Matrices Manipulation for the Implementation of a Hardwired Affine Projections Algorithm for Acoustic Echo Cancelling

[1]Aminadabe Soares , [1]Sérgio Almeida, [1]Eduardo Costa
amifractal@gmail.com, {smelo,ecosta}@ucpel.tche.br

[1]Catholic University of Pelotas

Abstract

The main goal of this work is the development of a dedicated hardware architecture for the adaptive algorithm Affine Projection (AP) for the acoustic echo cancellation. Among the main acoustic echo systems it can be mentioned: teleconferencing systems, voice-controlled systems, public systems of sound and hearing aids. The main motivation for using the AP algorithm is the high rate of convergence when compared to traditional Least Mean Square (LMS) and Normalized Least Mean Square (NLMS) algorithms. For the hardwired implementation of the AP algorithm, one of the main steps involves the proper manipulation of matrices. Thus, this work shows the steps for the implementation of the matrices with its variations (transpose and inversion).

1. Introduction

The least mean squares (LMS) and its normalized version (NLMS) are among the most often used algorithms in adaptive signal processing applications. However, their convergence speeds are insufficient when the input signals are highly correlated and the number of adaptive taps is large [1]. Acoustic echo cancellation is one important application with such characteristics. The Affine Projection (AP) algorithm was proposed by Ozeki and Umeda in 1984 [2] as a solution to this problem. The AP algorithm updates the weights in directions that are orthogonal to the last *P* input vectors. The whitening of the input signal allows an increase in the speed of convergence [3]. Thus, AP is a better choice than LMS or NLMS for applications with highly correlated input signals [4]. Higher computational complexity is the price of the faster convergence. This cost decreases as more advanced semiconductor elements are introduced. Complex algorithms have recently become feasible for applications such as echo cancellation, channel equalization and noise cancellation.

Although there exists a large number of papers applying the AP algorithm in acoustic echo cancellation in literature, whose implementations are made in boards based on DSP processors, to the best our knowledge, there exist no works that propose the implementation of this algorithm in a dedicated hardware structure.

This paper presents shows the faster convergence enabled by the AP algorithm, when compared with LMS and NLMS algorithms. Moreover, a preliminary study of matrices manipulation is also presented, since it represents the main step aiming the development of a dedicated architecture using adaptive AP algorithm.

2. Acoustic Echo

The problem of acoustic echo cancellation (AEC) was introduced in [5] and is still an active field of research. Acoustic Echo Cancellers are needed for removing the acoustic echoes resulting from the acoustic coupling between the loudspeaker(s) and the microphone(s) in communication systems.

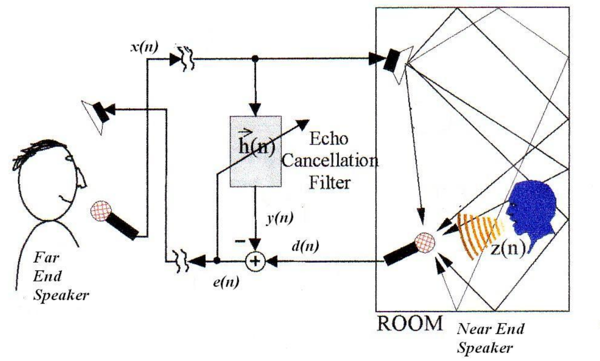

Fig. 1 – Typical AEC setup.

Acoustic echo occurs when an audio signal is reverberated in a real environment, resulting in the original intended signal plus attenuated, time delayed of this signal. In Fig.1, a typical setup for AEC is shown. The echo path is the ‘plant’ or ‘channel’ to be identified. The goal is to subtract a synthesized version of the echo from another signal (for example, picked up by a microphone) so that the resulting signal is ‘free of echo’ and really contains only the signal of interest [6]. This scheme applies to hands-free telephony inside a car, or teleconferencing in a conference room. The far end signal is fed into a loudspeaker (mounted on the dashboard, say, in the hands-free telephony application). The microphone picks up the near-end talker signal as well as an echoed version of the loudspeaker output, filtered by the room acoustics. The desired signal thus consists of the echo ('plant output') as well as the near-end talker signal.

The method used to cancel the echo signal is known as adaptive filtering. Adaptive filters are dynamic filters which iteratively alter their characteristics in order to achieve an optimal desired output. An adaptive filter alters its parameters in order to minimize a function of the difference between the desired output $d(n)$ and its actual output $y(n)$.

3. Adaptive Filter Structure

Different configurations are possible for the application of adaptive systems, which are: prediction, system identification, equalization and interference cancellation [7]. Fig.2 shows the configuration of an identification system using adaptive filtering. This class of application is discussed in this paper due to configure the basic structure of an adaptive system for acoustic echo cancellation.

