

Analysis of Adaptive Filtering Algorithms for Acoustic Echo Cancellation

Himavanth Reddy
ECE Department
IIIT-DM Kancheepuram, India - 600127
esd15i014@iiitdm.ac.in

Abstract—This paper presents an attempt to apply an adaptive filtering algorithm in order to overcome the echo generated in acoustic systems. The paper also presents a comparison between multiple adaptive algorithms in their ability to attenuate the echo in the system

Index Terms—Acoustic Echo Cancellation, Adaptive Algorithms, FDAF, LMS, Echo Return Loss Cancellation, Fast Fourier Transform (FFT)

I. INTRODUCTION

Acoustic echo cancellation (AEC) is generally necessary in full-duplex communication scenarios where loudspeaker echoes should be removed from a microphone signal. This is necessary for teleconferences where the microphone signal is sent to far-end communication partners who may be disturbed when hearing their own voices. Adaptive filters designed for acoustic echo cancellation schemes are confronted with several difficulties. First, the number of taps needed to model the impulse response of an acoustic echo path shall be very large. Moreover, non stationary signals such as speech signals are highly correlated also the signal picked up by the microphone is corrupted by the near-end speech, and by the ambient noise. Dealing with long impulse responses, and designing adaptive filtering algorithms for highly correlated speech signals, has led to many time domain and frequency domain algorithms developed. Here we have simulated few adaptive filtering algorithms and compared them from the point of view of the performance parameter i.e. echo return loss enhancement, by varying step size.

II. OBJECTIVES OF THIS PROJECT

- 1) Apply Frequency Domain Adaptive Filtering (FDAF) to eliminate echo in the system
- 2) Compare the performance of FDAF with the well known Least Means Square (LMS) algorithm

III. ALGORITHMS

A. LMS Algorithm

This algorithm adapts to a solution of minimizing mean-square error. This method is based on steepest-descent method. In this, the gradient of mean-squared error is found out with respect to h .

If $w(n)$ is the weight vector and $x(n)$ is the input signal then, output $y(n)$ of the adaptive filter is given by

$$y(n) = w(n)^T x(n),$$

The error signal $e(n)$ is given by

$$e(n) = d(n) - y(n),$$

The weight update equation is given by

$$w(n+1) = w(n) + \mu e(n)x(n)$$

where μ is the step size which controls the convergence rate. If the value of μ is small, then the convergence time is more. So the selection of suitable value of step size is very important. This algorithm is very simple and only requires few numbers of additions and multiplications per iteration for an N -tap filter. It has low computational complexity and the problem of double-talk is removed. But the step size chosen during every iteration in this method is of fixed size. The computational complexity of this algorithm involves $2M$ additions and $2M + 1$ multiplications.

B. Frequency Domain Adaptive Filtering Algorithm

In acoustic echo cancellation the impulse response of the system often reaches over 100ms in length. This would require an adaptive FIR filter with over 1000 coefficients. The linear convolution and the update of the adaptive filter with this length creates a significant computational burden for applications that require low power processors. Thus FFT is used to minimize the number of calculations required and therefore make the algorithm more efficient.

Let L be the length of the block and M be the length of tapped weight vector.

$$A_0 = \begin{bmatrix} u(kL) \\ u(kL+1) \\ \dots \\ u(kL+L-1) \end{bmatrix}, A_1 = \begin{bmatrix} u(kL-1) \\ u(kL) \\ \dots \\ u(kL+L-2) \end{bmatrix}$$

The data matrix can then be written as

$$\mathbf{A}(k) = \begin{bmatrix} \mathbf{A}_0 & \mathbf{A}_1 & \mathbf{A}_2 & \dots & \mathbf{A}_{M-1} \end{bmatrix} \\ = \begin{bmatrix} \mathbf{G}(kM) \\ \mathbf{G}(kM+1) \\ \dots \\ \mathbf{G}(kM+L-1) \end{bmatrix}$$

The matrix $\mathbf{A}(k)$ is an $L \times M$ matrix and length of the vector $\mathbf{G}^T(kM)$ is M . Let the weight vector be

$$\hat{\mathbf{W}}(k) = [w_0(k) \quad w_1(k) \quad w_2(k) \dots w_{L-1}(k)]^T$$

Output of the filter is

$$\begin{bmatrix} y(kL) & y(kL+1) & y(kL+2) \dots y(kL+L-1) \end{bmatrix}^T \\ = \mathbf{A}(k) \hat{\mathbf{W}}(k)$$

