

Efficient Implementation of a Variable Projection Order Affine Projection Algorithm

F. Albu^{#1}, C. Paleologu^{#1}, and J. Benesty^{*2}

[#] Faculty of Electronics, Telecommunications, and Information Technology, University Politehnica of Bucharest
1-3 Bd. Iuliu Maniu, Bucharest, Romania

¹ felix@comm.pub.ro

¹ pale@comm.pub.ro

^{*} INRS-EMT, Université du Québec
QC H5A 1K6, Montreal, Canada

² benesty@emt.inrs.ca

Abstract—Many fast affine projection (FAP) algorithms have been proposed in the last two decades. These algorithms offer a good tradeoff between convergence rate and computational complexity. Most of the existing FAP algorithms use a constant projection order. Recently, the evolutionary APA (E-APA) with a variable projection order has been proposed. In this paper, an efficient implementation of the E-APA based on dichotomous coordinate descent (DCD) method is proposed and investigated. It is shown that the proposed algorithm obtains both fast convergence speed and small steady-state error.

I. INTRODUCTION

In echo cancellation systems, an adaptive filter algorithm is used to reduce the echo. The well-known normalized least-mean-square (NLMS) algorithm has been widely used in this context. Nevertheless, it converges slowly for acoustic echo cancellation (AEC) applications, where long length adaptive filters are used in order to model the acoustic echo paths. The affine projection algorithm (APA) [1] can be considered as a generalization of the NLMS algorithm that provides an improved convergence speed, especially for high-correlated signals, like speech. The NLMS algorithm corresponds to an APA with a projection matrix of rank equal to $L - 1$, where L is the length of the adaptive filter. In terms of convergence rate, the APA has a performance that rivals with the more complex recursive least-squares (RLS) algorithm in many situations. There are some fast and stable RLS versions suitable for AEC (e.g., the error feedback least-squares lattice proposed in [2]) that behave well numerically even in finite precision and have a computational cost that increases linearly with the order of the filter. However, their numerical complexity is still several times higher than that of the affine projection-type algorithms for typical projection orders [3].

Many fast affine projection (FAP) algorithms have been proposed for AEC systems (e.g., based on embedded fast RLS algorithm [4], Gauss-Seidel iterations [5], etc). They use some approximation that leads to inferior performance when using small step sizes. Efficient alternatives that have less multiplications use the dichotomous coordinate descent (DCD) method proposed in [6] and [7]. All previously mentioned FAP versions use a fixed projection order. It is known that if

the projection order increases, the convergence speed is faster, but the steady-state error also increases. A variable projection order might lead to a lower steady-state error. In [8], an affine projection algorithm with an evolving order, called evolutionary APA (E-APA) was proposed. Based on findings from [9] and [10], the projection order was modified depending on the relationship between the output error and a threshold. However, the authors do not investigate practical implementations with reduced numerical complexity.

The outline of the paper is as follows. The proposed evolutionary APA based on DCD iterations (DCD-E-APA) is described in Section II. In Section III, the behaviour of this algorithm for AEC systems is examined and compared with those of the algorithms using a fixed projection order. Section IV concludes this work.

II. DCD-E-APA

In AEC configuration (Fig. 1), the far-end signal, x_n , goes through the echo path \mathbf{h} , providing the echo signal, y_n . This signal is added with the near-end signal, v_n , (which can contain both the background noise and the near-end speech), resulting the microphone signal, d_n . The adaptive filter, defined by the vector $\hat{\mathbf{h}}_n$, aims to produce at its output an estimate of the echo, \hat{y}_n , while the error signal, e_n , should contain an estimate of the near-end signal.

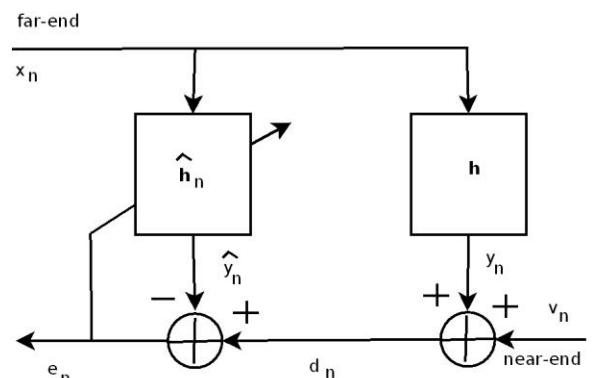


Fig. 1. AEC configuration.

