COMPLEXITY REDUCTION AND REGULARIZATION OF A FAST AFFINE PROJECTION ALGORITHM FOR OVERSAMPLED SUBBAND ADAPTIVE FILTERS

Edward Chau, Hamid Sheikhzadeh, Robert L. Brennan

Dspfactory Ltd., 611 Kumpf Drive, Unit 200, Waterloo Ontario, Canada N2V 1K8
E-mail addresses: {echau,hsheikh,rbrennan}@dspfactory.com

ABSTRACT

The Affine Projection Algorithm (APA) has been shown to improve the performance of Over-Sampled Subband Adaptive Filters (OS-SAFs) compared to classical Normalized Least Mean Square (NLMS) algorithms. Because of the complexity of APA, however, only low-order APAs are practical for real-time implementation. Thus, in this paper, we propose a reduced-complexity version of the Gauss-Seidel Fast APA (GSFAPA) for adapting the subband filters in OS-SAF systems. We propose modifying the GSFAPA with a complexity reduction method based on partial filter update, and also with a low-cost method for combined regularization and step size control. We show the advantage of the new algorithm – termed Low-Cost Gauss-Seidel Fast Affine Projection – compared to the APA in a subband echo canceller application.

1. INTRODUCTION

For many adaptive filtering applications, the Over-Sampled Subband Adaptive Filter (OS-SAF) [1, 2] can provide a significant improvement over other critically-sampled implementations because of the lower aliasing level in the subbands. With an oversampling factor (OS) of 2 or more, the group delay is also reduced, while aliasing in the subbands is maintained at a low level [2]. However, because of the larger eigenvalue spread in highly oversampled subband signals, the convergence of the popular NLMS algorithm in OS-SAF systems can be very slow. Thus, in [3], the Affine Projection Algorithm (APA) is proposed as a better adaptation technique for OS-SAF. The APA provides better convergence behavior than NLMS algorithms, while at the same time it avoids the high computation cost and instability associated with the Recursive Least Squares (RLS) algorithm [3]. Nevertheless, for real-time implementation on very low resource platforms, only an affine projection order of 2 or 3 is generally practical.

Recently, a fast version of APA based on Gauss-Seidel iterations (GSFAP) has been introduced [4]. The GSFAP algorithm is a stable and low-complexity version of the original fast APA (FAPA) [5]. It uses the Gauss-Seidel (GS) iterations instead of the sliding windowed Fast Recursive Least Squares algorithm to update the normalized residual echo vector, and thus has more stable convergence behavior when given a diagonally dominant autocorrelation matrix of the excitation signal.

In this paper, we propose to further reduce the complexity of the GSFAP algorithm for OS-SAF systems. We modify GSFAP with a method for partial filter update, and also with a low-cost method for combined regularization and step size control. The new algorithm, termed Low-Cost GSFAP (LC-GSFAP), is evaluated in a subband acoustic echo canceller (SAEC) application.

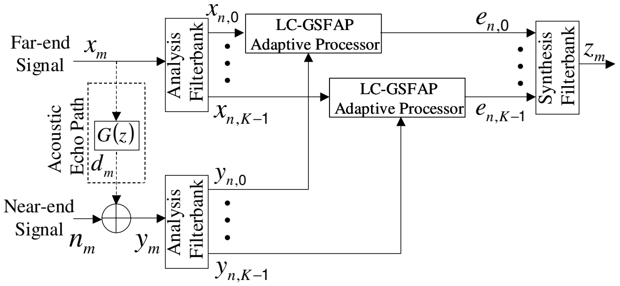

Fig. 1. Block Diagram of SAEC System

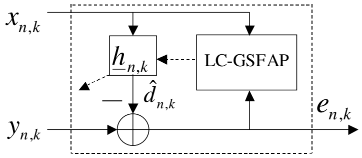

Fig. 2. LC-GSFAP Adaptive Processor

Figure 1 shows the block diagram of the OS-SAF system in an SAEC application. A highly efficient Weighted Overlap-Add (WOLA) filterbank [2] is used as the analysis/synthesis filterbank in the system. The WOLA is a uniform, generalized DFT filterbank with configurable, high over-sampling factors (OS $\geq$ 2). As shown in Figure 2, LC-GSFAP is used to adapt the filter weights of each subband. Note, however, in fast versions of the APA, the actual filter weights $\underline{h}_{n,k}$ are not directly calculated – only a "fast" version of the filter weights, $\underline{\hat{h}}_{n,k}$, is available. Figure 2 is thus only a conceptual illustration of the adaptive process.

