Sparsity-Aware Pseudo Affine Projection Algorithm for Active Noise Control

Felix Albu*, Amelia Gully† and Rodrigo de Lamare†
*Department of Electronics, Valahia University of Targoviste, Targoviste, Romania
E-mail: felix.albu@valahia.ro; Tel: +40-245-217-683
†Department of Electronics, University of York, UK
E-mail: ajg540@york.ac.uk, rcdl500@york.ac.uk

Abstract—**This paper describes new algorithms promoting sparsity in modified filtered-x algorithms for active noise control. The proposed algorithms are based on recent techniques incorporating approximations to the ℓ_0-norm in the cost function used. An approximation to the affine projection algorithm leads to new zero-attracting and reweighted zero-attracting modified filtered-x pseudo affine projection algorithms. The results of simulations indicate that the proposed techniques demonstrate good performance and reduce numerical complexity compared to the original affine projection based algorithms.**

I. INTRODUCTION

Active noise control (ANC) is a technique for removing noise from a system by subtracting the effect of a noise-generating plant from a signal [1]. ANC algorithms are based on classical adaptive algorithms, and take into account the additional electro-acoustic path between the filter output and measured error signal, known as the secondary path [2]. A common technique to account for this path is known as the filtered-x (FX) scheme, originally developed for least-mean square (LMS) algorithms [3] and since incorporated into other adaptive algorithms, including the affine projection (AP) algorithm [4]-[6]. The FX scheme eliminates potential algorithm instability caused by the additional delay in the secondary path [7], but it generally exhibits slow convergence speed [8]. The modified filtered-x (MFX) scheme was introduced in [9], and improves upon the convergence of FX algorithms by introducing additional filtering steps to approximate the instantaneous error signal.

Recent developments in the field of compressive sensing have led to the introduction of sparsity-promoting penalties in adaptive filtering algorithms [10]-[11], producing zero-attracting (ZA) and reweighted zero-attracting (RZA) algorithms. These have been shown to provide faster convergence when the system in question has a degree of sparsity, which is often the case in the systems encountered in ANC. In [12], these sparsity constraints were incorporated into MFX algorithms and shown to improve performance for systems with a degree of sparsity. However, these algorithms suffer from high computational complexity.

In this paper, two approximations are proposed that lead to significant complexity reduction in the zero-attracting and reweighted zero-attracting MFxAP algorithms of [12]. Experimental results demonstrate that the proposed algorithms do not exhibit significant performance loss for systems with a moderate degree of sparsity. In particular, there is almost no performance loss when the step size is close to one, or in case of non-sparse echo paths.

The rest of this paper is structured as follows. In Section II the sparsity-inducing modified filtered-x algorithms from [12] are presented. In Section III the proposed algorithms are developed with reference to the ZA-MFxAP and RZA-MFxAP algorithms from which they derive. In Section IV, the results of simulation trials performed with these algorithms are presented and further directions of research are described. Section V presents the conclusions.

Notation: Throughout this paper, uppercase boldface letters will be used to denote matrices, and lowercase boldface letters to denote vectors. $Sgn\{\cdot\}$ is the signum function, $\mathbf{I}$ is an identity matrix of appropriate dimensions, and $\|\cdot\|_p$ denotes the pth norm of a vector. All vectors are column vectors. Following convention, the term ℓ_0-norm and $\|\cdot\|_0$ notation denotes the number of non-zero elements in a vector.

II. SPARSITY-INDUCING FILTERED-X ALGORITHMS

In broadband feedforward ANC, an error signal is used to update an adaptive filter such that its output can be 'subtracted' from the acoustic signal at the listener through interference cancellation, reducing disturbing noise. However, the nature of an ANC system introduces a *secondary path*, comprising the transfer functions of each electro-acoustic component. It was shown in [7] that this path can be split into two parts, one estimated as part of the plant, and one occurring after the acoustic summing junction, denoted here by $\mathbf{s}(n)$. In FX algorithms, an estimate of the secondary path, $\hat{\mathbf{s}}(n)$, is used to filter the input signal $x(n)$, so that the algorithm input is the filtered signal $x_f(n)$. This accounts for the secondary path delay, but introduces strict limits on step size that negatively impact upon convergence speed [8]. The MFX algorithms improve the convergence speed by estimating the instantaneous error signal $\hat{e}(n)$ [9].

