

PARTIAL UPDATE SUBBAND IMPLEMENTATION OF COMPLEX PSEUDO-AFFINE PROJECTION ALGORITHM ON OVERSAMPLED FILTERBANKS

Hamid Sheikhzadeh, Kevin R. L. Whyte, and Robert L. Brennan

AMI Semiconductor Canada Company, 611 Kumpf Drive, unit 200, Waterloo, Ontario, Canada N2V 1K8
Emails: {hsheikh, Robert.brennan}@amis.com

ABSTRACT

Fast adaptive algorithms targeted for low-resource implementation on oversampled filterbanks are discussed. First, simplifications to the Pseudo Affine Projection (PAP) are proposed that decrease the computation cost without degrading the performance. Next, the sequential update PAP algorithm is proposed to further decrease the computational complexity. It is shown that, unlike the partial update fast affine projection, increasing the decimation rate of the partial update algorithm does not lead to more aliasing in the sequential update PAP. Moreover, it is experimentally observed that with proper regularization or step-size scaling, the convergence rate of the sequential PAP is identical for various decimation factors of the partial update. The proposed methods are implemented and evaluated on a low-resource oversampled filterbank platform.

1. INTRODUCTION

This research explores efficient and advanced methods of low-resource adaptive processing for acoustic echo cancellation. Due to the many known advantages of subband adaptive filters over time-domain filters, the system is targeted for low-resource oversampled subband adaptive filters (OS-SAFs). We have already investigated the use of Affine Projection Algorithm (APA) on OS-SAFs [1] as a superior adaptation technique. Considering the colored nature of oversampled subband signals, APA provides faster convergence than the NLMS algorithm while avoiding the high computation cost and instability associated with the recursive least squares. We recently proposed to further reduce the complexity of fast APA (FAPA) [2,3] by combining it with Partial Filter Update (PFU-FAPA) [4] employed on low-resource OS-SAF platforms. From a theoretical view, polyphase models based on delay-chain perfect reconstruction (PR) filterbanks were presented in [5] for various types of partial update LMS algorithms. The results were then extended in [6] to include PFU-FAPA. It is shown [6] that the polyphase filterbank model of the "transformed" version of the adaptive filter in PFU-FAPA (the "fast" adaptive coefficient vector in [2]) constitutes a PR filterbank. However, the polyphase filterbank model of the autocorrelation estimation part in PFU-FAPA does *not* represent a PR system. As a result, increasing the decimation rate of the PFU will result in performance degradation of the PFU-FAPA as reported in [4].

Another difficulty with the family of FAPA (including PFU-FAPA) is the fact that only a transformed version of the adaptive filter is adapted. As a result, for direct real-time system identification with FAPA, one has to obtain the adaptive filter

through extra computations. This same difficulty exists for zero-delay adaptive filters, where the time-domain filter is obtained through a transformation of the subband adaptive filters. The recently proposed Pseudo Affine Projection (PAP) algorithm [7,8,9], removes the above limitations by offering a computational complexity comparable to FAPA, while providing access to the adaptive filter without transformation.

In this paper, we employ the Gauss-Seidel PAP (GS-PAP) algorithm proposed in [8,9] since it can be better tailored to low-resource implementation compared to the PAP in [7]. We first apply a simplification to GS-PAP followed by a partial filter update scheme. The resulting algorithm is termed: Sequential GS-PAP (SGS-PAP). It is shown that unlike partial update FAPA, SGS-PAP does not suffer from the aliasing problem as the partial update decimation rate increases.

The proposed SGS-PAP is implemented and evaluated on a low-resource OS-SAF platform with complex subband signals. Section 2 summarizes the complex GS-PAP algorithm and Section 3 describes a proposed modification to the algorithm. The proposed SGS-PAP is introduced in Section 4, and system evaluations and conclusions are discussed in Sections 5 and 6, respectively.

