EFFICIENT IMPLEMENTATION OF THE AFFINE PROJECTION ALGORITHM FOR ACTIVE NOISE CONTROL APPLICATIONS

Miguel Ferrer, María de Diego, Alberto González and Gema Piñero

Audio and Communications Processing Group (GTAC). Departamento de Comunicaciones
Universidad Politecnica de Valencia, Camino de Vera s/n, 46022, Valencia, Spain
email: mferrer@dcom.upv.es, web:www.gtac.upv.es

ABSTRACT

In this paper, an efficient implementation of the Affine Projection algorithm (AP) is introduced in order to avoid the direct computation of matrix inversions. In the presented formulation, a recursive relation is devoted to calculate matrix inversion, thus improving the computational load (as much improvement as greater projection order). Furthermore, the proposed AP version has been introduced for multichannel Active Noise Control (ANC) and the new algorithm is named new efficient multichannel filtered-x affine projection algorithm (new efficient MFXAP). A comparative practical study of different AP implementations with the new approach has been also carried out and validates the use of the proposed method. Simulations and a practical implementation based on the TMS320C40 DSP took part in this study. The most meaningful results are shown throughout the paper.

1. INTRODUCTION

The use of adaptive algorithms for multichannel active noise control (ANC) [1] has been subject of continuous study and research since the decade of 80. The classical LMS algorithm applied to ANC systems is the widely used filtered-x LMS algorithm (FXLMS) [2]. The simplicity and robustness of the LMS algorithm together with the improvement of the computational capacities of new families of DSPs made possible several changes over the original structure of the LMS algorithms. Most of the times, the aim of these changes was the improvement of the convergence speed of the algorithm. An interesting alternative to the LMS algorithm is the Affine Projection algorithm (AP) [3], and their fast versions, Fast Affine Projection algorithms (FAP) [4], which were first proposed for single channel Active Noise Control (ANC) systems in [5] and for multichannel ANC in [6]. AP algorithms show a good tradeoff between computational effort and convergence speed preserving numerical stability. Nevertheless, the implementation of AP algorithms implies several matrix inversions that may become a drawback in a practical DSP implementation. Different methods have been recently proposed to improve the computational complexity in matrix inversions of AP algorithms. Recursive methods like the sliding window Fast Recursive Least Squares (FRLS), the sliding window Fast Transversal Filters (FTF), and the sliding window RLS [7] have been applied in fast versions of RLS and AP algorithms, nevertheless, numerical stability is not always assured. A multichannel ANC system using a FAP algorithm with sliding window RLS inverse matrix calculation was recently implemented in simulation [6]. Otherwise, the Gauss-Seidel method described in [8], which iteratively computes an approximation to inverse matrix, was introduced in [9] for a multichannel FAP algorithm. In this paper, the application to active noise control of the multichannel AP algorithm (standard MFXAP) is presented together with a recursive method to calculate the matrix inversions thus saving computational load with respect to the original formulation providing the new efficient multichannel filtered-x affine projection algorithm (new efficient MFXAP algorithm). In Section 2, we describe the standard MFXAP algorithm. In Section 3, the proposed recursive method to compute matrix inversions in the MFXAP algorithm instead of the direct computation is reported. The performance of the new efficient MFXAP algorithm will be compared with the standard MFXAP algorithm and also with two more algorithms derived from the MFXAP algorithm that apply alternative efficient techniques to compute inverse matrices: the MFXAP algorithm with an embedded sliding window RLS method (MFXAP-RLS) and the MFXAP algorithm with an embedded Gauss-Seidel technique (MFXAP-GS). In Section 4 simulation results for multichannel ANC systems comparing the four commented algorithms are presented. In Section 5 experimental results obtained in a real practical system are shown, concretely, the new efficient MFXAP and the multichannel FXLMS have been implemented. Section 6 concludes this work.

Partially supported by: CTDIA grant 2002/116 and CICYT grant TIC 2003-08238-C02.