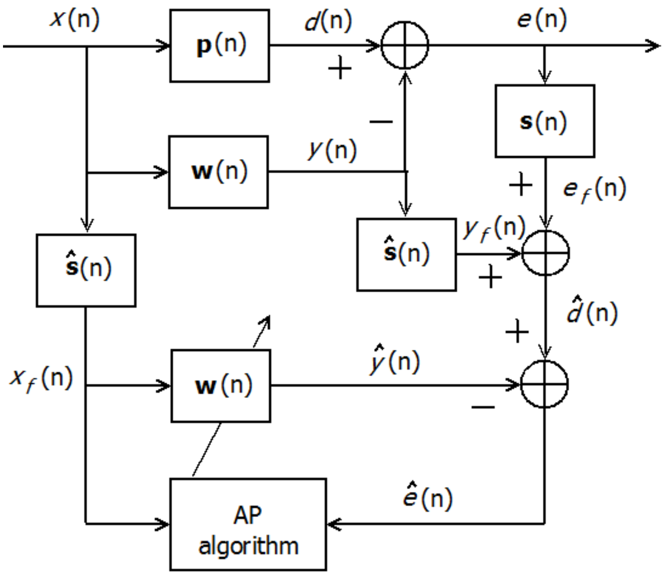

Fig. 1. Modified filtered-x AP structure [12]

The MFxAP structure is illustrated in Fig. 1. It is well suited to ANC as it does not require that $\mathbf{d}(n)=\mathbf{U}(n)\mathbf{w}(n+1)$ be available, where $\mathbf{U}(n)=[\chi(n)\ \chi(n-1)\ ...\ \chi(n-L)]$ is the regressor matrix, $\chi(n)=[x(n),x(n-1),...x(n-K+1)]^T$, K is the projection order and L is the filter length. Instead, the following condition is set:

$$\hat{\mathbf{d}}(n)=\mathbf{U}_f(n)\mathbf{w}(n+1) \tag{1}$$

We define the *filtered* regressor matrix as $\mathbf{U}_f(n)=[\chi_f(n)\ \chi_f(n-1)\ ...\ \chi_f(n-L)]$, where $\chi_f(n)=[x_f(n),x_f(n-1),...x_f(n-K+1)]^T$. Therefore, condition (1) states that the *estimated* output for the next iteration should equal the *estimated* desired signal for the current iteration. We have $\hat{\mathbf{y}}(n)=\mathbf{U}_f(n)\mathbf{w}(n)$, $\hat{\mathbf{e}}(n)=\mathbf{U}_f(n)\mathbf{U}_f^H(n)\underline{\lambda}$, and $\underline{\lambda}=\left[\mathbf{U}_f(n)\mathbf{U}_f^H(n)\right]^{-1}\hat{\mathbf{e}}(n)$ [12]. Therefore the MFxAP recursion is given by:

$$w(n+1)=w(n)+\mu\mathbf{U}_f^+(n)\hat{\mathbf{e}}(n) \tag{2}$$

where $\mathbf{U}_f^+(n)=\mathbf{U}_f^H(n)\left[\mathbf{U}_f(n)\mathbf{U}_f^H(n)+\delta\mathbf{I}\right]^{-1}$ and μ is the step size.

Next, the sparsity-inducing filtered-x affine projection algorithms proposed in [12] are presented. These are obtained by incorporating the zero-attracting and reweighted zero-attracting strategies considered in [10] into the modified filtered-x recursion (2).

II.1. Zero-attracting MFxAP (ZA-MFxAP) algorithm

The ZA-MFxAP incorporates a sparsity-inducing penalty to attract coefficients towards zero. Zero-attracting algorithms use an ℓ_1-norm penalty as an approximation and the weight update recursion is [12]:

$$\begin{aligned} w(n+1)=w(n)&+\mu\mathbf{U}_f^+(n)\hat{\mathbf{e}}(n)\\ &+\rho\mathbf{U}_f^+(n)\mathbf{U}_f(n)\operatorname{sgn}\{\mathbf{w}(n)\}\\ &-\alpha\operatorname{sgn}\{\mathbf{w}(n)\}\end{aligned} \tag{3}$$

where $\rho=\mu\alpha$ is known as the zero-attraction strength.