Through this paper, the following notation will be used: K_n is the projection order of the E-APA at iteration n , K_{\max} is the maximum projection order, δ is a regularization parameter, μ is the step-size, \mathbf{I}_n denotes the $K_n \times K_n$ identity matrix, at time instant n , $\mathbf{d}_n = [d_n \ d_{n-1} \dots \ d_{n-K_n+1}]^T$ is the desired vector, $\mathbf{y}_n = [\hat{y}_n \ \hat{y}_{n-1} \dots \ \hat{y}_{n-K_n+1}]^T$ is the filter output vector, superscript T denotes transposition, $\mathbf{x}_n = [x_n \ x_{n-1} \dots \ x_{n-L+1}]^T$ is the input signal vector. Also, we have the data matrix $\mathbf{X}_n = [\mathbf{x}_n \ \mathbf{x}_{n-1} \dots \ \mathbf{x}_{n-K_n+1}]^T$, \mathbf{R}_n is the $K_n \times K_n$ auto-correlation matrix of the input signal, i.e., $\mathbf{R}_n = \mathbf{X}_n \mathbf{X}_n^T + \delta \mathbf{I}_{K_n}$, $\hat{\mathbf{h}}_n = [\hat{h}_{0,n} \dots \ \hat{h}_{L-1,n}]^T$ is the adaptive filter coefficients vector, $\xi_n = [x_n \ x_{n-1} \dots \ x_{n-K_{\max}+1}]^T$ is a $K_{\max} \times 1$ vector. Also, \mathbf{r}_n is a $K_{\max} \times 1$ autocorrelation vector, \mathbf{p}_n is a $K_n \times 1$ solution vector, and finally, \mathbf{e}_n is a $K_n \times 1$ vector.

The computation of the matrix \mathbf{R}_n can be made in an efficient way taking into account its symmetry. However, the size of this square matrix varies depending on the chosen projection order. There are two possible situations. In the first situation, if the projection order at time n is smaller than or equal to the projection order at time $n-1$, the matrix \mathbf{R}_n is updated by replacing the first row and column with the elements of \mathbf{r}_n , while the bottom-right $(K_n-1) \times (K_n-1)$ sub-matrix is replaced with the top-left $(K_n-1) \times (K_n-1)$ sub-matrix of \mathbf{R}_{n-1} . This updating rule is common to the APA or its fast versions for fixed projection orders [3-5]. In the second situation, if $K_n > K_{n-1}$, the matrix \mathbf{R}_n is updated by replacing the first row and column with the elements of \mathbf{r}_n , while the other elements are replaced with \mathbf{R}_{n-1} . The update of \mathbf{r}_n is made on full $K_{\max} \times 1$ vector, although only $K_n \times 1$ vector is needed for updating \mathbf{R}_n . The complexity of updating \mathbf{R}_n is reduced to only $2K_n$ multiplications. The equations that define the proposed DCD-E-APA are summarized in the following table. The noise variance, σ_v^2 , can be estimated online as in [10], [11], or during the periods of silence.

DCD-E-APA EQUATIONS

Initialization: (1)

$$\mathbf{x}_0 = \mathbf{0}_{L \times 1}, \hat{\mathbf{h}}_0 = \mathbf{0}_{L \times 1}, K_0 = K_{\max},$$

$$C_1 = \mu \sigma_v^2 / (2 - \mu), C_2 = 2\sigma_v^2 / (2 - \mu)$$

$$\mathbf{r}_0 = \mathbf{0}_{K_{\max} \times 1}, \xi_0 = \mathbf{0}_{K_{\max} \times 1},$$

For time index $n = 1, 2, \dots$

$$\mathbf{r}_n = \mathbf{r}_{n-1} + x_n \xi_n - x_{n-L} \xi_{n-L} \quad (2)$$

$$\mathbf{y}_n = \mathbf{X}_n^T \hat{\mathbf{h}}_{n-1} \quad (3)$$

$$\mathbf{e}_n = \mathbf{d}_n - \mathbf{y}_n \quad (4)$$

$$\eta_n = C_1 K_{n-1} + C_2 \quad (5)$$

$$\theta_n = \eta_n - C_1 \quad (6)$$

$$\text{IF } e_n^2 > \eta_n \quad (7)$$

$$K_n = \min\{K_{n-1} + 1, K_{\max}\}$$

$$\text{ELSE IF } e_n^2 \leq \theta_n \quad (8)$$

$$K_n = \max\{K_{n-1} - 1, 1\}$$

$$\text{ELSE} \quad (9)$$

$$K_n = K_{n-1}$$

$$\text{END} \quad (10)$$

$$\text{update } \mathbf{R}_n \text{ using } \mathbf{r}_n$$

$$\text{Solve } \mathbf{R}_n \mathbf{p}_n = \mathbf{e}_n \text{ using DCD} \quad (11)$$