2. MULTICHANNEL AFFINE PROJECTION ALGORITHM APPLIED TO ACTIVE NOISE CONTROL

Application of AP algorithms to ANC [5, 6] is similar to the implementation of gradient descent algorithms, among which the LMS algorithm [1] is the most commonly used. The filtered-x LMS (FXLMS) structure [2] applied to the AP algorithm for multichannel ANC systems, leads to the multichannel filtered-x affine projection algorithm (MFXAP). To describe this algorithm we will use the following notation:

- ANC system definition: I:J:K ANC system, where it means, *Number of reference signals:Number of secondary sources: Number of error sensors*.
- L: Length of adaptive filters.
- N: Affine projection order.
- M: Length of the FIR filters modelling the acoustic plants between the actuators and the error sensors.
- x_i(n): Value at time n of the ith reference signal.
- y_j(n): Value at time n of the signal at the jth actuator.
- e_k(n): Value at time n of the kth error sensor.
- $w_{i,j,l}(n)$: Value at time n of the lth coefficient in the adaptive filter linking x_i(n) and y_j(n).
- $\mathbf{w}_{i,j} = [w_{i,j,1}(n), w_{i,j,2}(n), ..., w_{i,j,L}(n)]^T$
- $h_{j,k,m}$: Value of the mth coefficient in the FIR filter modelling the plant between y_j(n) and e_k(n).
- $\mathbf{h}_{j,k} = [h_{j,k,1}, h_{j,k,2}, ...h_{j,k,M}]^T$
- $v_{i,j,k}(n)$: Value at time n of the reference signal x_i(n) filtered by the plant model $\mathbf{h}_{j,k}$.
- $\mathbf{v}_{i,j,k}(n)$=$[v_{i,j,k}(n), v_{i,j,k}(n-1), ...v_{i,j,k}(n-L+1)]^T$
- $\mathbf{V}_{i,j,k}(n) = [\mathbf{v}_{i,j,k}(n) \dots \mathbf{v}_{i,j,k}(n-N+1)]$
- $\mathbf{x}_i(n) = [x_i(n), x_i(n-1), ...x_i(n-L+1)]^T$
- $\mathbf{x'}_i(n) = [x_i(n), x_i(n-1), ...x_i(n-M+1)]^T$
- $\mathbf{E}_k(n) = [e_k(n), e_k(n-1), ... e_k(n-N+1)]^T$

According to the preceding notation the MFXAP algorithm would be described as follows:

$$v_{i,j,k}(n-1) = \mathbf{h}_{j,k}^T \mathbf{x'}_i(n-1), \quad (1)$$

$$\mathbf{w}_{i,j}(n) = \mathbf{w}_{i,j}(n-1) - \mu \sum_{k=1}^{K} \mathbf{V}_{i,j,k}(n-1) \\ \{\mathbf{V}_{i,j,k}(n-1)^T \mathbf{V}_{i,j,k}(n-1) + \delta \mathbf{I}\}^{-1} \mathbf{E}_k(n), \tag{2}$$

$$y_j(n) = \sum_{j=1}^{J} \mathbf{w}_{i,j}^T \mathbf{x}_i(n), \tag{3}$$

where in equation (2) we have included the convergence step, μ, and the regularization factor, δ, which controls numerical instability. If δ is not properly chosen the convergence speed may be affected. $\mathbf{I}$ represents the $N \times N$ identity matrix. It should also be noticed that in equations (1) and (2) the reference signals are delayed by one sample. As e_k(n)=d_k(n-1)+$\sum_{j=1}^{J} \mathbf{h}_{j,k}^T \mathbf{y}_j(n-1)$, and because of the mentioned delay, now signals coming from the reference signal ($\mathbf{V}_{i,j,k}(n-1)$) and signals coming from error sensors ($\mathbf{E}_k$(n)) are derived from the same iteration of the signal $x_i(n)$ thus meaningfully improving the convergence speed in practice.

In order to update the coefficients in each iteration (equation (2)), the calculation of the inverse of a $N \times N$ matrix is needed as well as matrix multiplications of sizes $(L \times N) \times (N \times N) \times (N \times 1)$. Previously developed fast versions of AP algorithm (FAP algorithms) [4, 5, 6] reduce computational complexity by means of efficient methods to invert these matrices like the sliding window RLS [6] and the Gauss-Seidel [9] techniques. Moreover, those FAP algorithms compute, instead of the normal coefficients, some auxiliary coefficients which require less operations to be updated. Likewise, for values of the convergence step μ close to unity, the coefficients update can be approximated with matrix multiplications of size $(L \times N) \times (N \times 1) \times (1 \times 1)$, achieving a meaningful computational saving as higher as the projection order increases [5]. However, it should be pointed out that ANC applications work well with small projection orders (lower than 5), see [6, 10], and in those cases the complexity of FAP algorithms may be similar to the complexity of the AP algorithm with a much simpler implementation. That is why an efficient version of the AP algorithm avoiding only direct computation of matrix inversions is proposed in this work and applied to the MFXAP algorithm resulting in the new efficient MFXAP algorithm. In fact, this paper shows hereafter how to avoid these matrix inversions using the previous inverse matrix and an iterative computation. The recursive inversion method can be used to compute the matrix inversion of order $N \times N$ in equation (2). The proposed algorithm is as robust as the standard MFXAP algorithm and with a computational complexity similar to the MFXAP-GS and the MFXAP-RLS. Even these last two algorithms can exhibit a poorer performance in comparison with the new approach, with instabilities and initialization problems.