The notations and definitions used throughout this paper is based heavily on [5] and [6]. For the rest of this paper, K denotes the number of subbands, L denotes the subband adaptive filter length, N denotes the affine projection order, and a subscript of m, n, and k (as in x_m and $x_{n,k}$), is used to denote, respectively, the time-domain sample index, the subband frame index, and the frequency band index. This paper is organized as follows. Section 2 briefly describes the GSFAP algorithm. Section 3 presents the LC-GSFAP algorithm, while Section 4 presents the simulation result. Finally, Section 5 presents the conclusions of this work.

2. THE GSFAP ALGORITHM

The time-domain GSFAP algorithm is summarized as follows:

Initialization: assume, for $m < 0$, $x_m = y_m = 0$, $\underline{\hat{h}}_m = \underline{0}^t$,

0-7803-8484-9/04/$20.00 ©2004 IEEE

$\underline{E}_m = \underline{0}^t$, $\underline{\hat{\alpha}}_m = \underline{0}^t$, $\mathbf{R}_m = \mathbf{X}_m^t \mathbf{X}_m = \mathbf{0} + \delta\mathbf{I}$, $\underline{\hat{r}}_{xx,m} = [\delta, \underline{0}]^t$, $0.7 < \mu < 1$, and $\underline{P}_m = \underline{b}/\delta$.

Then, at each sample $m \geq 0$:

1) $$\underline{\hat{r}}_{xx,m} = \underline{\hat{r}}_{xx,m-1} + x_m \underline{\hat{\alpha}}_m - x_{m-L} \underline{\hat{\alpha}}_{m-L} \tag{1}$$

2) $$\hat{e}_m = y_m - \underline{x}_m^t \underline{\hat{h}}_{m-1} \tag{2}$$

3) $$e_m = \hat{e}_m - \mu \underline{\tilde{r}}_{xx,m}^t \overline{\underline{E}}_{m-1} \tag{3}$$

4) update $\mathbf{R}_m$ using $\underline{\hat{r}}_{xx,m}$

5) solve $\mathbf{R}_m \underline{P}_m = \underline{b}$ for $\underline{P}_m$ using one GS iteration

6) $$\underline{\varepsilon}_m = e_m \underline{P}_m \tag{4}$$

7) $$\underline{E}_m = \begin{bmatrix} 0 \\ \overline{\underline{E}}_{m-1} \end{bmatrix} + \underline{\varepsilon}_m \tag{5}$$

8) $$\underline{\hat{h}}_m = \underline{\hat{h}}_{m-1} + \mu \underline{x}_{m-(N-1)} E_{N-1,m} \tag{6}$$

In the above, $\underline{\hat{h}}_m$ is the $L \times 1$ fast adaptive filter coefficients, $\underline{E}_m$ is an $N \times 1$ vector consisting of a sum of the fast normalized residual echo $\underline{\varepsilon}_m$ [5], $E_{N-1,m}$ is the last element of $\underline{E}_m$, $\overline{\underline{E}}_m$ is a vector consisting of the uppermost $N-1$ elements of $\underline{E}_m$, $\underline{x}_m$ is the $L \times 1$ excitation signal vector, y_m is the reference signal, $\underline{\hat{\alpha}}_m = [x_m, \ldots, x_{m-N+1}]^t$, $\underline{b} = [1, \underline{0}]^t$, $\mathbf{X}_m$ is the $L \times N$ excitation signal matrix, $\mathbf{I}$ is the identity matrix, and δ is the regularization factor. Note, $\underline{\tilde{r}}_{xx,m}$ in Equation 3 has the same definition as in [5], and is simply the $N-1$ lower elements of $\underline{\hat{r}}_{xx,m}$. Also, in step 4, $\mathbf{R}_m$ is updated simply by replacing the first row and column with the elements of $\underline{\hat{r}}_{xx,m}$, and the bottom $(N-1) \times (N-1)$ submatrix is replaced with the top $(N-1) \times (N-1)$ submatrix of $\mathbf{R}_{m-1}$.