2. COMPLEX GAUSS-SEIDEL PSEUDO-APA

We first establish the notation:

N : affine order, L : adaptive filter length

x_n : reference signal, s_n : primary signal

$e_n = s_n - \underline{h}_{n-1}^H \underline{x}_n$: error signal,

$\delta, \delta_1, \delta_2$: regularization parameters

\underline{x}_n : $[x_n, x_{n-1}, \dots, x_{n-L+1}]^T$ $L \times 1$

\underline{X}_n : $[\underline{x}_n, \underline{x}_{n-1}, \dots, \underline{x}_{n-N+1}]$ $L \times N$

$\underline{\alpha}_n$: $[x_n, x_{n-1}, \dots, x_{n-N+1}]^T$ $N \times 1$

\mathbf{R}_n : $\mathbf{X}_n^H \mathbf{X}_n$ autocorrelation matrix $N \times N$

\underline{s}_n : $[s_n, s_{n-1}, \dots, s_{n-L+1}]^T$ $N \times 1$

\underline{p}_n : first column of $[\mathbf{X}_n^H \mathbf{X}_n + \delta \mathbf{I}]^{-1}$ $N \times 1$

\underline{h}_n : adaptive filter taps $L \times 1$

\underline{U}_{n-1} : top $N-1$ elements of \underline{U}_{n-1} $(N-1) \times 1$

The complex affine projection algorithm (APA) [2] of order N (without regularization) adapts the filter according to

$$\underline{h}_n = \underline{h}_{n-1} + \mu \mathbf{X}_n [\mathbf{X}_n^H \mathbf{X}_n]^{-1} (\underline{s}_n - \mathbf{X}_n^T \underline{h}_{n-1}^*) \quad (1)$$

The Pseudo Affine Projection (PAP) as originally proposed in [7] solves the matrix inversion in (1) with a linear prediction while the Gauss-Seidel PAP (GS-PAP) [8,9] employs the Gauss-

Seidel method to do the same. The GS-PAP algorithm is summarized here.

1. Initialization: $\mathbf{R}_{-1} = \delta_1 \mathbf{I}$, $\mathbf{P}_{-1} = \begin{bmatrix} 1/\delta_1 & \mathbf{0}^T \end{bmatrix}^T$, $\mathbf{h}_{-1} = \mathbf{0}$, $\mathbf{U}_{-1} = \mathbf{0}$
2. Autocorrelation matrix update:

$$\mathbf{R}_n = \mathbf{R}_{n-1} + \underline{\alpha}_n^* \underline{\alpha}_n^T - \underline{\alpha}_{n-1}^* \underline{\alpha}_{n-1}^T$$

3. Update matrix inversion estimate through one GS

$$\text{iteration: } \mathbf{R}_n \mathbf{P}_n = \begin{bmatrix} 1 & \mathbf{0}^T \end{bmatrix}^T$$

4. Calculate or Update \mathbf{U}_n :

$$\mathbf{U}_n = \frac{1}{\mathbf{P}_{n,0}} \mathbf{X}_n \mathbf{P}_n \quad \text{or} \quad \mathbf{U}_n \approx \begin{bmatrix} u_n \\ \mathbf{U}_{n-1} \end{bmatrix}, u_n = \frac{1}{\mathbf{P}_{n,0}} \underline{\alpha}_n^T \mathbf{P}_n$$

5. Calculate output error: $e_n = s_n - \mathbf{h}_{n-1}^H \mathbf{x}_n$
6. Update adaptive filter: $\mathbf{h}_n = \mathbf{h}_{n-1} + \mu \frac{\mathbf{U}_n}{\mathbf{x}_n^H \mathbf{U}_n + \delta_2} e_n^*$