3. RECURSIVE IMPLEMENTATION OF THE MATRIX INVERSION

A recursive method is presented in order to obtain the exact matrix inversion in eq. (2) which presents numerical stability and a computational cost similar to the sliding window RLS method [6].

From eq. (2) and assuming that:

$$\mathbf{M}(n) = \mathbf{V}_{i,j,k}(n)^T \mathbf{V}_{i,j,k}(n), \tag{4}$$

then the following relation can be written:

$$\mathbf{M}(n) = \mathbf{M}(n-1) + \mathbf{v}_{N_{i,j,k}}(n)\mathbf{v}_{N_{i,j,k}}^T(n) \\ - \mathbf{v}_{N_{i,j,k}}(n-L-N)\mathbf{v}_{N_{i,j,k}}^T(n-L-N), \tag{5}$$

where

$$\mathbf{v}_{N_{i,j,k}}(n) = [v_{i,j,k}(n), v_{i,j,k}(n-1), ..., v_{i,j,k}(n-N+1)]^T. \tag{6}$$

This recursive expression would save us operations before the computation of the inverse, but we still have to compute $\mathbf{M}^{-1}(n)$ for the actualization of the weights in the AP algorithms. However the matrix inversion lemma states that two positive definite matrices $\mathbf{A}$ and $\mathbf{B}$ of dimensions $N \times N$ fulfill the following relation:

$$\mathbf{A} = \mathbf{B}^{-1} + \mathbf{C}\mathbf{D}^{-1}\mathbf{C}^T. \tag{7}$$

Then we are able to compute $\mathbf{A}^{-1}$ as:

$$\mathbf{A}^{-1} = \mathbf{B} - \mathbf{B}\mathbf{C}(\mathbf{D} + \mathbf{C}^T\mathbf{B}\mathbf{C})^{-1}\mathbf{C}^T\mathbf{B}, \tag{8}$$

where $\mathbf{D}$ and $\mathbf{C}$ are other two matrices of $M \times M$ and $N \times M$ respectively. Now, using expressions (5) and (8) we can write:

$$\mathbf{M}(n) = \mathbf{M}(n-1) + \mathbf{F}(n)\mathbf{D}\mathbf{F}^T(n), \tag{9}$$

being $\mathbf{F}(n) = [\mathbf{v}_{N_{i,j,k}}(n), \ \mathbf{v}_{N_{i,j,k}}(n-L-N)]$ of dimensions $N \times 2$, and $\mathbf{D} = \mathbf{D}^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}$.

Then we can identify:

$$\mathbf{A} = \mathbf{M}(n), \tag{10}$$

$$\mathbf{B}^{-1} = \mathbf{M}(n-1), \tag{11}$$

$$\mathbf{C} = \mathbf{F}(n), \tag{12}$$

$$\mathbf{D}^{-1} = \begin{bmatrix} 1 & 0 \\ 0 & -1 \end{bmatrix}. \tag{13}$$

Furthermore, we can use the inversion lemma for matrices to compute $\mathbf{M}(n)^{-1}$:

$$\mathbf{M}(n)^{-1} = \mathbf{M}(n-1)^{-1} - \mathbf{M}(n-1)^{-1}\mathbf{F}(n)\{\mathbf{D}+ \\ +\mathbf{F}^T(n)\mathbf{M}(n-1)\mathbf{F}(n)\}^{-1}\mathbf{F}^T(n)\mathbf{M}(n-1)^{-1}. \tag{14}$$

This way we are able to compute the inverse matrix of $\mathbf{M}(n)$ from the inverse of that same matrix in the previous iteration plus a few operations that involve the computation of a 2×2 inverse matrix (almost reduced to the evaluation of a determinant). It is clear that computational time saving increases with increasing order of the matrix to invert (higher projection order). The recursive approach leads us to a feasible implementation of the MFXAP algorithm.