In OS-SAF systems, because the WOLA filterbank provides near-orthogonal subbands, the GSFAP algorithm can be applied independently in each subband. Hence, for subband implementations, we can simply exchange the subscript m with the pair (n, k), and generalize all operations for complex-valued signals (i.e. transposes become Hermitian transposes, multiplications become complex-valued multiplications, and the filter is now a complex filter, etc.). The complexity of GSFAP in this implementation is $N^2 + 3N + 2L$ MACs plus 1 multiplication and 1 division (per subband). Due to the regular structure of $\mathbf{R}_m$ (or $\mathbf{R}_{n,k}$ in subband), step 4 of GSFAP can be implemented efficiently – for example, by simply reorganizing memory buffers.

3. THE LC-GSFAP ALGORITHM

3.1. Partial Filter Update

The Partial Filter Update (PFU) method in LC-GSFAP aims to reduce the computation requirement of GSFAP without sacrificing too much of the system performance. It is similar in concept to the Sequential Least Mean Square (S-LMS) algorithm in [7], and is implemented as follows. Let D be a positive integer, $D \geq 1$ and $L \bmod D = 0$. Then, the updating of the subband "fast" filter coefficients $\underline{\hat{h}}_{n,k}$ (Equation 6) is modified to

$$\underline{\hat{h}}_{n,k} = \underline{\hat{h}}_{n-1,k} + \mu \underline{\tilde{x}}_{n-(N-1),k} E_{N-1,n,k} \tag{7}$$

where

$$\underline{\tilde{x}}_{n,k} = [\tilde{x}_{n,k}, \tilde{x}_{n-1,k}, \ldots, \tilde{x}_{n-L+1,k}]^t \tag{8}$$

$$\tilde{x}_{n,k} = \begin{cases} x_{n,k} & \text{if } n \bmod D = 0 \\ 0 & \text{otherwise} \end{cases} \tag{9}$$

In effect, when $D > 1$, only every D-th element of $\underline{\hat{h}}_{n,k}$ is updated at a time. When $D = 1$, the algorithm reverts to the original GSFAP method. Also, $\underline{\hat{\alpha}}_{n,k}$ in Equation 1 is replaced with $\underline{\breve{\alpha}}_{n,k} = [\tilde{x}_{n,k}, \ldots, \tilde{x}_{n-N+1,k}]^t$, which means only every D-th element of $\underline{\hat{r}}_{xx,n,k}$ is updated at a time. As a result, $\mathbf{R}_{n,k}$ can also be updated at every D-th frame (corresponding to step 4 in Section 2), which is when $\underline{\hat{r}}_{xx,n,k}$ is fully updated. The complexity of GSFAP with PFU implementation is $N^2 + N + L + \frac{2N+L}{D}$ MACs plus 1 multiplication and 1 division (per subband). Note that the update rate of $\mathbf{R}_{n,k}$ is also reduced by a factor of D.

In [8], it is shown that the PFU method in the S-LMS algorithm can be seen as a polyphase implementation of LMS where only one polyphase component is updated at a time. It can be seen clearly that a similar polyphase analysis will also apply here for LC-GSFAP. However, one significant difference here is that the polyphase decomposition is applied to $\underline{\hat{h}}_{n,k}$, which is a "transformed" version of the actual filter $\underline{h}_{n,k}$. The relationship between the two filters can be seen easily from Equation 2 and Equation 3. Since now $\underline{\tilde{r}}_{xx,n,k}$ is updated at a decimated rate, it is therefore important to note that the PFU method here is not necessarily modeled by a Perfect Reconstruction (PR) filterbank as noted in [8] for S-LMS. Nevertheless, in the over-sampled subband domain, aliasing and misadjustment due to decimation is not expected to be significant so long as D does not exceed the filterbank over-sampling factor (i.e. $D \leq OS$). The effect of D on the system performance can be seen in the result presented in Section 4.