For individual element,

$$y(kL) = \mathbf{G}(kM) \hat{\mathbf{W}}(k)$$

$$\begin{aligned}
y(kL+1) &= G(kM+1)\hat{W}(k) \\
&\dots\dots\dots \\
y(kL+i) &= G(kM+i)\hat{W}(k) \\
&= [w_0(k) \quad w_1(k) \quad w_2(k) \dots \dots \dots w_{L-1}(k)] * \\
&\quad \begin{bmatrix} u(kL+i) \\ u(kL+i-1) \\ u(kL+i-2) \\ \vdots \\ u(kL+i+M-1) \end{bmatrix} \\
&= \sum_{j=0}^{M-1} w_j(k) \cdot u(kL+i-j)
\end{aligned}$$

Let the desired response of $(kL+i)$ -th element is $d(kL+i)$.
Therefore, the error signal is

$$e(kL+i) = d(kL+i) - y(kL+i)$$

$$\begin{aligned}
&\text{In matrix form,} \\
&\begin{bmatrix} d(kL) \\ d(kL+1) \\ d(kL+2) \\ \dots \\ \dots \\ \dots \\ d(kL+L-1) \\ e(kL) \\ e(kL+1) \\ e(kL+2) \\ \dots \\ \dots \\ \dots \\ e(kL+L-1) \end{bmatrix} - \begin{bmatrix} y(kL) \\ y(kL+1) \\ y(kL+2) \\ \dots \\ \dots \\ \dots \\ y(kL+L-1) \end{bmatrix} = \mathbf{e}(\mathbf{k})
\end{aligned}$$

The cross correlation vector is

$$\Phi(k) = \mathbf{A}^T(k)\mathbf{e}(k)$$

The update equation of the weight vector can be written as

$$\hat{\mathbf{W}}(k+1) = \hat{\mathbf{W}}(k) + \mu\Phi(k)$$

where μ is the step-size parameter.

Now using $\hat{\mathbf{W}}(k+1)$ we have

$$\begin{aligned}
&[y(kL+L)y(kL+L+1)y(kL+L+2)\dots y(kL+2L-1)]^T \\
&= \mathbf{A}(k+1) \cdot \hat{\mathbf{W}}(k+1)
\end{aligned}$$

The updated error matrix is

$$\begin{aligned}
&\begin{bmatrix} d(kL+L) \\ d(kL+L+1) \\ d(kL+L+2) \\ \dots \\ \dots \\ \dots \\ d(kL+2L-1) \\ e(kL+L) \\ e(kL+L+1) \\ e(kL+L+2) \\ \dots \\ \dots \\ \dots \\ e(kL+2L-1) \end{bmatrix} - \begin{bmatrix} y(kL+L) \\ y(kL+L+1) \\ y(kL+L+2) \\ \dots \\ \dots \\ \dots \\ y(kL+2L-1) \end{bmatrix} \\
&= \mathbf{e}(\mathbf{k}+1)
\end{aligned}$$

The updated cross correlation vector and the weight vector are respectively as

$$\Phi(k+1) = \mathbf{A}^T(k+1)\mathbf{e}(k+1)$$

and

$$\hat{\mathbf{W}}(k+2) = \hat{\mathbf{W}}(k+1) + \mu\Phi(k+1)$$

Similarly,

$$\begin{aligned}
&[y(kL+iL)y(kL+iL+1)y(kL+iL+2)\dots \\
&y(kL+(i+1)L-1)]^T = \mathbf{A}(k+i) \cdot \hat{\mathbf{w}}(k+i)
\end{aligned}$$

$$\begin{aligned}
&\text{Let} \\
&\begin{bmatrix} d(kL+iL) \\ d(kL+iL+1) \\ d(kL+iL+2) \\ \dots \\ \dots \\ \dots \\ d(kL+(i+1)L-1) \\ e(kL+iL) \\ e(kL+iL+1) \\ e(kL+iL+2) \\ \dots \\ \dots \\ \dots \\ e(kL+(i+1)L-1) \end{bmatrix} - \begin{bmatrix} y(kL+iL) \\ y(kL+iL+1) \\ y(kL+iL+2) \\ \dots \\ \dots \\ \dots \\ y(kL+(i+1)L-1) \end{bmatrix} = \\
&= \mathbf{e}(\mathbf{k}+i)
\end{aligned}$$

In generalized form,

$$\Phi(k+i) = \mathbf{A}^T(k+i)\mathbf{e}(k+i)$$

and

$$\hat{\mathbf{W}}(k+i+1) = \hat{\mathbf{W}}(k+i) + \mu\Phi(k+i)$$

In overlap-save method $N = 2M$ point FFT is used and the size of the filter of M tap weights.