3. MODIFICATION TO GS-PAP

Rather than using \mathbf{U}_n , one can define $\mathbf{V}_n = \mathbf{X}_n \mathbf{P}_n$ to arrive at the following update equation

$$\mathbf{h}_n = \mathbf{h}_{n-1} + \mu \mathbf{V}_n e_n^* \quad (2)$$

In fact when the \mathbf{P}_n estimate is exact and there is no regularization, step 6 of GS-PAP above and Eq. (2) are equivalent. To prove this, notice how the terms in step 6 may be simplified:

$$\frac{\mathbf{U}_n}{\mathbf{x}_n^H \mathbf{U}_n} = \frac{\mathbf{X}_n \mathbf{P}_n}{\mathbf{x}_n^H \mathbf{X}_n \mathbf{P}_n} = \mathbf{X}_n \mathbf{P}_n$$

The denominator is identically one since $\mathbf{x}_n^H \mathbf{X}_n$ is the first row of the matrix $\mathbf{R}_n = \mathbf{X}_n^H \mathbf{X}_n$ while \mathbf{P}_n is the first column of its inverse. Thus Eq. (2) can replace Step 6 of GS-PAP.

Furthermore, to gain the same computation advantage offered by GS-PAP, \mathbf{V}_n can be approximated in a manner similar to \mathbf{U}_n . Thus Step 4 of GS-PAP is replaced as follows:

4. Calculate or Update $\mathbf{V}_n = \mathbf{X}_n \mathbf{P}_n$ or

$$\mathbf{v}_n = \mathbf{P}_n^T \underline{\alpha}_n, \quad \mathbf{V}_n \approx \begin{bmatrix} \mathbf{v}_n \\ \mathbf{V}_{n-1} \end{bmatrix} \quad (3)$$

where \mathbf{V}_{n-1} is a vector containing the top $N-1$ elements of \mathbf{V}_{n-1} . When the reference signal is stationary, the approximation offered by Eq. (3) is appropriate. \mathbf{P}_n is slowly changing since it estimates the first column of the inverse of the autocorrelation matrix. Similar to the one offered in Step 4 of GS-PAP, the approximation degrades as the reference signal becomes less stationary.

The advantage of the modified GS-PAP (using the approximation in Step 4 for both) is a complexity reduction by 2 Multiply and Accumulations (MAC's), and 2 divisions per sample. While this saving may seem insignificant, it can add up quickly in an oversampled subband adaptive filter design.

We have experimentally verified that the proposed modifications do not alter the convergence and steady-state properties of the GS-PAP. As a result, in all evaluations, the modified version of the GS-PAP (Eq.'s (2) and 3) is employed.

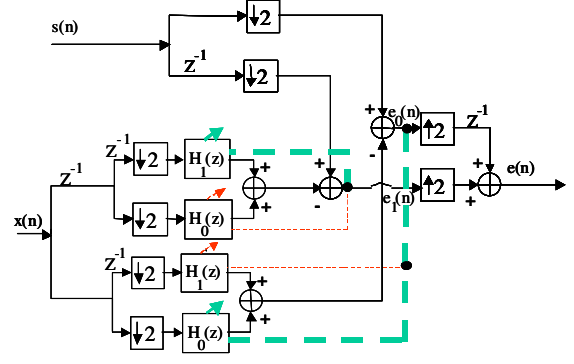


Figure 1: Polyphase adaptive filter representation using a delay chain. For sequential update, ignore the thin dashed lines.

4. SEQUENTIAL GS-PAP

A polyphase analysis of the sequential LMS (S-LMS) and partial update FAPA has already been provided in [5] and [6], respectively. It was shown in [5] that the block diagram in Fig. 1 could express the sequential update LMS using a sequential decimation factor of 2.

Note that in Fig. 1 $H_0(z)$ and $H_1(z)$ are the polyphase components of the adaptive filter. In order to obtain the sequential update scheme, only one of the polyphase components is updated at a time using a single decimated version of the reference signal. This update scheme corresponds to ignoring the thin dashed lines in Fig. 1.