It should be noted that direct computation of matrix inversion in eq. (4) depends on N and L. However, computation using the recursive approach only depends on N, see eq. (14). In case of low projection orders of the MFXAP algorithm and adaptive filters with only a few coefficients, the recursive approach is not an improvement compared to the direct computation of the inverse matrix, but in case of a large length of the adaptive filters is a very useful tool to improve computational saving.

In the recursive inverse computation, an initial value of the inverse is needed as a starting point. A direct (non-recursive) computation of the inverse matrix could be carried out in a setup stage of the adaptive algorithm, thus it can start with an exact value of the inverse matrix. Nevertheless, experience has confirmed that using an estimation of the inverse as starting point the algorithm still works properly (even though the performance can be improved using an exact inverse initial value). An implementation using identity matrix multiplied by a small positive constant number shows results comparable to the results achieved with direct inverse computation.

4. SIMULATIONS OF THE MULTICHANNEL AFFINE PROJECTION ALGORITHMS FOR ANC

In order to test the performance of the new efficient MFXAP compared with the MFXAP, the MFXAP-RLS and the MFXAP-GS, several simulations have been carried out using *Matlab*^{TM} and real acoustic paths measured in an enclosed room. Computational cost and convergence speed were used to compare the different implementations. The measurement setup consisted in one loudspeaker acting as primary source, two loudspeakers acting as secondary sources and two error sensors. Therefore the system was configured with I=1, J=2 and K=2 (1:2:2 ANC system), see an scheme in Fig. 1. All acoustic paths (primary and secondary ones) have been modelled as FIR filters of 250 coefficients. A zero mean white noise signal has been used as perturbation signal. Adaptive filters of 50 coefficients and different projection orders have been applied.

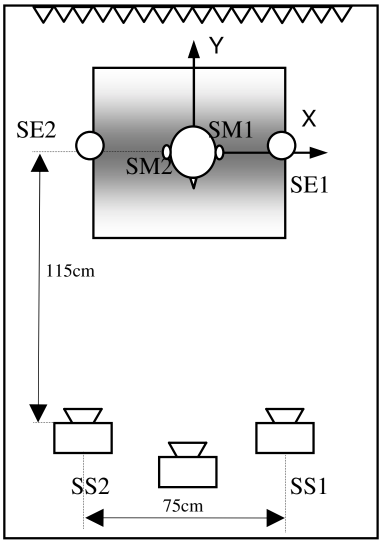

Figure 1: Scheme of the 1:2:2 ANC system

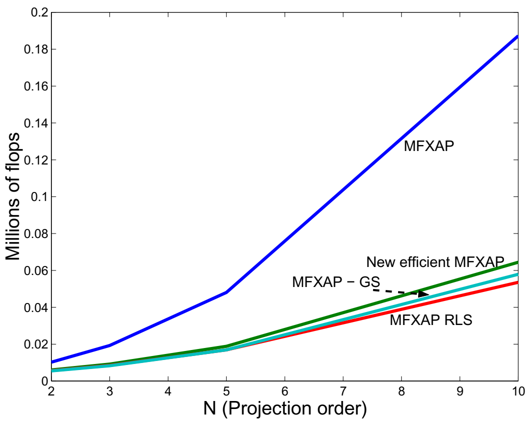

Figure 2: Comparison of the computational cost of the MFXAP, the new efficient MFXAP, the MFXAP-RLS and the MFXAP-GS for different projection orders.

The computational complexity of the algorithms were estimated by the number of flops required per iteration. Fig. 2 compares the performance of the four selected algorithms in terms of computational cost. As expected, the complexity of the new efficient MFXAP algorithm is significantly less than for the MFXAP algorithm. Therefore, the improvement increases with the projection order. The results of the new algorithm were found to be almost the same (up to the projection order $N = 5$) as for the MFXAP-RLS and the MFXAP-GS algorithms. Although the proposed algorithm is less efficient than the MFXAP-RLS and the MFXAP-GS algorithms for increasing projection orders, it has been empirically showed a superior performance in numerical stability and almost in convergence speed as it can be seen in Fig. 3.