II.2. Reweighted zero-attracting MFxAP (RZA-MFxAP) Algorithm

The RZA-MFxAP algorithm uses a log-sum penalty in place of the ℓ_1-norm, as this provides a closer approximation to the behavior of the ℓ_0-norm. The RZA-MFxAP recursion is [12]:

$$\begin{aligned} w(n+1)=w(n)&+\mu\mathbf{U}_f^+(n)\hat{\mathbf{e}}(n)\\ &+\rho'\mathbf{U}_f^+(n)\mathbf{U}_f(n)\mathbf{\Psi}(n)-\gamma\varepsilon\mathbf{\Psi}(n)\end{aligned} \tag{4}$$

where $\mathbf{\Psi}(n)=\dfrac{\operatorname{sgn}\{\mathbf{w}(n)\}}{1+\varepsilon\operatorname{sgn}\{\mathbf{w}(n)\}}$.

In this case the strength of the zero-attraction is controlled by $\rho'=\mu\gamma\varepsilon$ where ε is known as the shrinkage magnitude.

III. THE PROPOSED ALGORITHMS

We found by simulations that for all investigated sparseness levels, the usual step size values and typical ANC situations the common term of MFxAP recursion has the greatest contribution to the performance for both ZA-MFxAP and RZA-MFxAP algorithms. For the ZA-MFxAP, we find:

$$\left\|\mu\mathbf{U}_f^+(n)\hat{\mathbf{e}}(n)\right\|_2 \gg \left\|\left(\rho\mathbf{U}_f^+(n)\mathbf{U}_f(n)-\alpha\right)\operatorname{sgn}\{\mathbf{w}(n)\}\right\|_2 \tag{5}$$

For the RZA-MFxAP, we find:

$$\left\|\mu\mathbf{U}_f^+(n)\hat{\mathbf{e}}(n)\right\|_2 \gg \left\|\rho'\mathbf{U}_f^+(n)\mathbf{U}_f(n)\mathbf{\Psi}(n)-\gamma\varepsilon\mathbf{\Psi}(n)\right\|_2 \tag{6}$$

This is illustrated by a simulation example shown in Fig. 2 for a non-sparse plant. Also, we note that for a step size close or equal to 1 we have:

$$\hat{\mathbf{e}}(n)\approx\left(e(n),0...0\right)^T \tag{7}$$

These observations allow us to reduce both the computational complexity and the memory requirements of the ZA-MFxAP and RZA-MFxAP algorithms in a similar way to that of [13].

If we denote the first column of $\mathbf{U}_f^+(n)$ as $\mathbf{U}_{f1}^+(n)$, and use (7), we obtain:

$$\mathbf{U}_f^+(n)\hat{\mathbf{e}}(n)=\mathbf{U}_{f1}^+(n)e(n) \tag{8}$$

This step reduces the numerical complexity from KL to L. Therefore the complexity reduction is higher for high projection orders.

By including (8) in (3) we obtain the weight recursion of a new algorithm called the Zero-Attracting Modified Filtered-X Pseudo Affine Projection (ZA-MFxPAP):

$$\begin{aligned} w(n+1)&=w(n)+\mu\mathbf{U}_{f1}^+(n)e(n)\\ &+\rho\mathbf{U}_f^+(n)\mathbf{U}_f(n)\operatorname{sgn}\{\mathbf{w}(n)\}\\ &-\alpha\operatorname{sgn}\{\mathbf{w}(n)\}\end{aligned} \tag{9}$$

Similarly, using (8) in (4) we obtain the weight recursion of the Reweighted Zero-Attracting Modified Filtered-X Pseudo Affine Projection (RZA-MFxPAP) as follows:

$$\begin{aligned} w(n+1)&=w(n)+\mu\mathbf{U}_{f1}^+(n)e(n)\\ &+\rho'\mathbf{U}_f^+(n)\mathbf{U}_f(n)\mathbf{\Psi}(n)-\gamma\varepsilon\mathbf{\Psi}(n)\end{aligned} \tag{10}$$

