf{h}_n = \hat{\mathbf{h}}_n + \mu \bar{X}_n \bar{\eta}_n \tag{8}$$

where $\bar{X}_n$ consists of the $N-1$ left-most columns of X_n and $\bar{\eta}_n$ consists of the upper $N-1$ element of η_n, which is an intermediate vector in C^N and is updated as:

$$\eta_n = \begin{bmatrix} 0 \\ \bar{\eta}_{n-1} \end{bmatrix} + \epsilon_n. \tag{9}$$

The vector $\hat{\mathbf{h}}_n$ can also be computed recursively:

$$\hat{\mathbf{h}}_n = \hat{\mathbf{h}}_{n-1} + \mu \eta_{n,N} \mathbf{x}_{n-(N-1)} \tag{10}$$

where $\eta_{n,N}$ is the Nth element of η_n.

Finally, the computation of the residual signal $e(n)$ can be obtained by substituting (8) into (1):

$$e(n) = s(n) - \hat{\mathbf{h}}_{n-1}^H \mathbf{x}_n - \mu \bar{\eta}_{n-1}^H \bar{X}_{n-1}^H \mathbf{x}_n. \tag{11}$$

Further simplifications in computing the last term on the right-hand side of (11) can be made by introducing the vector $\mathbf{r}_n = \bar{X}_{n-1}^H \mathbf{x}_n$, which can be recursively calculated as:

$$\mathbf{r}_n = \mathbf{r}_{n-1} + x(n)\bar{\xi}_{n-1}^* - x(n-L)\bar{\xi}_{n-L-1}^* \tag{12}$$

where $\bar{\xi}_n$ consists of the upper $N-1$ elements of the N-dimensional input signal vector ξ_n:

$$\xi_n = [x(n), x(n-1), ..., x(n-N+1)]^T \tag{13}$$

Note that ξ_n differs from $\mathbf{x}_n$ only by its length.

2.2. Modifications on the FAP

In the FAP algorithm described above, the sliding window fast RLS algorithm is needed to update P_n (4) and ϵ_n (5). As we mentioned before, the fast RLS algorithm is sensitive to finite precision effects. The problem of instability becomes more crucial for a set of subband filters since higher risks will be brought in when multiple algorithms are run synchronously. In our work, we propose using the well-known matrix inverse lemma to update P_n (4), or equivalently, using the conventional RLS algorithm, instead of using a fast version of the latter.

To this end, we express the inverse of P_n in (4) as:

$$P_n^{-1} = \sum_{i=n-(L-1)}^{n} \xi_i^* \xi_i^T + \delta I \tag{14}$$

A recursive expression for P_n^{-1} can be easily found as:

$$P_n^{-1} = Q_n - \xi_{n-L}^* \xi_{n-L}^T \tag{15}$$

$$Q_n = P_{n-1}^{-1} + \xi_n^* \xi_n^T \tag{16}$$

Using the matrix inverse lemma in (15), we have

$$P_n = Q_n^{-1} - \beta \mathbf{b}_n \mathbf{b}_n^H \tag{17}$$

where $\mathbf{b}_n = Q_n^{-1}\xi_{n-L}^*$, $\beta = (-1 + \xi_{n-L}^T \mathbf{b}_n)^{-1}$ and Q_n^{-1} can be obtained by applying the matrix inverse lemma again to (16):

$$Q_n^{-1} = P_{n-1} - \alpha \mathbf{a}_n \mathbf{a}_n^H \tag{18}$$

where $\mathbf{a}_n = P_{n-1}\xi_n^*$ and $\alpha = (1 + \xi_n^T \mathbf{a}_n)^{-1}$.

The complexity of the above approach for updating P_n is $3N^2 + 7N$ complex multiplications per iteration. Since the value of N will be chosen much smaller than L in the present AEC application, this will amount to only a small part of the total computation cost.