$$\hat{\mathbf{h}}_n = \hat{\mathbf{h}}_{n-1} + \mu \cdot \mathbf{X}_n \mathbf{p}_n \quad (12)$$

The DCD algorithm updates a solution of a linear system of equations in directions of Euclidian coordinates in the cyclic order and with a step size α that takes one of M_b (number of bits) predefined values corresponding to a binary representation bounded by an interval $[-H, H]$ [6], [7]. The algorithm starts the iterative search from the most significant bits of the solution and continues until the least significant bits were updated. The algorithm complexity is limited by N_u , the maximum number of “successful” iterations [7]. The DCD method is implemented only with additions and bit shifts, and does not require multiplications. More details about the DCD algorithm can be found in [6] and [7]. In our implementations, for specific algorithm iteration where the chosen projection order is K_n , the DCD-E-APA has $(2L+1)K_n + 2K_{\max} + 2$ multiplications, while APA has $(2L+3)K_{\max} + O(K_{\max}^3)$ multiplications (the projection order is K_{\max}) [6] and E-APA with the presented matrix update procedure has $(2L+3)K_n + O(K_n^3)$.

III. SIMULATIONS

Simulations were performed in an AEC context as shown in Fig. 1. The NLMS, APA, E-APA, and DCD-E-APA were compared. The length of the adaptive filter is set to 512 coefficients. The measured impulse response of the acoustic echo path is plotted in Fig. 2a (the sampling rate is 8 kHz); its entire length has 1024 coefficients. This length is truncated to the first 512 coefficients for most of the experiments performed in an exact modeling case. Also, the entire length of the acoustic impulse response is used for one experiment performed in the under-modeling case. An independent white Gaussian noise signal is added to the echo signal, y_n , with 30 dB signal-to-noise ratio (SNR) for most of the experiments and $\mu = 0.2$.

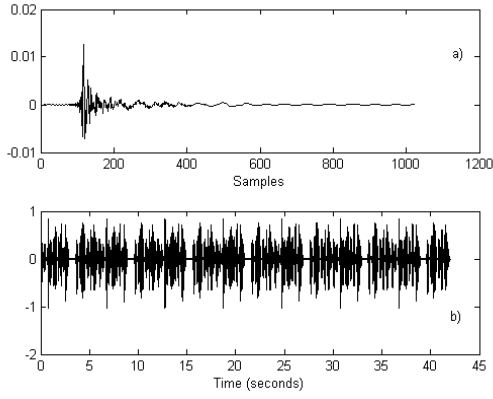


Fig. 2. (a) Measured room acoustic impulse response; (b) far-end speech signal used in the experiments.

The maximum considered value of the projection order for all simulations is $K_{\max} = 8$. In all the following experiments, in order to approach the context of typical AEC applications, the speech sequence from Fig. 2b is used as the far-end signal. The performance for the exact modeling scenario is evaluated in terms of the normalized misalignment (in dB), defined as $20\log_{10}\left[\frac{\|\mathbf{h}-\hat{\mathbf{h}}_n\|}{\|\mathbf{h}\|}\right]$, where $\|\bullet\|$ denotes the l_2 norm and \mathbf{h} is the true impulse response. Another measure used in this paper is the Echo Return Loss Enhancement (ERLE) typically used for AEC systems defined as $10\log_{10}\left(\frac{E[d_n^2]}{E[e_n^2]}\right)$.

The parameters of the DCD algorithm were $H = 2^{-10}$, $M_b = 16$, $N_u = 8$. They lead to a good compromise between performance and complexity (similar conclusions were obtained as in [6] and [7]).