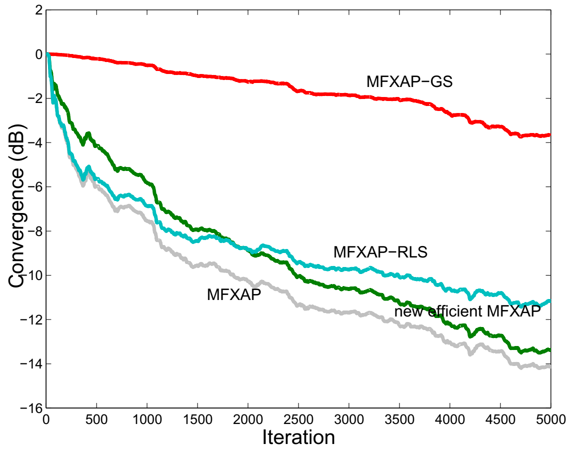

Figure 3: Convergence curves measured at error sensor SE1 for the four algorithms (N=10).

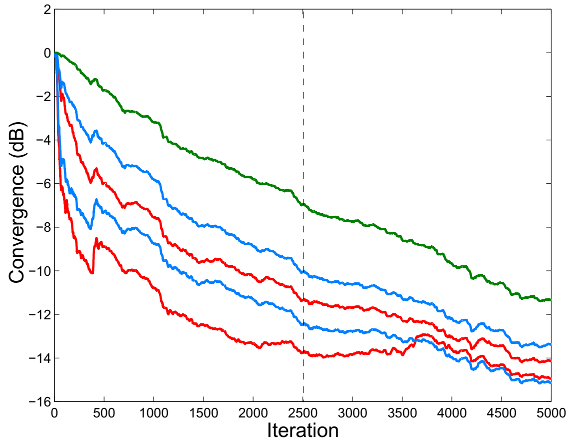

Figure 4: Convergence curves measured at error sensor SE1. From top to bottom at the dashed line position: multichannel FXLMS, new efficient MFXAP (N=5), new efficient MFXAP (N=10), MFXAP (N=5) and MFXAP (N=10).

In Fig. 3 and 4, those curves, called *convergence curves*, are obtained by plotting the ratio of the instantaneous estimated power at each error sensor over its initial value in decibels. Moreover, they provide information about the temporary evolution of the attenuation level measured at error sensors referred to its initial value in decibels.

In order to compare the convergence of the new MFXAP algorithm for projection orders $N = 5$ and $N = 10$ with the classical multichannel FXLMS algorithm [2] and the standard MFXAP, Fig. 4 is shown. As expected, the convergence speed of the MFXAP algorithms, including the proposed version, over the multichannel FXLMS is considerable. Since in practice it may not always be possible to compute matrix inversions, the fact that the new efficient MFXAP has a good convergence speed (comparing with the standard MFXAP) and seems more robust to numerical instabilities with a similar computational load than the MFXAP-GS and MFXAP-RLS algorithms, are good reasons to consider this algorithm for practical implementations.

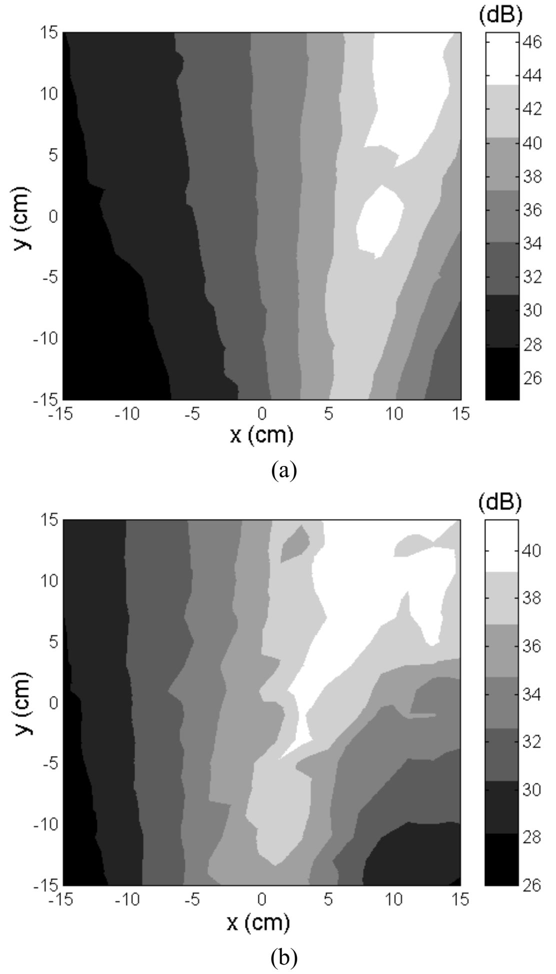

Figure 5: Real 1:1:2 ANC system. Attenuation noise levels measured at the left mannequin's microphone for a single tone using: (a) the FXLMS and (b) the new efficient MFXAP with N=2.