Note that P_n^{-1} is an $N \times N$ sample covariance matrix of the input signal based on a rectangular sliding window of length L. Since the exponential window is often used in the estimation of the signal covariance matrix, the following alternate form of P_n^{-1} is proposed in [5]:

$$\hat{P}_n^{-1} = \Sigma_{i=0}^{n} \lambda^{n-i} \xi_i^* \xi_i^T + \delta I \tag{19}$$

where $0 < \lambda < 1$ is the forgetting factor of the exponential window. A similar recursive algorithm as above for updating $\hat{P}_n$ (19) can be obtained by using the matrix inverse lemma (but only once). The use of an exponential window as in (19) is supposed to improve the conditioning factor of the signal covariance matrix so as to increase the stability of the matrix inverse [5]. However, in the subband filtering application of FAP considered in this paper, we prefer not to use the exponential windowing form (19) because of the difficulties of selecting proper values of λ for the different subbands, as observed in our simulation work.

Once P_n is obtained, ϵ_n can be calculated as in (5). If μ is set to 1 in (7), a value often used in full-band adaptive filters, (5) can be simplified as:

$$\epsilon_n = \mu e(n)^* P_{n,1} \quad (20)$$

where $P_{n,1}$ is the first column of P_n. We have found experimentally that (20) can still be used to compute ϵ_n when μ is slightly different from 1 (say 0.7) with little effects on the results.

The modified (complex) FAP algorithm that we propose to use in this study can now be summarized as follows:

0. Initialization:

$$P_0 = \delta^{-1} I, \quad \mathbf{r}_0 = \mathbf{0}, \quad \eta_0 = \mathbf{0} \quad (21)$$

1. Update P_n:

$$\mathbf{a}_n = P_{n-1}\xi_n^* \quad (22)$$

$$\alpha = (1 + \xi_n^T \mathbf{a}_n)^{-1} \quad (23)$$

$$\mathbf{q}_n = P_{n-1}\xi_{n-L}^* \quad (24)$$

$$\mathbf{b}_n = \mathbf{q}_n - \alpha(\mathbf{a}_n^H \xi_{n-L}^*)\mathbf{a}_n \quad (25)$$

$$\beta = (-1 + \xi_{n-L}^T \mathbf{b}_n)^{-1} \quad (26)$$

$$P_n = P_{n-1} - \alpha \mathbf{a}_n \mathbf{a}_n^H - \beta \mathbf{b}_n \mathbf{b}_n^H \quad (27)$$

2. Compute $e(n)$:

$$\mathbf{r}_n = \mathbf{r}_{n-1} + x(n)\bar{\xi}_{n-1}^* - x(n-L)\bar{\xi}_{n-L-1}^* \quad (28)$$

$$e(n) = s(n) - \hat{\mathbf{h}}_{n-1}^H \mathbf{x}_n - \mu \bar{\eta}_{n-1}^H \mathbf{r}_n \quad (29)$$

3. Update $\hat{\mathbf{h}}_n$ and η_n:

$$\epsilon_n = \mu e(n)^* P_{n,1} \quad (30)$$

$$\eta_n = \begin{bmatrix} 0 \\ \bar{\eta}_{n-1} \end{bmatrix} + \epsilon_n \quad (31)$$

$$\hat{\mathbf{h}}_n = \hat{\mathbf{h}}_{n-1} + \mu \eta_{n,N} \mathbf{x}_{n-(N-1)} \quad (32)$$

The total complexity of the above algorithm is $2L + 3N^2 + 12N$ multiplications per iteration. This amount is close to that of the NLMS algorithm when N is much smaller than L, as is the case in our applications.

3. USING THE MODIFIED FAP ALGORITHM IN SUBBANDS

3.1. Subband FAP

The input signal $x(n)$ and the echo signal $s(n)$ are all split into K subband signals by the analysis filter banks. Each subband signal is then decimated by a factor $M \leq K$. In our application, uniform DFT banks are used for the analysis/synthesis system. These are implemented by the weighted overlap-add method [6], such that the subband signals at an arbitrary down-sampling rate M can be produced efficiently. A suitable choice of M is somewhere between $K/2$ and K.