The input signal $x(n)$ is applied to the plant W^o and the adaptive filter $w(n)$, resulting in the output $d(n)$ and $y(n)$, according to eq. (1) and (2).

$$\mathrm{d(n)} = \mathbf{u}^T(\mathrm{n})\mathbf{W}^o + \mathrm{r(n)} \quad (1)$$

$$\mathrm{y(n)} = \mathbf{u}^T(\mathrm{n})\mathbf{w}(\mathrm{n}) \quad (2)$$

Where $r(n)$ represents the noise of the plant, $x(n) = [x(n)\ x(n\text{-}1\ ...\ x(n\text{-}N\text{+}1)]^T$ and N is the length of the adaptive filter. The error signal is given by

$$\mathrm{e(n)} = \mathrm{d(n)} - \mathrm{y(n)} \quad (3)$$

The error signal, as will be seen below, is used as reference parameter for the adjustment of filter coefficients.

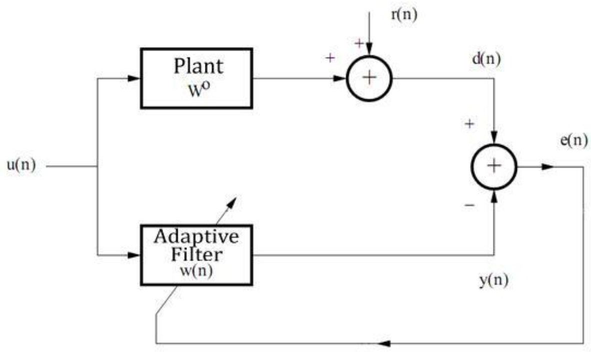

Fig. 2 – System identification.

4. Affine Projection Algorithm

As it is an algorithm with higher computational complexity than other algorithms mentioned, has not been studied as much as others. Nevertheless, due to increasing developments in the field of microelectronics, the AP algorithm has been widely used because it has become the solution for many applications where real-time response has been made possible by new digital signal processors (DSP), enabling new projects implementations and most modern hardware and software. The equation for adjustment of coefficients of the AP algorithm is given by

$$\mathbf{w}(\mathrm{n}+1) = \mathbf{w}(\mathrm{n}) + \alpha U(\mathrm{n})[U^T(\mathrm{n})U(\mathrm{n})]^{-1}\mathbf{e}(\mathrm{n}) \quad (4)$$

where α is the step of adaptation of the algorithm, $\mathbf{e}(n) = [\mathrm{e}(n)\ \mathrm{e}(n\text{-}1)\ ...\ \mathrm{e}(n\text{-}P\text{+}1)]^T$, $\mathbf{U}(n) = [\mathbf{u}(n\text{-}1)\ \ \mathbf{u}(n\text{-}2)\ \dots\ \mathbf{u}(n\text{-}P)]^T$ is a collection of P past input vectors $\mathbf{u}(n) = [\mathrm{u}(n)\ \mathrm{u}(n\text{-}1)\ ...\ \mathrm{u}(n\text{-}N\text{+}1)]^T$. In practical applications, the use of $P = 3$ allows to obtain better performance compared to other algorithms, with a small increase in computational complexity .

The structure used in this proposal to the adaptive filter is the transversal structure shown in Fig. 3. The number of delay elements, z^{-1}, determines the finite duration of its impulse response, ie the order of the adaptive filter.

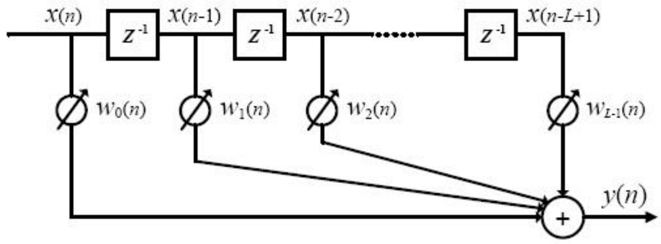

Fig. 3 – Single-input adaptive transversal filter.

Figure 4 shows the graph of the mean square error, defined by equation (5) (E{.} represents the expected value), of LMS, NLMS and AP algorithms. We can observe that for the same input signal correlated, the AP algorithm has a faster convergence than the others. However, the error in steady-state more than others. In applications where the speed of convergence is predominant, such as the acoustic echo cancellation, the algorithm AP appears to be an excellent choice.

$$E\{e^2(n)\} = E\{(d(n) - y(n))^2\} \tag{5}$$

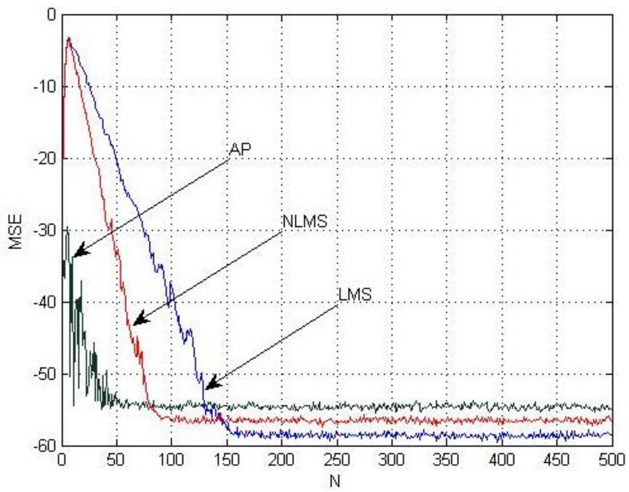

Fig. 4 – MSE behavior of the LMS, NLMS and AP algorithms.