Input signal in n domain is express as, $[(k-1)th \text{ block}, kth \text{ block}]$, where the $(k-1)th$ block is

$$\begin{bmatrix} u(kL-L) & u(kL-L-1) & \dots & u(kL-2M+1) \\ u(kL-L+1) & u(kL-L) & \dots & u(kL-L-2M+2) \\ \ddots & \ddots & \ddots & \ddots \\ u(kL-1) & u(kL-2) & \dots & u(kL-2M) \end{bmatrix}$$

And the k-th block is

$$\begin{bmatrix} u(kL) & u(kL-1) & \dots & u(kL-2M+1) \\ u(kL+1) & u(kL) & \dots & u(kL-2M+2) \\ \ddots & \ddots & \ddots & \ddots \\ u(kL+M-1) & u(kL+M-2) & \dots & u(kL-M) \end{bmatrix}$$

Input signal in k domain,

$$\mathbf{U}(k) = \text{diag}(FFT[(k-1)th \text{ block}, kth \text{ block}])$$

an $N * N$ matrix. The initial weight in k domain is a $1 * N$ vector:

$$\hat{\mathbf{W}}(k) = FFT \begin{bmatrix} \hat{\mathbf{w}}(k) \\ \mathbf{O} \end{bmatrix}$$

Here $\hat{\mathbf{w}}(k)$ is tap-weight vector (in n domain) of length M and \mathbf{O} is a null vector of length M .

Output signal of the filter in k domain is

$\hat{\mathbf{y}}(k) = \text{Last } M \text{ elements of } \text{IFFT}[U(k)\hat{\mathbf{W}}(k)]$

The desired response vector,

$$\mathbf{d}(k) = [d(kM), d(kM+1), \dots, d(kM+M-1)]^T$$

Error signal vector in n domain,

$$\mathbf{e}(k) = [e(kM), e(kM+1), \dots, e(kM+M-1)]^T$$

$$= \mathbf{d}(k) - \mathbf{y}(k)$$

The error signal in the frequency domain is

$$\mathbf{E}(k) = \text{FFT} \begin{bmatrix} \mathbf{O} \\ \mathbf{e}(k) \end{bmatrix}$$

The cross-correlation vector is

$$\phi(k) = \text{First } M \text{ elements of } \text{IFFT}[U^H(k)E(k)]$$

Thus, the updated tap-weights can be written as,

$$\hat{\mathbf{W}}(k+1) = \hat{\mathbf{W}}(k) + \mu \text{FFT} \begin{bmatrix} \phi(k) \\ \mathbf{O} \end{bmatrix}$$

FDAF provide several advantages over its time domain counterpart. Besides being able to perform the filter convolution by a multiplication in frequency domain, also the length of the adaptive filter are effectively decimated by the transformation. Thus, the computational complexity of the adaptive algorithm is reduced. In addition, to reduced computational complexity, FDAF can also provide an increased convergence speed. This is a result from the decreased eigenvalue spread of the auto-correlation matrix of the signals in the filter update.

These advantages of the FDAF ultimately come with a tradeoff. The main costs of FDAF are increased latency and an increased memory requirement. The latency cost comes from the need to delay the desired signal (or microphone signal in echo cancellation) by the delay through the frequency domain filter. This results in an increased memory storage over time domain approaches because both the excitation and desired signals need to be stored.

The total computational complexity of FDAF is given by: $10M \log(M) + 26M$

IV. IMPLEMENTATION

The entire system is simulated in MATLAB with the help of the DSP toolbox. Initially, a room impulse response is generated which acts as the echo path. microphone which is called as near end speech signal. A voice travels out the loudspeaker, bounces around in the room, and this voice is picked up by the system's microphone, this voice signal is called as far end speech signal. The microphone signal contains both the near end and far end speech signals and some additional noise. The adaptive algorithm is then applied on the microphone signal in order to attenuate the noise and obtain the original near end speech signal. The adaptive filters are chosen to be of length 2048 and the input signal is sampled at a rate of 16000Hz

V. PERFORMANCE PARAMETERS

The performance parameter which is used here for the performance evaluation of different AEC Algorithms is