Fig. 3a shows the misalignment curves in case of a single-talk scenario. In terms of the final misalignment, it can be seen that the variable projection order versions improve the overall performance of the fixed projection order APA (where p denotes its projection order). Also, it can be noticed the performance of DCD-E-APA is very close to that of E-APA. Both combine the fast convergence of the APA with a high projection order and the low misadjustment obtained with a low projection order. Initially, in the first convergence phase, most of projection orders are chosen closer to the maximum allowed projection order, while in the steady state phase the projection orders are close to the minimum allowed projection order. It can be noticed in Fig. 3b that most of projection orders are 1 (corresponds to NLMS) or 2. The ERLE performance of the considered algorithms is shown in Fig. 4 and confirms the conclusions of the misalignment simulations and the superiority of the APA over the NLMS algorithm. A similar conclusion was obtained from all subsequent ERLE plots (not shown here).

There are very small differences in the chosen projection orders between the E-APA and DCD-E-APA (Fig. 5).

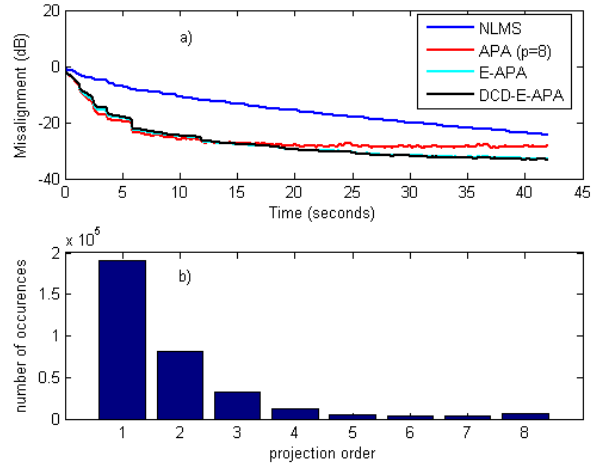


Fig. 3. a) Misalignment of NLMS, APA ($p=8$), E-APA, and DCD-E-APA. Single-talk case, $L = 512$, and $\text{SNR} = 30\text{dB}$; b) the number of occurrences of each projection order between 1 and 8 for DCD-E-APA.

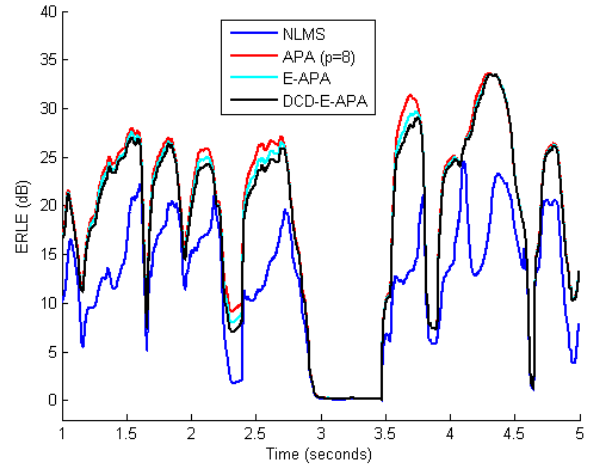


Fig. 4. The ERLE performance for the first 5 seconds of NLMS, APA ($p=8$), E-APA, and DCD-E-APA (conditions of Fig.3).

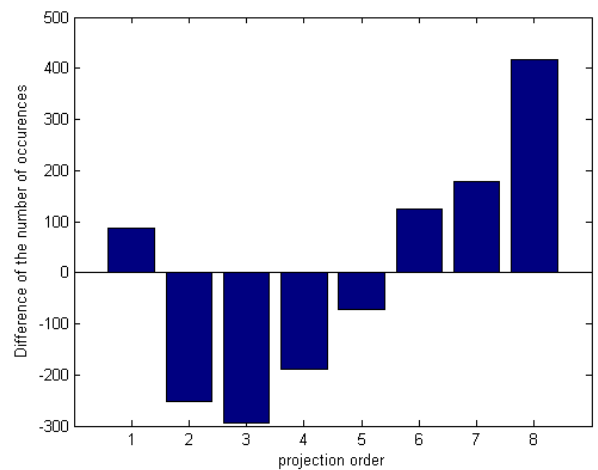


Fig. 5. The difference of the number of occurrences of each projection order between 1 and 8 of DCD-E-APA and E-APA (conditions of Fig.3).

The highest difference is small ($\sim 400 \ll 336000$ samples) for a projection order equal to 8 and it explains some small differences seen in Fig. 3a.