3.2. Low-Cost Regularization

In [4], the regularization factor δ is a constant value that is used only when initializing $\mathbf{R}_m$ (or $\mathbf{R}_{n,k}$ in subband). In practice, however, δ should be time-varying and is often determined *on-line* – for example, in [9]. It can be seen in [9] that the regularization factor δ can also be used as a step size control for filter adaptation. In an echo canceller application, it is well-known that the filter adaptation should be stopped or slowed down when there is a high level of near-end signal activity. However, the method described in [9] for finding the value of δ requires an estimation of the system mismatch, which can be computationally expensive. It is also unclear whether the adaptation will remain stable when there is little or no near-end signal activity and the system is nearly converged, because the method does not explicitly ensure that $\mathbf{R}_m$ is not ill-conditioned.

Thus, we propose a low-cost alternative method that effectively combines regularization with step size control. We start with the pseudo-optimal regularization factor from [9],

$$\delta_{opt,n,k} \approx \frac{L\sigma_{x,n,k}^2 (\sigma_{e,n,k}^2 - \sigma_{\varepsilon,n,k}^2)}{\sigma_{\varepsilon,n,k}^2} \tag{10}$$

where $\sigma_{x,n,k}^2$ is the power of the subband excitation signal, $\sigma_{e,n,k}^2$ is the power of the subband error signal, and $\sigma_{\varepsilon,n,k}^2$ is the power of the subband undistorted error signal. Applying the approximations as described in [9], and realizing that $\sigma_{e,n,k}^2 - \sigma_{\varepsilon,n,k}^2$ can be exchanged with $\sigma_{y,n,k}^2 - \sigma_{d,n,k}^2$, we obtain,

$$\delta_{opt,n,k} \approx \frac{L(\sigma_{y,n,k}^2 - \sigma_{d,n,k}^2)}{\mathrm{E}\{\|\Delta_n\|^2\}} \tag{11}$$

where $\sigma_{y,n,k}^2$ is the power of the subband reference signal (near-end signals plus the echo signal), $\sigma_{d,n,k}^2$ is the power of the subband echo signal, and Δ_n is the system mismatch. Since the subband echo signal $d_{n,k}$ is not directly available, and the estimation of Δ_n can be computationally expensive, we simplify the above to

$$\delta_{opt,n,k} \approx L\sigma^2_{y,n,k} \qquad (12)$$

As a step size control, Equation 12 will tend to slightly over-suppress the adaptation because the power of the echo signal is included, but will nevertheless achieve the desired effect of slowing down the adaptation when there is a high level of near-end signal activity. We have removed the system mismatch estimation entirely because when there is a high level of near-end signal activity, we need to slow down the adaptation regardless of whether the system is near convergence. When there is little or no near-end signal activity, we will rely on an explicit matrix regularization approach (described below) for step size control.

To incorporate the new regularization factor into LC-GSFAP, we compute, before the GS iteration in step 5 of the GSFAP algorithm (Section 2),

$$\tilde{\mathbf{R}}_{n,k} = \mathbf{R}_{n,k} + \underline{\delta}_{n,k}\mathbf{I} \qquad (13)$$

Then, $\tilde{\mathbf{R}}_{n,k}$ is used in the GS iteration instead of $\mathbf{R}_{n,k}$. The regularization factor $\underline{\delta}_{n,k}$ is now a $1 \times N$ vector. Each element of $\underline{\delta}_{n,k}$, i.e. $\delta^{(i)}_{n,k}$, for $0 < i \leq N-1$, is determined by

$$\delta^{(i)}_{n,k} = \overline{max\{L|y_{n,k}|^2, D\delta^{(i)}_{R,n,k}\}} \qquad (14)$$

where $y_{n,k}$ is the subband reference signal, and

$$\delta^{(i)}_{R,n,k} = \left(\sum_{j=0,j\neq i}^{N-1} |x^*_{n-j,k}x_{n-i,k}|\right) - |x^*_{n-i,k}x_{n-i,k}| \qquad (15)$$

The superscript * is used to denote complex conjugate. The term $D\delta^{(i)}_{R,n,k}$ is used to ensure that the matrix $\tilde{\mathbf{R}}_{n,k}$ is always diagonally dominant, which is especially important for convergence stability when the near-end signal level is low. The factor D is to take into account of the frequency scaling effect due to the decimation of the excitation signal in the PFU method (note the regularization factor is still updated at every frame even when $D > 1$). The overline in Equation 14 denotes averaging by a dual-time-constant attack-release filter (i.e. $\delta^{(i)}_{n,k} = \alpha\delta^{(i)}_{n,k} + (1-\alpha)\delta^{(i)}_{n-1,k}$ if $\delta^{(i)}_{n,k} \geq \delta^{(i)}_{n-1,k}$, otherwise $\delta^{(i)}_{n,k} = \beta\delta^{(i)}_{n-1,k}$, and $0 < \alpha, \beta < 1$).