Additional computational savings can be obtained from an efficient computation of $\mathbf{R}(n)=\left[\mathbf{U}_f(n)\mathbf{U}_f^H(n)+\delta\mathbf{I}\right]$ as follows [13]:

$$\mathbf{R}(n)=\begin{bmatrix} r_0(n) & \underline{\mathbf{r}}^T(n)\\ \underline{\mathbf{r}}(n) & \bar{\mathbf{R}}(n-1)\end{bmatrix} \tag{11}$$

where

$$\begin{aligned}\mathbf{r}(n)&=\mathbf{r}(n-1)+x_f(n)\boldsymbol{\chi}_f(n)-x(n-L)\boldsymbol{\chi}_f(n-L)\\ &=\left[r_0(n),\underline{\mathbf{r}}^T(n)\right]^T\end{aligned} \tag{12}$$

and $\bar{\mathbf{R}}(n-1)$ is the top left $(K-1)\times(K-1)$ submatrix of $\mathbf{R}(n-1)$. Therefore, the update of $\mathbf{R}(n)$ needs only $2K$ multiplications instead of K^2L multiplications. The approximation in (7) leads to other computational and memory savings due to use of scalars instead of K length vectors. The complexity of each algorithm in terms of multiplications is given in Table 1. In these complexity calculations we ignore the cost of inversing a $K\times K$ matrix, $O\left(K^3\right)$ [14]. These results show that the complexity reduction from ZA-MFxAP to ZA-MFxPAP is the same as that between RZA-MFxAP and RZA-MFxPAP, and is given by:

$$C_{REDUCTION}=K^2L+6L(K-6)-2K \tag{13}$$

It can be seen that there is a small complexity difference between the original algorithms and the proposed ones.

Algorithm	**Number of multiplications per iteration**
ZA-MFxAP	$3K^2L + L^2(K+1) + 7KL + 3L$
ZA-MFxPAP	$2K^2L + L^2(K+1) + K(L+2) + 9L$
RZA-MFxAP	$3K^2L + L^2(K+1) + 7KL + 4L$
RZA-MFxPAP	$2K^2L + L^2(K+1) + K(L+2) + 10L$

Table 1. Algorithm complexity in terms of multiplications per iteration

IV. Simulation Results

This section compares the results of simulations of the proposed algorithms outlined in (9) and (10) with those of the conventional ZA-MFxAP and RZA-MFxAP algorithms of (3) and (4). The results are averaged over 10 simulation trials.

Three types of plant were used to perform simulations: a non-sparse path (density 785/800), a partially-sparse path (density 73/800) and a sparse path in which the fourth coefficient is set to one and all remaining coefficients to zero (density 1/800), all of which are identical to those used and described in [12]. Three types of secondary path, also identical to those in [12], were also generated. In Fig. 2 the non-sparse plant was used and the ZA-MFxAP and RZA-MFxAP algorithms were run for 12,000 iterations with the secondary path set as sparse at the start of the experiment, changed to partially-sparse at iteration 4,000 and to non-sparse at iteration 8,000. It can be seen that the norms of the same terms of weight update for MFxAP are much greater than those of the specific terms of the zero-attracting versions from [12].

Figs. 3-5 show mean-square deviation (MSD) convergence curves for each of these experiments. For all the algorithms, the parameters were tuned to the same values as in [12], with the exception of step size, μ, for which a value of 1 was used throughout. For each primary path formulation, the algorithms were run for 220,000 iterations with the secondary path set as sparse at the start of the experiment, changed to partially-sparse at iteration 10,000 and to non-sparse at iteration 70,000, giving sufficient time for algorithm convergence in each case. The first 10,000 iterations in each figure illustrate algorithm performance when the secondary path is sparse. The advantages and disadvantages of ZA-MFxAP and RZA-MFxAP algorithms over FxAP and MFxAP were shown in [12] and as such are not considered here.