Through careful consideration of Fig. 1, the Sequential Gauss-Seidel Pseudo Affine Projection (SGS-PAP) algorithm will be derived. Let D be the sequential decimation factor, assumed to evenly divide L . Also define

$$\underline{\zeta}_n = [x_n, x_{n-D}, \dots, x_{n-L+D}]^T \quad (L/D \times 1)$$

$$\underline{\xi}_n = [x_n, x_{n-D}, \dots, x_{n-DN+D}]^T \quad (N \times 1)$$

$$\mathbf{Z}_n = \begin{bmatrix} \underline{\zeta}_n & \underline{\zeta}_{n-D} & \dots & \underline{\zeta}_{n-DN+D} \end{bmatrix} \quad (L/D \times N)$$

It can be seen in Fig. 1 that only a decimated version of the reference signal is used to adapt the adaptive filter, thus it makes sense to use the autocorrelation matrix of the decimated signal instead of the autocorrelation matrix of the complete reference signal. Thus \mathbf{R}_n becomes $\mathbf{R}_n = [\mathbf{Z}_n^H \mathbf{Z}_n + \delta \mathbf{I}]$ instead of its typical form $\mathbf{R}_n = [\mathbf{X}_n^H \mathbf{X}_n + \delta \mathbf{I}]$. Note that in order to ensure that the matrix $\mathbf{Z}_n^H \mathbf{Z}_n$ is non-singular, the affine order and sequential decimation rate must be set satisfying the inequality $N \leq L/D$. Assuming this and expanding on the update pattern suggested by Fig. 1, one can write out the SGS-PAP algorithm step-by-step as:

1. Initialization:

$$\mathbf{R}_{-D} = \delta_1 \mathbf{I}, \mathbf{P}_{-D} = \begin{bmatrix} 1/\delta_1 & \mathbf{0}^T \end{bmatrix}^T, \mathbf{h}_{-1} = \mathbf{0}, \mathbf{U}_{-D} = \mathbf{0}, \mathbf{V}_{-D} = \mathbf{0}$$

2. If $n \bmod D \neq 0$ then skip to 6.

3. Update autocorrelation matrix:

$$\mathbf{R}_n = \mathbf{R}_{n-D} + \underline{\xi}_n^* \underline{\xi}_n^T - \underline{\xi}_{n-1}^* \underline{\xi}_{n-1}^T$$

4. Update inversion estimate

$$\text{Put } \mathbf{R}_n \mathbf{P}_n = \begin{bmatrix} 1 & \mathbf{0}^T \end{bmatrix}^T \text{ through one GS iteration.}$$

5. Calculate or Update \mathbf{U}_n or \mathbf{V}_n :

$$\underline{U}_n = \frac{1}{P_{n,0}} Z_n P_n, \text{ or } u_n = \frac{1}{P_{n,0}} \xi_n^T P_n, \underline{U}_n \approx \begin{bmatrix} u_n \\ \underline{U}_{n-D} \end{bmatrix} \text{ or}$$

$$\underline{V}_n = Z_n P_n \quad \text{or} \quad v_n = \xi_n^T P_n, \quad \underline{V}_n \approx \begin{bmatrix} u_n \\ \underline{V}_{n-D} \end{bmatrix}$$

6. Calculate output error: $e_n = s_n - \underline{h}_{n-1}^H \underline{x}_n$
7. Update one polyphase component ($\tilde{h}_{n,k}$) of the adaptive filter:

$$\tilde{h}_{n,k} = \tilde{h}_{n-1,k} + \mu \frac{\underline{U}_\ell}{\xi_\ell^H \underline{U}_\ell + \delta_2} e_n^* \quad \text{or} \quad \tilde{h}_{n,k} = \tilde{h}_{n-1,k} + \mu \underline{V}_\ell e_n^*$$

$$\ell = \left\lfloor \frac{n}{D} \right\rfloor D, \quad k \in \{0, 1, \dots, D-1\}, \quad k = n - \ell$$

$$\tilde{h}_{n,k} = [h_{n,k}, h_{n,k+D}, \dots, h_{n,k+L-D}]^T$$

Note that when switching from the GS-PAP algorithm to the SGS-PAP algorithm, the \underline{U}_n or \underline{V}_n vector reduces from a size of $L \times 1$ to $L/D \times 1$. Using the modifications proposed in Section 3 and the approximation in the Step 5, the complexity of an optimal implementation of the SGS-PSP algorithm is $L + 2 + (L + N^2 + 3N)/D$ MAC's and $1/D$ divisions.