In Fig. 6, a sudden change in the echo path after 21 seconds is simulated. It can be noticed that both have better tracking performance than NLMS. Also, by comparing Fig. 6b with Fig. 3b, it can be noticed that the number of occurrences of small projection orders is reduced, and the number of occurrences of higher projection orders is increased.

In Fig. 7 the under-modelling case is investigated. The part of the echo path that is not modeled leads to increased misadjustment (seen as an additional noise by the system). It can be noticed from Fig. 7b that the number of occurrences of higher projection orders is much higher than in previous investigated cases. As expected, the misalignment performance in the under-modelling case (Fig. 7a) is worse than that of the exact modelling case (Fig. 6a).

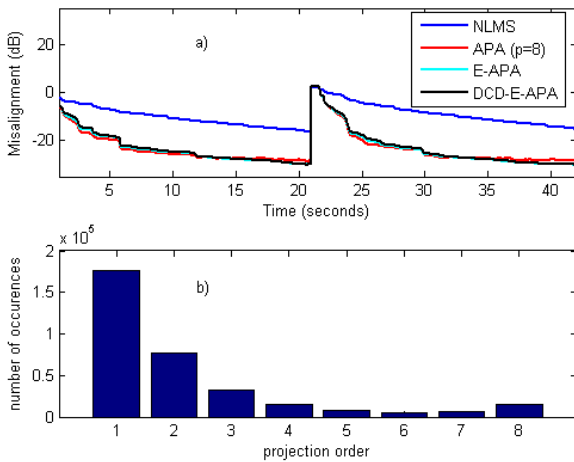


Fig. 6. a) Misalignment of NLMS, APA ($p=8$), E-APA, and DCD-E-APA. Single-talk case, echo path change after 21 seconds, $L=512$, and $\text{SNR}=30$ dB; b) the number of occurrences of each projection order between 1 and 8 for DCD-E-APA.

Like for E-APA [8], the convergence performance of the proposed DCD-E-APA is not very sensitive to the noise estimation error. Fig. 8 shows the misalignment difference in dB between DCD-E-APA with exact noise estimation σ_v^2 and DCD-E-APA using $0.5\sigma_v^2$ and $2\sigma_v^2$ respectively. It can be seen that the misalignment difference is smaller than 1 dB in absolute value. The same conclusion has been drawn in all considered cases.

The NLMS algorithm has around 1030 multiplications, while APA with $p=8$ has 8360 multiplications. The average number of multiplications of both E-APA and DCD-E-APA depends on the number of occurrences of each projection order. For the considered cases, for DCD-E-APA, it ranges from around 1940 for the case of Fig. 3 to around 3640 for the case investigated in Fig. 7. In most cases, the number of multiplications for the E-APA is slightly higher than those of DCD-E-APA. The complexity reduction is quite significant, especially for high values of the maximum projection order.

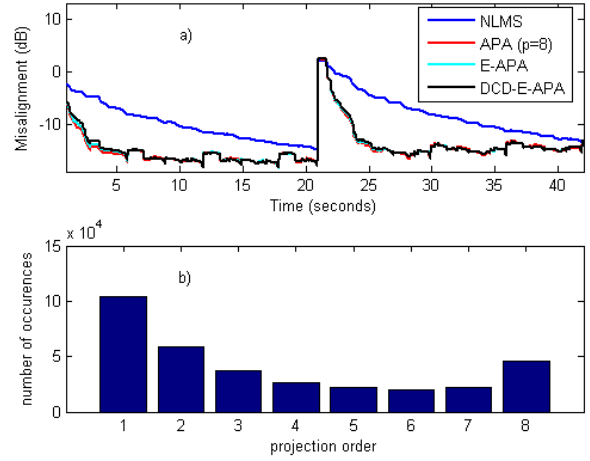


Fig. 7 a) Misalignment of NLMS, APA ($p=8$), E-APA, and DCD-E-APA. Single-talk case, under-modelling case, $L=512$; b) the number of occurrences of each projection order between 1 and 8 for DCD-E-APA.