It is seen in Fig. 3a that as the secondary path density increases, differences between the convergence curves ZA-MFxAP and ZA-MFxPAP algorithms start to appear. The proposed algorithms exhibit faster convergence, but also a higher steady state MSD. The same conclusions can be obtained from Fig. 3b regarding the RZA-MFxAP and RZA-MFxPAP when the projection order is 4. However, when the projection order is increased to 16, the original algorithms perform better than the proposed algorithms.

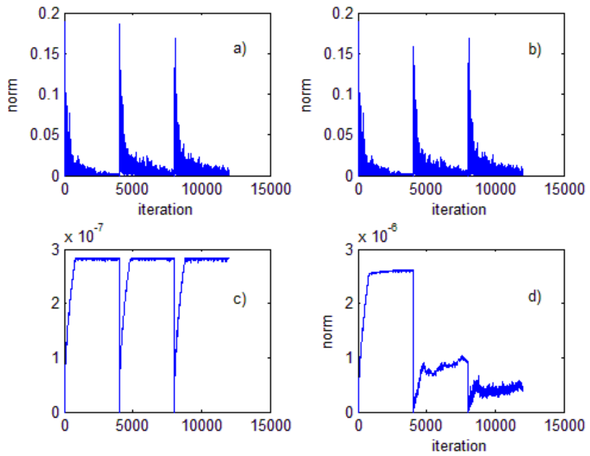

Fig. 2. a) The norm of the left side of (5) for ZA-MFxAP algorithm; b) The norm of the left size of (6) for RZA-MFxAP algorithm; c) The norm of the right side of (5) for ZA-MFxAP algorithm; d) The norm of the right side of (6) for RZA-MFxAP algorithm.

It can be deduced that for a sparse plant, the simplifying approximations used in the derivation of the proposed algorithms are not effective.

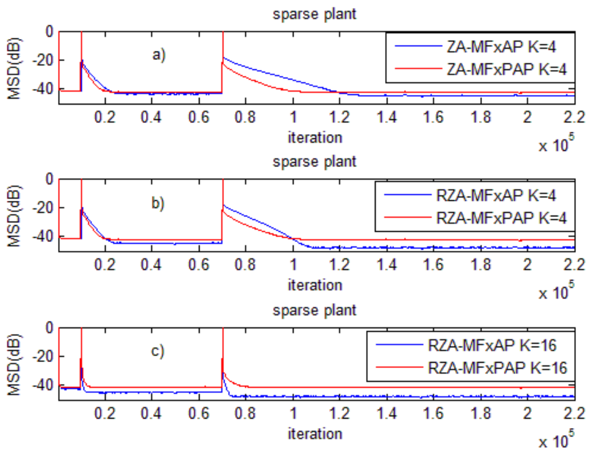

Fig. 3. MSD results for sparse plant.

The performance of the algorithms when the plant is partially sparse can be seen in Fig. 4. When the secondary path is semi-sparse, the ZA-MFxPAP algorithm exhibits similar behavior – in terms of both MSD convergence speed and steady-state performance – to the ZA-MFxAP algorithm (Fig. 4a).

In the case of the RZA-MFxPAP algorithm, as the secondary path density increases in Fig. 4b and 4c, steady-state differences are clearly visible between RZA-MFxAP RZA-MFxPAP.

The results of running the algorithms for a non-sparse plant can be seen in Fig. 5. In this case, the performance of the proposed algorithms is almost identical to that of the original algorithms. This recommends these numerically less complex algorithms as an alternative to FxAP, MFxAP, ZA-MFxAP and RZA-MFxAP algorithms, particularly for high projection orders, when the plant has a low degree of sparsity. Also, it can be noted that when the plant is non-sparse, the approximations used to derive ZA-MFxPAP and RZA-MFxPAP are valid. For all simulations considered here, when the secondary path is sparse, the proposed algorithms have almost identical performance to the original algorithms.