4.1 Online Regularization

Online regularization based on the scheme presented in [10] was employed for the SGS-PAP with slight modifications to consider the effect of decimation in the sequential update. The scheme detailed in (4) determines the regularization parameter, δ_n , smoothed with a first-order attack-release IIR filter. In (4), R is the filterbank decimation rate (see Section 5), D is the sequential update decimation rate, and F_s is the sampling frequency. Estimates of the reference and primary signal powers ($\sigma_{x,n}^2$ and $\sigma_{s,n}^2$ respectively) are also obtained using a first-order attack-release IIR filter. The reference signal power estimate is shown as an example.

$$\Delta_n = \max \left\{ L/D \cdot (N-1) \cdot \sigma_{x,n}^2, \quad L \cdot \sigma_{s,n}^2 \right\}$$

$$\delta_n = \begin{cases} \Delta_n & \text{if } \Delta_n \geq \delta_{n-D} \\ (1 - \alpha_\delta) \Delta_n + \alpha_\delta \delta_{n-D} & \text{if } \Delta_n < \delta_{n-D} \end{cases} \quad \alpha_\delta = 1 - \left(\frac{F_s}{R \cdot D} \right)^{-1} \quad (4)$$

$$\sigma_{x,n}^2 = \begin{cases} |x_n|^2 & \text{if } |x_n|^2 \geq \sigma_{x,n-1}^2 \\ (1 - \alpha_\sigma) |x_n|^2 + \alpha_\sigma \sigma_{x,n-1}^2 & \text{if } |x_n|^2 < \sigma_{x,n-1}^2 \end{cases} \quad \alpha_\sigma = 1 - L^{-1}$$

Note that the attack-release IIR filters both have instant attacks to ensure stability of the adaptive algorithm. The IIR filters for the signal power estimates each have a release time-constant set to the filter tap-length (determined by α_σ in (4)) because it is approximately the last L samples of the reference signal that influence adaptation and the estimate of the autocorrelation matrix.

The regularization parameter's attack-release IIR filter has a release time-constant of one second as specified by α_δ in (4). While this choice is not critical, it must be long enough to ensure that the regularization parameter does not drop too quickly, especially during the short pauses between words when the reference signal or near-end disturbance consists of speech.

5. SYSTEM EVALUATION

For system evaluation, an oversampled GDFT filterbank with a Weighted Overlap-Add (WOLA) implementation (detailed in [1,5]) was used with the following parameters: The analysis and synthesis window lengths were 128 and 64 samples, respectively. The number of complex bands was $K=16$; the filterbank decimation rate (input frame shift) was $R=4$; and the stacking of the GDFT filterbank was odd.

For the first test case, white noise was used as the reference signal. The primary signal consisted solely of the echo with no near-end disturbance. In order to generate the echo, the eighth plant of the ITU-T G. 168 standard, [11], was used. The echo signal was then normalized to have an echo return loss (ERL) of 10 dB. Each of the complex subband adaptive filters had a tap-length of $L=32$ samples, an affine order of $N=4$ and used a step-size of $\mu=1$ with online regularization of Section 4.1. Fig. 2-A depicts the echo return loss enhancement (ERLE) for the APA and the GS-FAP, GS-PAP and modified GS-PAP algorithms. As expected, the affine projection algorithm outperforms all other algorithms. However, ERLE curves of the GS-FAP, GS-PAP and modified GS-PAP algorithms are indistinguishable. This result is not surprising as the key approximation in deriving the GS-FAP [3] is also present in the GS-PAP [8,9]. The approximation allows for reducing the matrix inversion in Eq. (1) to solving a set of N linear equations as

$$\mathbf{R}_n \underline{P}_n = \begin{bmatrix} \mathbf{I} & \mathbf{0}^T \end{bmatrix}^T \text{ in Step 3 of GS-PAP}$$