The subband signals at the output of the uniform DFT analysis bank satisfy a conjugate symmetric property: the ith subband signal is the complex conjugate of the $(K+2-i)$th for $1 < i \leq K/2$ (assuming K is even), while the first one and the $(K/2+1)$th one are real signals. Thus, there are a total of $K/2+1$ independent subband signals: two real and $K/2-1$ complex. The modified FAP algorithm in Section 2.2 is used in all these $K/2+1$. The length of the adaptive filters used in different subbands can be chosen differently, based on the consideration of the echo energy distribution among the subbands.

We note that in a subband acoustical echo canceler, different algorithms could be used in different subbands. In [2], a mixed structure is proposed where a QR-RLS algorithm is used in lower frequency bands due to the presence of higher echo level whereas the NLMS algorithm is used in higher frequency bands. One advantage of the FAP algorithm used here is that it is possible to achieve this effect easily by choosing different values of the parameter N in the different subbands. For example, we can set $N = 1$ (which corresponds to the NLMS algorithm) in subbands with low echo energy and use larger values of N in subbands with higher echo energy.

3.2. Computational requirements

The total computational requirement of the subband FAP algorithm consists of the following two parts.

The first part is associated to the analysis and synthesis filtering. We have two analysis filter banks and one synthesis bank. Each bank needs $(J + K log_2 K)$ real multiplications per M input samples, where J is the length (number of taps) of the prototype filters used in the analysis and synthesis filters.

The second part is due to the adaptive filtering in each subband. For simplicity, assume that the number of taps of all the subband filters is chosen equal to L/M, where L is the length of the full-band adaptive filter, which is supposed to match the duration of the echo path. Furthermore, the parameter N in the subband FAP algorithms is assumed to be equal for all subbands. Then, the bank of $K/2+1$ parallel subband adaptive filters require $2(K-1)(2L/M + 3N^2 + 12N)$ real multiplications every M input samples. Here, one complex multiplication is assumed to be equal to 4 real multiplications and K is even.

Thus, the total number of real multiplications per every M input samples for the subband FAP algorithm is:

$$E = 2(K-1)(2L/M + 3N^2 + 12N) + 3(J + K log_2 K) \quad (33)$$

The computational gain of the subband FAP over the full-band NLMS algorithm, which requires $2L$ real multiplications per input sample, will be $G = 2LM/E$. Examples of the gain G will be given in the next section.

4. EXPERIMENTAL RESULTS

The room impulse response used in the simulations is truncated to 1000 samples. Accordingly, the length of the full-band adaptive filter can then be chosen as $L = 1000$. Assume the number of subbands is $K = 8$ and the subband adaptive filters have equal length L/M. The computational gain of the subband FAP over the full-band NLMS, versus the subband downsampling rate M, are shown in Fig.1 for different values of N.

The composite source signal (CCS) is first used as the excitation signal of the adaptive echo canceler. Fig. 2 shows the residual echoes of the echo canceler when the full-band NLMS and the subband FAP algorithms are used. The input echo level is also shown in the figure, where the echo-to-noise ratio (ENR) is 40 dB. For the NLMS algorithm, the step size is set to $\mu = 1$. For the subband FAP algorithm, the following parameters are used: downsampling rate $M = 6$, $\mu = 1$ and $\delta = 2\sigma_x^2$, where σ_x^2 is an estimate of the average power of $x(n)$. As shown in Fig 2, when we increase N from $N = 1$ (NLMS) to $N = 3$, the convergence speed is significantly improved, while the corresponding complexity is only slightly increased.

The results of another set of simulations with a speech signal are illustrated in Fig. 3. In this experiment, a sudden change of room impulse response is made at sample $n = 14000$, as indicated by the vertical dashed line in Fig. 3. For subband FAP, N is set to 5, which results in a computational gain $G = 1.75$. When compared to the full-band NLMS algorithm, the subband FAP algorithm demonstrates much better performance in terms of both convergence rate and tracking capability.