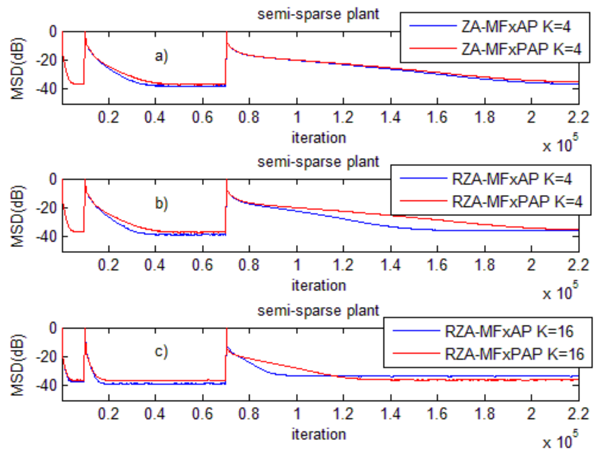

Fig. 4. MSD results for partially-sparse plant.

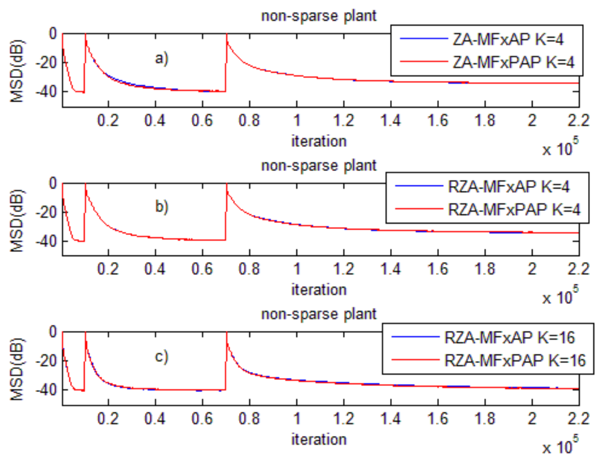

Fig. 5. MSD results for non-sparse plant.

Fig. 6 shows the multiplication saving of the proposed algorithms over the original ones. In Fig. 6a the path length was varied and the projection order was 16. In Fig. 6b the echo path length is 800 and the projection order is varied from 1 to 20. As expected from (13), the complexity saving is seen to have a linear relationship with respect to L, and a quadratic relationship with respect to K.

It can be seen that the computational complexity saving increases with the increase of the path lengths or projection order. Since there is only a difference of L multiplications between the zero-attracting versions and re-weighted zero-attracting versions the multiplication saving of the RZA-MFxPAP over the RZA-MFxAP is the same as those of ZA-MFxPAP over the ZA-MFxAP. However, the reweighed algorithms require L divisions per iteration, making them more complex to implement than the zero-attracting algorithms.

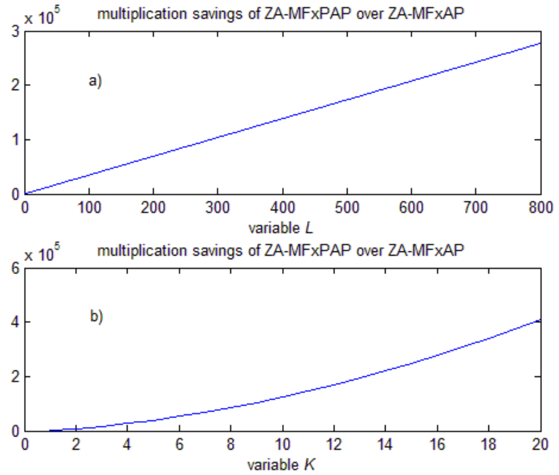

Fig. 6. a) The multiplication savings of ZA-MFxPAP over ZA-MFxPAP for a variable L; b) The multiplication savings of ZA-MFxPAP over ZA-MFxPAP for a variable K

Further work might incorporate recent improvements to sparse techniques [15]. It would therefore be worthwhile to consider algorithms incorporating a variable shrinkage magnitude parameter as an improvement to the algorithms proposed here, allowing steady-state performance to be improved without negatively affecting convergence rate. In addition, variable step-size versions or variable projection versions can be designed as in [5], and [16]. Also, a practical implementation is envisaged in order to verify the effectiveness of the proposed method.

V. CONCLUSIONS

This paper has proposed new adaptive algorithms that exploit sparsity and approximations to the error and weight update of the affine projection algorithm, based on modified filtered-x algorithms for active noise control. The simulation results demonstrate that with any degree of sparsity in the primary or secondary path, a good compromise between convergence speed, steady-state error and numerical complexity is obtained by the proposed ZA-MFxPAP and RZA-MFxPAP algorithms, particularly at a relatively high projection orders, such as 16, and for non-sparse echo paths.