5.1 Sequential GS-PAP Evaluations

The SGS-PAP algorithm (using the last option in Step 5 of SGS-PAP which is the simplest) was also evaluated through simulation. The first test case it went through was similar to the test case described above using white noise as the reference signal. The only difference between this test and the one described above is that a fixed regularization parameter was used, and the step-size was no longer fixed to one. Instead, for different decimation rates of $D=1,2,4,8$, the step-size was set to be dependent on the decimation rate as $\mu=0.125, 0.25, 0.5, 1$, respectively. The regularization parameter was set to $\delta=10^{11}$ considering the worst-case scenario from Eq. (4), $\max \{ L/D \cdot (N-1) \cdot \sigma_{x,n}^2 \}$ assuming stationary noise as the subband reference signal with a power of around 10^9 . Fig. 2-B depicts the resulting ERLE curves. One can note from the figure that due to the step-size scaling, the different ERLE curves are indistinguishable. Identical behaviour is reported in [5] for the S-LMS algorithm.

Figs. 2-C,D depict the ERLE results for tests similar to Fig. 2-B but using the on-line regularization of Eq. (4). When the on-line regularization is used, the step-size μ should be fixed to unity for various D values since δ_n in (4) is inversely proportional to the sequential decimation rate. This compensates for the slower convergence of increased sequential decimation rates as confirmed by Figs. 2-C and 2-D.

The second test case used the same set-up as Fig. 2-C (on-line regularization and $\mu=1$) but with different signals. Here the reference signal consisted of a recorded male voice with additive white noise at SNR of 30 dB. The primary signal consisted of the echo and a near-end disturbance. The echo was generated using

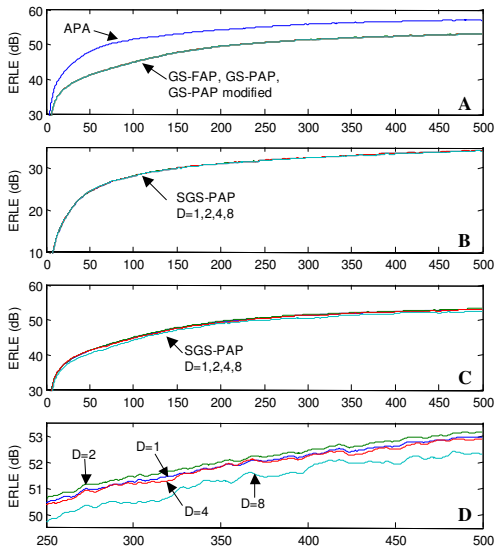


Figure 2: ERLE of white excitation with no near end disturbance for A) APA, GS-FAP, GS-PAP, B) SGS-PAP with fixed regularization and decimation dependent step-size, C) SGS-PAP with $\mu = 1$ and on-line regularization, D) Details of Curve C in the second half of time.

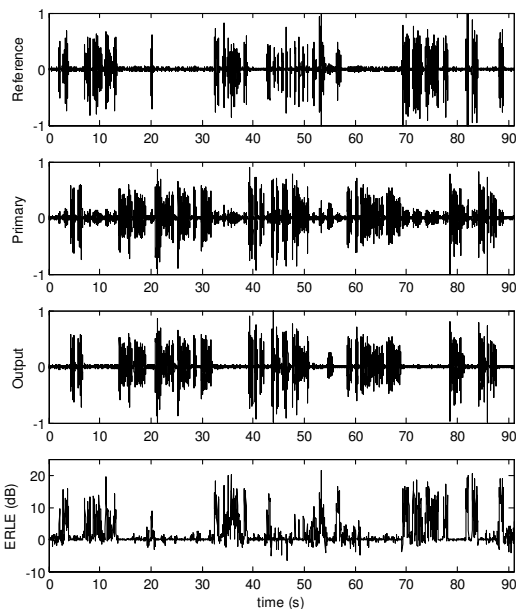


Figure 3: Reference, Primary and Output speech signals with the ERLE results using SGS-PAP with $D = 4$.