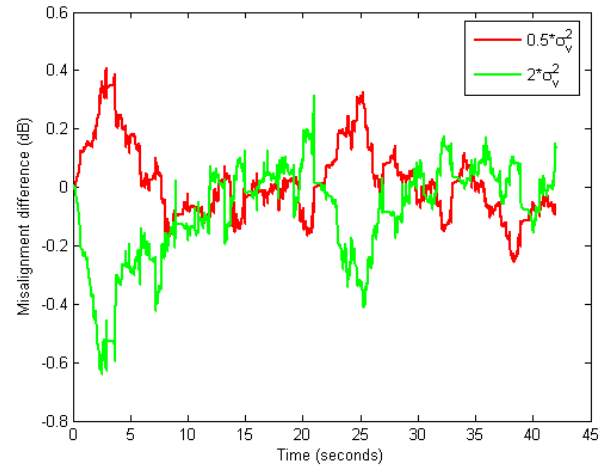


Fig. 8 Misalignment difference in dB between DCD-E-APA with exact noise estimation σ_v^2 and DCD-E-APA with $0.5\sigma_v^2$ and $2\sigma_v^2$ respectively (conditions of Fig. 7).

IV. CONCLUSIONS

The DCD-E-APA has been proposed for AEC. The dichotomous coordinate descent method was used and an efficient update of the variable size autocorrelation matrix has been proposed. It is shown that it provides both faster convergence and smaller steady state error at much reduced complexity than the APA. Future work will be focused into a derivation of a variable projection order version of an exact evolutionary APA using the displacement structure theory [12] and a comparison of our proposed algorithm with another algorithm, such as the EF-LSL of [2].

ACKNOWLEDGMENT

This work was supported by the UEFISCSU Romania under Grant PN-II-PCE-“Idei” no. 331/01.10.2007.

The authors wish to thank to the anonymous reviewers for their valuable comments.

REFERENCES

- [1] 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, pp. 19–27, May 1984.
- [2] F. Ling, D. Monolakis, and J. G. Proakis, "Numerically robust least squares lattice-ladder algorithms with direct updating of the reflection coefficients," *IEEE Trans. Acoust., Speech, Signal Process.*, vol. ASSP-34, pp. 837–845, Aug. 1986.
- [3] P. S. R. Diniz, "Adaptive filtering: Algorithms and practical implementations", *Springer*, Boston, MA, 3rd edition, 2008
- [4] S. L. Gay "A fast converging, low complexity adaptive filtering algorithm," in *Proc. IWAENC*, 1993, pp. 223–226.
- [5] F. Albu, J. Kadlec, N. Coleman, and A. Fagan, "The Gauss-Seidel fast affine projection algorithm," in *Proc. IEEE SIPS*, 2002, pp. 109–114.
- [6] Y. Zakharov, "Low complexity implementation of the affine projection algorithm," *IEEE Signal Processing Letters*, vol. 15, pp. 557–560, 2008.
- [7] Y. Zakharov and F. Albu, "Coordinate descent iterations in fast affine projection algorithm," *IEEE Signal Processing Letters*, vol. 12, pp. 353–356, May 2005.
- [8] S. -E. Kim, S. -J. Kong, and W. -J. Song, "An affine projection algorithm with evolving order," *IEEE Signal Processing Letters*, vol. 16, no. 11, Nov. 2009
- [9] H. -C. Shin, and A. H. Sayed, "Mean-square performance of a family of affine projection algorithms," *IEEE Transactions on Signal Processing*, vol. 52, no. 1, pp. 90–102, Jan. 2004
- [10] N. R. Yousef and A. H. Sayed, "A unified approach to the steady-state and tracking analysis of adaptive filters," *IEEE Signal Processing Letters*, vol. 11, pp. 132–135, Feb. 2004.
- [11] J. Benesty, H. Rey, L. R. Vega, and S. T essens, "A nonparametric VSS NLMS algorithm," *IEEE Signal Process. Lett.*, vol. 13, no. 10, pp. 581–584, Oct. 2006.
- [12] M. C. Tsakiris and P. A. Naylor, "Fast exact affine projection algorithm using displacement structure theory", in *Proc. of DSP 2009*, pp. 69–74, Santorini, Greece, July 2009

The codes for the proposed algorithms can be obtained from http://falbu.50webs.com/List_of_publications_aec.htm

The reference for the paper is:

F. Albu, C. Paleologu, J. Benesty, "Efficient Implementation of a Variable Projection Order Affine Projection Algorithm", *International Symposium on Wireless Communication Systems (ISWCS)*, York, U.K., Sept. 2010, pp. 369–373