For practical implementation, it is also possible to further reduce the computation cost by replacing the vector $\underline{\delta}_{R,n,k}$ with a scalar $\delta_{R,n,k}$ (correspondingly, $\underline{\delta}_{n,k}$ with $\delta_{n,k}$), where $\delta_{R,n,k} = \max_i\{\delta^{(i)}_{R,n,k}\}$ or $\delta_{R,n,k} = \delta^{(0)}_{R,n,k}$. Clearly, this approximation will make the algorithm more susceptible to highly non-stationary signals, but a compromise can likely be reached depending on the sizes of N and L, and the nature of the intended environment.

3.3. LC-GSFAP

The proposed algorithm is summarized in the following. It should be understood that the subband-domain description can be easily applied to the time domain by exchanging (n,k) with m, and treating all imaginary parts as zero. The superscript ^{H} is used to denote Hermitian transpose.

Constraint: D is an integer ≥ 1, $L \bmod D = 0$.

Initialization: assume, for $n < 0$, $x_{n,k} = y_{n,k} = 0$, $\underline{\hat{h}}_{n,k} = \underline{0}^H$, $\underline{E}_{n,k} = \underline{0}^H$, $\underline{\check{\alpha}}_{n,k} = \underline{0}^H$, $\mathbf{R}_{n,k} = \mathbf{X}^H_{n,k}\mathbf{X}_{n,k} = \mathbf{0} + \mathbf{I}$, $\underline{\hat{r}}_{xx,n,k} = [1, \underline{0}]^H$, $\mu = 1$, and $\underline{P}_{n,k} = \underline{b}$.

Then, at each sample $n \geq 0$, $0 \leq k \leq K-1$:

1) $\underline{\hat{r}}_{xx,n,k} = \underline{\hat{r}}_{xx,n-1,k} + x_{n,k}\underline{\check{\alpha}}_{n,k} - x_{n-L,k}\underline{\check{\alpha}}_{n-L,k}$ (16)

2) $\hat{e}_{n,k} = y_{n,k} - \underline{x}^H_{n,k}\underline{\hat{h}}_{n-1,k}$ (17)

3) $e_{n,k} = \hat{e}_{n,k} - \mu\underline{\tilde{r}}^H_{xx,n,k}\underline{\overline{E}}_{n-1,k}$ (18)

4) if $n \bmod D = 0$, update $\mathbf{R}_{n,k}$ using $\underline{\hat{r}}_{xx,n,k}$

5) $\tilde{\mathbf{R}}_{n,k} = \mathbf{R}_{n,k} + \underline{\delta}_{n,k}\mathbf{I}$

6) solve $\tilde{\mathbf{R}}_{n,k}\underline{P}_{n,k} = \underline{b}$ for $\underline{P}_{n,k}$ using one GS iteration

7) $\underline{\varepsilon}_{n,k} = e_{n,k}\underline{P}_{n,k}$ (19)

8) $\underline{E}_{n,k} = \begin{bmatrix} 0 \\ \underline{\overline{E}}_{n-1,k} \end{bmatrix} + \underline{\varepsilon}_{n,k}$ (20)

9) $\underline{\hat{h}}_{n,k} = \underline{\hat{h}}_{n-1,k} + \mu\underline{\tilde{x}}_{n-(N-1),k}E_{N-1,n,k}$ (21)

4. SIMULATION

We compare the performance of LC-GSFAP with APA by implementing the OS-SAF systems in an echo canceller application. The number of subbands in the systems is $K = 8$, with $OS = 4$. The acoustic echo path $G(z)$ (Figure 1) has an echo delay of 55 ms and a return loss of -15 dB. Two different male speech segments about 90 seconds long are used as the far-end and near-end signals, with double-talk occurring near the middle of the segment. White noise is also added to both the far-end and near-end signals at an SNR of 20 dB. The sampling frequency is 8 kHz.