the same plant as before but was normalized for an ERL of 15 dB. The near-end disturbance was a recorded male voice with white noise added at an SNR of 30 dB. The test employed only SGS-PAP algorithm with $D = 4$ as it performed very similarly to the GS-PAP and GS-FAP. Fig. 4 depicts the reference, primary, and output signal waveforms of the algorithm together with the ERLE results. In the best case (reference single-talk), the echo signal power is 15 dB higher than the near-end noise floor. That is why the ERLE performance is limited to around 15 dB. The presented speech test together with many other similar tests

show that the adaptation method is robust to near-end disturbance and double-talk. Listening tests revealed no distortion in the near-end speech. Repeating the test with $D=1,2,8$ proved that all four decimation factors generate similar results. Hence the sequential GS-PAP algorithm proves to be quite suitable when used with speech inputs.

6. CONCLUSIONS

Complex GS-PAP algorithm is further simplified and employed for subband adaptive filtering on oversampled filterbanks. Sequential partial update PAP is introduced as a low-resource, efficient alternative algorithm to the PFU-FAPA. It was shown that unlike the PFU-FAPA, the SGS-PAP algorithm does not suffer from the aliasing problem due to decimation in sequential update. Moreover, with the proposed regularization scheme, increased decimation in the sequential update does not degrade the convergence performance. The SGS-PAP method is rigorously evaluated with speech signals in presence of noise and double-talk. The performance proved to be robust and consistent while maintaining a low computation cost.

7. REFERENCES

- [1] H. R. Abutabeh, H. Sheikhzadeh, R. L. Brennan, and G. H. Freeman, "Affine projection algorithm for oversampled subband adaptive filters," *Proc. ICASSP*, 2003, pp. VI 209–212.
- [2] S. L. Gay, Fast Projection Algorithms with Application to Voice Excited Echo Cancellers, doctoral dissertation, Rutgers Univ., Piscataway, N.J., Oct. 1994.
- [3] F. Albu et al., "The Gauss-Seidel fast affine projection algorithm," *IEEE Workshop on Signal Processing Systems 2002 (SIPS '02)*, Oct. 2002, pp. 109-114.
- [4] E. Chau, H. Sheikhzadeh and R. L. Brennan, "Complexity reduction and regularization of a fast affine projection algorithm for oversampled subband adaptive filters," *Proc. ICASSP*, May 2004.
- [5] H. Sheikhzadeh, H. R. Abutabeh, R. L. Brennan, K. R. L. Whyte, and E. Chau, "Sequential LMS for low-resource subband adaptive filtering: oversampled implementation and polyphase analysis," *Proc. EUSIPCO*, Sep. 2004.
- [6] E. Chau, H. Sheikhzadeh, and R. L. Brennan, "A polyphase model for fast affine projection with partial filter update", *Proc. EUSIPCO*, Sep. 2004.
- [7] F. Bouteille, P. Scalart and M. Corazza, "Pseudo affine projection algorithm new solution for adaptive identification," *Proc. Eurospeech*, 1999, vol. 1, pp. 427-430.
- [8] F. Albu and A. Fagan, "The Gauss-Seidel pseudo affine projection algorithm and its application for echo cancellation," *37th Asilomar Conf. Signals, Systems & Computers*, Nov. 2003.
- [9] F. Albu and M. Bouchard, "A low-cost and fast convergence Gauss-Seidel pseudo affine projection algorithm for multichannel active noise control," *Proc. ICASSP*, May 2004, vol. 4, pp. 121-124,
- [10] Sheikhzadeh, R. L. Brennan, and K. R. L. Whyte, "Near-end distortion in over-sampled subband adaptive implementation of affine projection algorithm", *Proc. EUSIPCO*, Sep. 7-10, 2004.
- [11] Recommendation ITU-T G.168, Digital Network Echo Cancellers, Int'l Telecommunication Union, 2000.