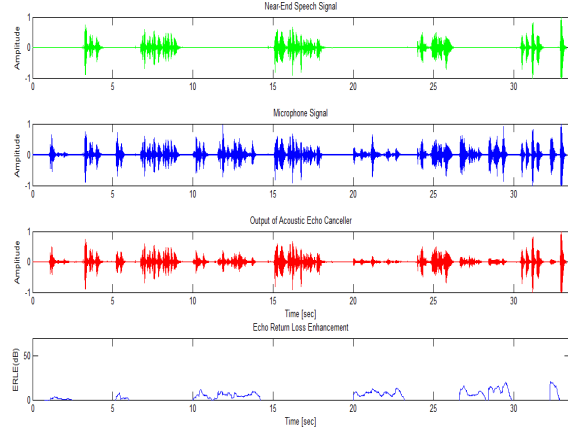


Fig. 1. Results with LMS algorithm, Step size = 0.05

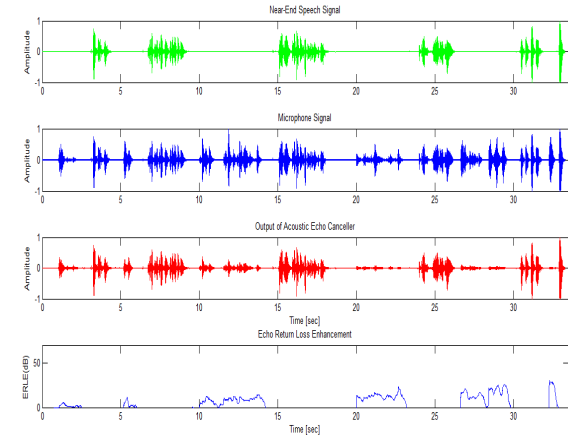


Fig. 2. Results with LMS algorithm, Step size = 0.13

known as Echo Return Loss Enhancement (ERLE).

It is a smoothed measure of the amount (in dB) that the echo has been attenuated. It is mathematically defined as

$$\text{ERLE(dB)} = 10 \log_{10} \frac{d^2(n)}{e^2(n)}$$

Where,

$d(n)$ is the far-end echoed signal

$e(n)$ is the residual echo after cancellation

VI. RESULTS

The results of LMS and FDAF are compared, each with 2 different values of step sizes

VII. CONCLUSION

Based on the simulation results we can see that the FDAF algorithm provides results that are significantly better than its LMS counterpart. This is primarily because of the long impulse responses and high number of filter coefficients of the system which are effectively handled by FDAF but at the

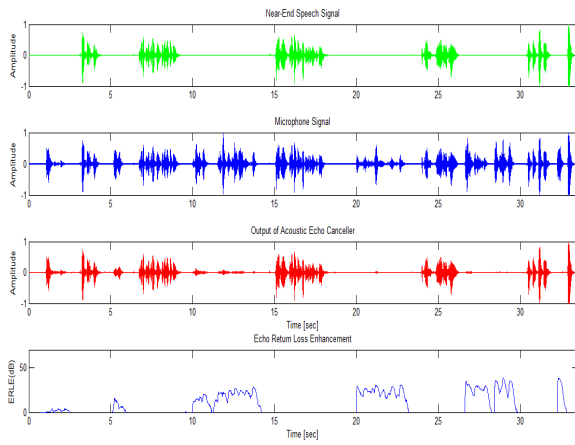


Fig. 3. Results with FDAF algorithm, Step size = 0.025

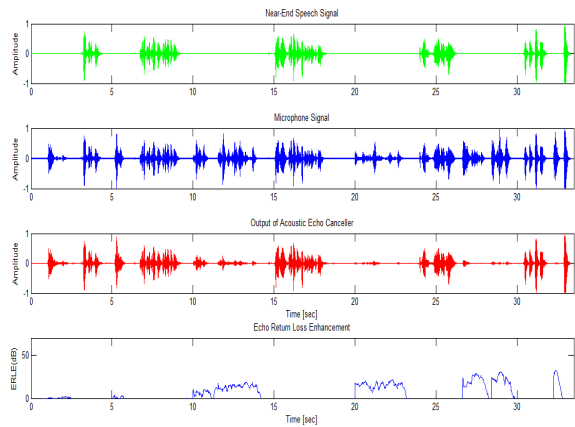


Fig. 4. Results with FDAF algorithm, Step size = 0.05

cost of latency and memory requirement.

Algorithms such as Partition Block FDAF are known to show better performance compared to FDAF in terms of Echo Return Loss Enhancement

REFERENCES

- [1] Paleologu, Constantin and Ciochină, Silviu and Benesty, Jacob and Grant, Steven L. *An overview on optimized NLMS algorithms for acoustic echo cancellation*. EURASIP Journal on Advances in Signal Processing 2015. Nov 19, 2015
- [2] Simon Haykin. *Adaptive Filter Theory*, 4th edition. Upper Saddle River, NJ: Prentice-Hall 2002
- [3] AP Patil and MR Patil *Performance analysis of Adaptive Filtering Algorithms for Acoustic Echo Cancellation*. International Journal of Engineering Research and Technology, Vol.7, Issue.08, Aug 2018.
- [4] Kamarudin, Noraziahtulhidayu and Al-Haddad, S. A. R. and Abushariah, Mohammad A. M. and Hashim, Shaiful Jahari and Hassan, Abd Rauf Bin *Acoustic echo cancellation using adaptive filtering algorithms for Quranic accents (Qiraat) identification*. International Journal of Speech Technology, Vol.19, No.2, Jun 2016, Pages 393-405