The adaptive filter length is 112 taps per band. The affine projection order is 4 in both LC-GSFAP and APA versions. In the APA implementation, a constant regularization factor is used, which has been determined off-line to yield the best stable performance. The performance of the echo canceller is determined by the magnitude of the echo estimation error, i.e. $\left||d_{n,k}| - |\hat{d}_{n,k}|\right|$, where $d_{n,k}$ is the subband echo signal, and $\hat{d}_{n,k}$ is the subband estimated echo signal. We have chosen this measure instead of the commonly used ERLE measure because we intend to better show the variation of the error over time due to near- and far-end signal activities.

Figure 3 shows the time domain echo signal, the time domain near-end local signal (i.e. no echo) and the echo estimation error for the second subband (center frequency at 750 Hz). Different "PFU factors" (i.e. the value D described in Section 3) are also used in the LC-GSFAP implementation, and their performances are shown in Figure 3 and Figure 4. Note that, in Figure 3, the signal amplitudes are represented using denormalized 16-bit integers.

As seen in Figure 3, for $D \leq OS$, the performance of LC-GSFAP is significantly better than APA for both the quiet and voiced parts of the echo, achieving as much as 10 dB lower estimation error. The value of D (for $D < OS$) also has only minimal impact on the estimation error. However, for $D > OS$, the subband excitation signal becomes under-sampled. As a result, as seen in Figure 4, the estimation error starts to increase as the subband excitation signal becomes critically- or under-sampled, although even at $D = 8$, the error is not significantly worse than that of APA. Furthermore, the performances of APA and LC-GSFAP (for $D \leq OS$) are very similar in the double-talk situation (the period between 40 and 45 seconds in the simulation), even though the regularization factor in APA is optimized off-line, while in LC-GSFAP it is determined on-line. This shows that the regularization method in LC-GSFAP is robust enough to ensure adaptation stability during double-talk situation.

The regularization factor $\delta_{n,2}^{(0)}$ is shown in Figure 5 for the case of $D = 2$. For the simulation we have used instantaneous attack and a slow release time-constant (1 s) in the averaging filter in Equation 14. The tracking of both near- and far-end signal activities can be seen clearly in the figure.

5. CONCLUSION

In this paper, we described a low-complexity alternative to APA for OS-SAF systems – LC-GSFAP. The computation cost is reduced by using a PFU method in GSFAP, with a low-complexity method for combined regularization and step size control. The regularization and step size control is based partially on the well-known conditions for adaptive filter convergence and stability in an echo canceller application. Our simulation result has shown that LC-GSFAP has lower echo estimation error compared to APA using a constant, pre-determined regularization factor. The result has also shown that the value of D has only minimal impact on the convergence behavior of LC-GSFAP, so long as $D \leq OS$. Nevertheless, the estimation error of LC-GSFAP in the simulation result is still not significantly worse than that of APA even at $D = 2 \times OS$. Finally, for future work, we plan to carry out a more in-depth analysis of the convergence behavior of LC-GSFAP using the polyphase decomposition approach.

6. REFERENCES

[1] S. Weiss, *On Adaptive Filtering in Oversampled Subbands*, Ph.D. thesis, University of Strathclyde, 1998.

[2] R. Brennan and T. Schneider, "A flexible filterbank structure for extensive signal manipulation in digital hearing aids," *Proc. IEEE Int. Symp. Circuits and Systems*, pp. 569–572, 1998.

[3] H. R. Abutabeli *et. al.*, "Affine projection algorithm for oversampled subband adaptive filters," *Proc. IEEE Int. Conf. Acous., Speech, and Sig. Proc.*, pp. VI 209–212, 2003.

[4] F. Albu, J. Kadlec, N. Coleman, and A. Fagan, "The gauss-seidel fast affine projection algorithm," *Proc. IEEE Workshop on Signal Processing Systems*, pp. 109–114, 2002.