5. PRACTICAL ANC SYSTEM IMPLEMENTATION

We now investigate the behavior of the new MFXAP algorithm compared with the classical multichannel FXLMS algorithm in a real ANC system based on a TMS320C40 DSP processor. A 1:1:2 ANC system using the secondary loudspeaker *SS*1 in Fig. 1, working with a 80 Hz single tone disturbance signal, adaptive filters of 14 coefficients and a projection order of $N = 2$, has been tested measuring the attenuation levels achieved into a controlled zone of $30 \times 30\ cm^2$ placed between the error sensors. A mannequin with two calibrated microphones at the ear canals, was displaced into the area using a mobile platform in order to monitor the noise levels at different points of the zone.

Fig. 5 shows the attenuation levels measured at the left ear of the mannequin using the multichannel FXLMS and the new MFXAP. Meaningful *quiet zones* can be observed in both cases. It must be noted that these two algorithms exhibit similar final attenuation levels, the new MFXAP is faster than the FXLMS and its performance can be significantly improved increasing its projection order, as it is shown in Fig. 4.

6. CONCLUSIONS

In this paper, a computationally efficient MFXAP algorithm has been presented, the new efficient MFXAP. The proposed algorithm avoids the direct computation of matrix inversions, which are now calculated by means of a recursive form based on the matrix inversion lemma. It was shown through simulations that the new algorithm provides a performance as good as other MFXAP algorithms in terms of convergence speed and attenuation levels. The computational cost of the proposed approach is significantly better than the standard MFXAP algorithm and similar to the complexity of the MFXAP-GS and MFXAP-RLS algorithms. Finally, experiments in a real ANC system validate the use of the new efficient MFXAP algorithm compared with the classical multichannel FXLMS algorithm.

REFERENCES

[1] S.M. Kuo and D. Morgan, *Active Noise Control Systems*, John Wiley-Sons, 1996.

[2] B. Widrow and S.D. Stearns, *Adaptive Signal Processing*, Englewood Cliffs, N.J: Prentice-Hall, 1985.

[3] K. Ozeki and T. Umeda, "An adaptive filtering algorithm using an orthogonal projection to an affine subspace and its properties," *Proceedings of the Elec. Comm. Japan*, vol. J67-A, pp. 126–132, February 1984.

[4] S.L. Gay and S. Tavathia, "The fast affine projection algorithm," *Proceedings of the International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, vol. 5, pp. 3023–3026, May 1995, Detroit, MI.

[5] S.C. Douglas, "The fast affine projection algorithm for active noise control," *Proceedings of the 29th Asilomar Conference on Signals, Systems and Components*, vol. 2, pp. 1245–1249, October 1995, Pacific Grove,CA.

[6] M. Bouchard, "Multichannel affine and fast affine projection algorithms for active noise control and acoustic equalization systems," *IEEE Transactions on Speech and Audio Processing*, vol. 11, no. 1, pp. 54–60, January 2003.

[7] J.M. Cioffi and T. Kailath, "Windowed fast transversal filters adaptive algorithms with normalization," *IEEE Transactions on Acoustics, Speech and Signal Processing*, vol. 33, no. 3, pp. 307–625, June 1985.

[8] M. Barret et alt., *Templates for the solutions of linear systems: Building blocks for iteratives methods*, SIAM, 1994.

[9] M. Bouchard and F. Albu, "The multichannel gauss-seidel fast affine projection algorithm for active noise control," *Proceed. of the 7th International Simposium on Signal Processing and its applicactions (ISSPA)*, pp. 1–4, July 2002.

[10] M Ferrer, A. González, P. Zuccarello, and A. Camacho, "On the practical implementation of multichannel adaptive filters based on LMS, RLS, FTF and FAP algorithms for active control," *Proc. of the Tenth International Congress on sound and vibration*, July 2003.