5. Steps for the Matrices Manipulation

One of the main steps for the implementation of the AP algorithm is the manipulation of the inversion matrix defined by the expression $M = [U^T(n)U(n)]^{-1}$ as previously shown in equation (4). Initially, the matrix U(n) is specified with the dimension 16x3. After that, the transpose of U(n) is calculated. The results from the matrices are multiplied and the results are inverted. Since the matrix U(n) has a dimension of an array 16x3, thus, when it is multiplied by its transpose (3x16 array dimension), one obtains a matrix with dimensions 3x3 square, which can be obtained by the algebraic expression presented in equation (6), where $\mathbf{1 \le \boldsymbol{i} \le m}$ and $\mathbf{1 \le \boldsymbol{j} \le q}$.

$$p_{ij} = \sum_{k=1}^{p} a_{ik} b_{kj} \tag{6}$$

The inversion of the matrix is based on equation (7), where det(M) is the determinant of M and adj(M) is the matrix of cofactors of M.

$$M^{-1} = \frac{adj(M)}{det(M)} \tag{7}$$

Assuming that the matrix M is given by:

$$M = U^T(n)U(n) = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix}$$

Thus the matrix of cofactors is given by:

$$adj(M) = \begin{bmatrix} \begin{vmatrix} a & f \\ h & i \end{vmatrix} & -\begin{vmatrix} b & c \\ h & i \end{vmatrix} & \begin{vmatrix} b & c \\ a & f \end{vmatrix} \\ -\begin{vmatrix} d & f \\ g & i \end{vmatrix} & \begin{vmatrix} a & c \\ g & i \end{vmatrix} & -\begin{vmatrix} a & c \\ d & f \end{vmatrix} \\ \begin{vmatrix} d & e \\ g & h \end{vmatrix} & -\begin{vmatrix} a & b \\ g & h \end{vmatrix} & \begin{vmatrix} a & b \\ d & e \end{vmatrix} \end{bmatrix}$$

To complete the calculation of M-1 is it necessary to divide each element of the matrix of cofactors (adj(M)) by the determinant of the matrix M, according to equation (7).

6. Conclusions

This paper presented an initial study from matrices manipulation aiming a hardwired implementation of Affine Projection algorithm for Echo Acoustic Cancelling. As could be observed, the main aspect of this algorithm is the faster convergence when compared to traditional LMS and NLMS algorithms. This aspect motivates us to implement a hardwired version of this algorithm, since it can be used as part of portable equipments. As one of the main steps of the implementation of the AP algorithm involves the suitable manipulation of matrices, we presented an initial study by showing the main steps of matrices manipulation with its variations (transpose and inversion). We have implemented the presented matrices by using Hardware Description Language – VHDL. However, we have omitted preliminary results in this paper because they have been thoroughly tested and compared with MATLAB results. Thus, as future work, we hope show solid results from these implementations.

7. References

[1] S. Haykin, “Adaptive Filter Theory”. Prentice-Hall, 4 Ed. 4, 2002.

[2] K. Ozeki and T. Umeda, “An adaptive filtering algorithm using orthogonal projection to an affine subspace and its properties”. *Electron. Commun. Jpn.*, vol 67-A, no.5, pp. 19-27, Feb. 1984.

[3] M. Rupp, “A family of filter algorithms with decorrelating properties”, *IEEE Transactions on Signal Processing*, vol. 46, no. 3, pp. 771–775, Mar. 1998.

[4] SANKARAN, S. G.; A.A. BEEX. “Convergence behavior of affine projections algorithms”, *IEEE Trans. Signal Processing*, vol. 48, pp. 1086-1096. 2000.

[5] M. M. Sondhi, “An Echo Canceller”, Bell Syst. Tech. J. vol. XLVI, no. 3, pp. 497 – 510, 1967.

[6] BREINING, C. et al., “Acoustic Echo Control”. *Signal Processing Magazine*, vol. 16, no.4, pp.42-69, 1999.

[7] WIDROW, B., STEARNS, S. D., “Adaptive Signal Processing”. Englewood Cliffs, NJ: Prentice-Hall. 1985.