[5] S. L. Gay and S. Tavathia, "The fast affine projection algorithm," *Proc. IEEE Int. Conf. Acous., Speech, and Sig. Proc.*, pp. 3023–3026, 1995.

[6] S. L. Gay, *Fast Projection Algorithms with Application to Voice Excited Echo Cancellers*, Ph.D. thesis, Rutgers University, 1994.

[7] S. C. Douglas, "Adaptive filters employing partial updates," *IEEE Trans. Circuits and Systems - II*, vol. 44, pp. 209–216, 1997.

[8] H. Sheikhzadeh *et. al.*, "Sequential LMS for low-resource subband adaptive filtering: Oversampled implementation and polyphase analysis," *submitted to the 12th European Signal Processing Conference*, EUSIPCO 2004.

[9] V. Myllylä and G. Schmidt, "Pseudo-optimal regularization for affine projection algorithms," *Proc. IEEE Int. Conf. Acous., Speech, and Sig. Proc.*, pp. II 1917–1920, 2002.

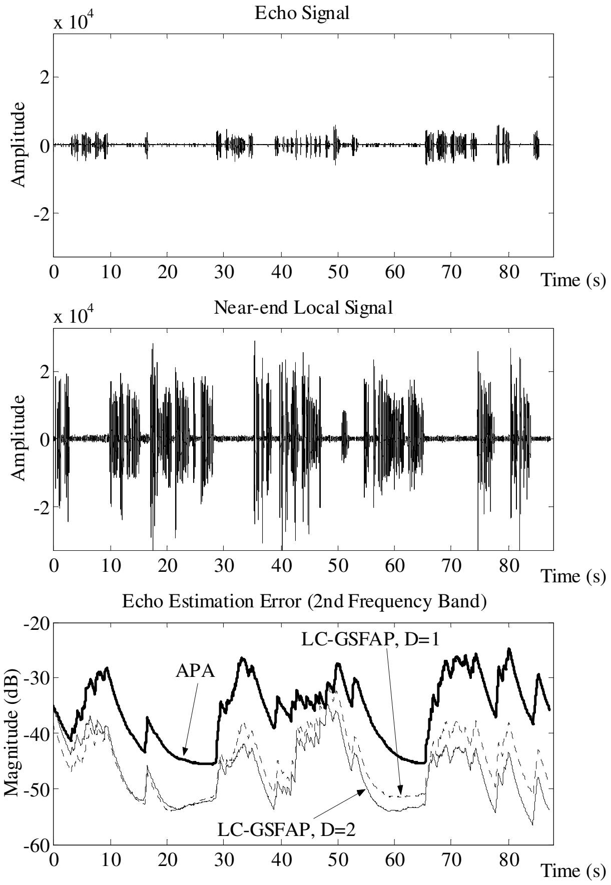

Fig. 3. Echo, Near-End Local Signals and Magnitude of Echo Estimation Errors (Low PFU factors)

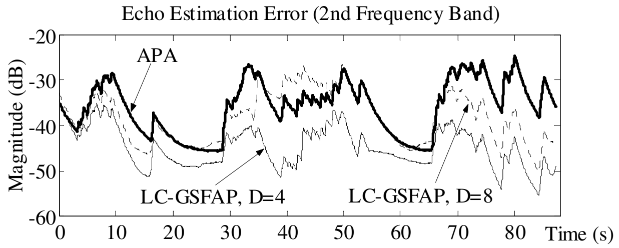

Fig. 4. Magnitude of Echo Estimation Errors (High PFU factors)

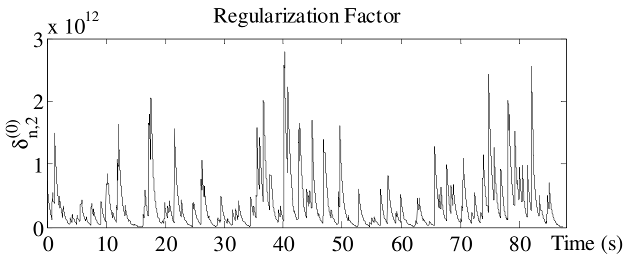

Fig. 5. Regularization Factor $\delta_{n,2}^{(0)}$ For $D = 2$