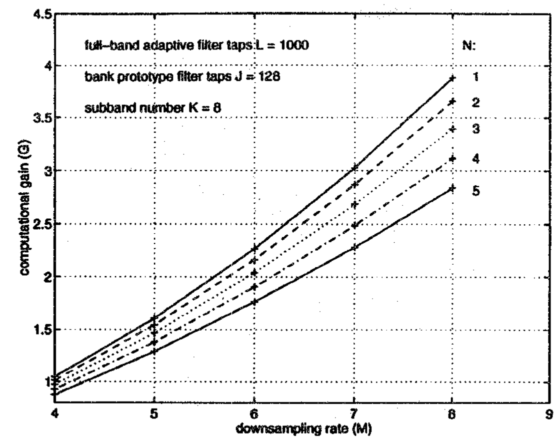

Figure 1: Computational gain of subband FAP over full-band NLMS.

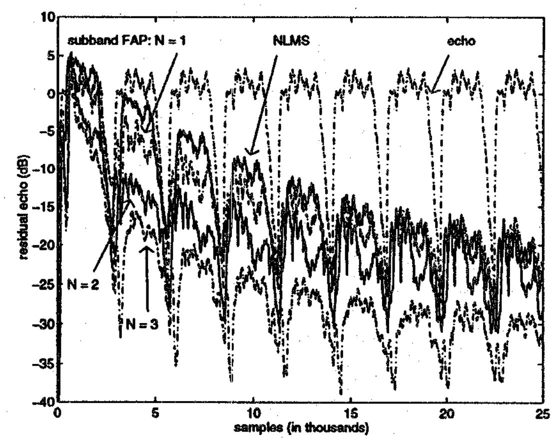

Figure 2: Residual echoes of subband FAP ($N = 1, 2, 3$) and full-band NLMS with CSS as excitation. Subband number $K = 8$, downsampling rate $M = 6$ and ENR = 40 dB.

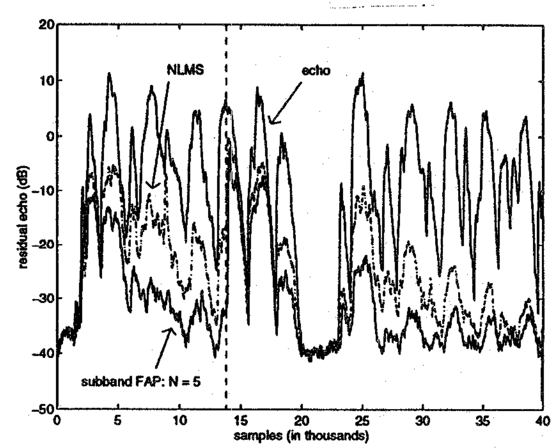

Figure 3: Residual echoes of subband FAP ($N = 5$) and full-band NLMS with speech as excitation. Subband number $K = 8$, downsampling rate $M = 6$ and ENR = 40 dB.

REFERENCES

[1] D. R. Morgan, "Slow asymptotic convergence of LMS echo canceler algorithm", *IEEE Trans. on Speech and Audio Processing*, vol. 3, pp. 126 - 136, Mar. 1995.

[2] F. Capman, J. Boudy and P. Lockwood, "Acoustic echo cancelation using a fast QR-RLS algorithm and multirate schemes, " in *Proc. IEEE Int. Conf. Acoust. Speech Signal Processing*, 1995, pp. 969 - 972.

[3] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Electronics and Communications in Japan*, vol. 67-A, No. 5, 1984.

[4] S. L. Gay and S. Tavathia, "The fast affine projection algorithm, " in *Proc. IEEE Int. Conf. Acoust. Speech Signal Processing*, 1995, pp. 3023 - 3026.

[5] M. Montazeri and P. Duhamel "A set of algorithm link NLMS and RLS algorithms, " in *Proc. EUSIPCO-94, Sixth European Signal Processing Conference*, 1995, pp. 744 - 747.

[6] R. E. Crochiere and L. R. Rabiner, *Multirate Digital Signal Processing*, Englewood Cliffs, NJ: Prentice-Hall 1983.