Acknowledgment: This work was supported by a grant of the Romanian National Authority for Scientific Research, CNCS-UEFISCDI, project number PN-II-ID-PCE-2011-3-0097.

REFERENCES

[1] B. Widrow, J. R. Glover, J. M. McCool, J. Kaunitz, C. S. Williams, R. H. Hearn, J. R. Zeidler, E. Dong, and R. C. Goodlin, "Active noice cancelling: principles and applications," *Proc. IEEE*, vol. 63, no. 12, pp. 1692–1716, Dec. 1975.

[2] S. M. Kuo and D. R. Morgan, "Active noise control: a tutorial review," *Proc. IEEE*, vol. 87, no. 6, pp. 943–973, Jun. 1999.

[3] B. Widrow, D. Shur, and S. Shaffer, "On adaptive inverse control," in *Proc. 15th Asilomar Conf. Circuits Systems Computers*, Pacific Grove, CA, 1981, pp. 185–189.

[4] S. M. Kuo and D. R. Morgan, *Active Noise Control Systems: Algorithms and DSP Implementations*, John Wiley and Sons Inc., New York, NY, 1996.

[5] A. Gonzalez, F. Albu, M. Ferrer, and M. de Diego, "Evolutionary and variable step size strategies for multichannel filtered-x affine projection algorithms," *IET Signal Process.*, vol. 7, no. 6, pp. 471–476, Aug. 2013.

[6] M. Bouchard and F. Albu, "The Gauss-Seidel fast affine projection algorithm for multichannel active noise control and sound reproduction systems", Special Issue on Adaptive Control of Sound and Vibration, *International Journal of Adaptive Control and Signal Processing*, vol. 19, nr. 2-3, pp. 107-123, March-April 2005.

[7] E. Bjarnason, "Analysis of the filtered-x LMS algorithm," *IEEE Trans. Speech Audio Process.*, vol. 3, no. 6, pp. 504–514, Nov. 1995.

[8] M. Rupp and A. H. Sayed, "Modified FxLMS algorithms with improved convergence performance," in *Proc. 29th Asilomar Conf. Signals Systems Computers*, Pacific Grove, CA, 1995, pp. 1255–1259.

[9] E. Bjarnason, "Active noise cancellation using a modified form of the filtered-x LMS algorithm," in *Proc. 6th Eur. Signal Process. Conf.*, Brussels, Belgium, 1992, pp. 1053–1056.

[10] Y. Chen, Y. Gu, and A. O. Hero, "Sparse LMS for system identification," in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Taipei, Taiwan, 2009, pp. 3125–3128.

[11] R. Meng, R. C. de Lamare, and Nascimento V. H., "Sparsity-aware affine projection adaptive algorithms for system identification," in *Proc. Sensor Signal Process. Defence*, London, UK, 2011, MOD, pp. 1–5.

[12] A. Gully, R. C. de Lamare, "Sparsity aware filtered-x affine projection algorithms for active noise control", in *Proc. IEEE Int. Conf. Acoust. Speech Signal Process.*, Florence, Italy, 2014, pp. 6707-6711.

[13] F. Albu, M. Bouchard, Y. Zakharov, "Pseudo Affine Projection Algorithms for Multichannel Active Noise Control", *IEEE Trans. Audio, Speech Language Process.*, Vol. 15, Issue 3, March 2007, pp. 1044-1052.

[14] G. H. Golub and C. F. Van Loan, *Matrix computation*, 3rd edition. Baltimore, MD: The John Hopkins Univ. Press, 1996.

[15] R. C. de Lamare and R. Sampaio-Neto, "Sparsity-aware adaptive algorithms based on alternating optimization and shrinkage," *IEEE Signal Process. Lett.*, vol. 21, no.2, pp. 225–229, Feb. 2014.

[16] F. Albu, C. Paleologu and J. Benesty, "A Variable Step Size Evolutionary Affine Projection Algorithm", in Proc. *of ICASSP 2011*, Prague, Czech Republic, May 2011, pp. 429-432.