International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 9 Issue 08, August-2020

Adaptive Noise Cancellation using Least Mean Sqaure Filter Algorithm (Matlab)

Archit Prakash Raut[1] , Aayesha Raj[2] , Shalvi Patil[3] and Tanvi Patil[4]
Department of Electronics and Telecommunication Engineering
Vidyavardhini's College of Engineering and Technology
Affiliated to the University of Mumbai.

Abstract **:- Adaptive filtering is a wide area of researcher in present decade in the field of communication. Adaptive noise cancellation is an approach used for noise reduction in speech signal. As received signal is continuously corrupted by noise where both received signal and noise signal both changes continuously, then this arise the need of adaptive filtering. This paper deals with cancellation of noise on speech signal using two adaptive algorithms Least Mean Square (LMS) algorithm and Normalized Least Mean Square (NLMS) Algorithm. Choose the algorithms that provide efficient performance with less computational complexity.**

Keywords :- Adaptive noise cancellation (ANC), Least Mean square (LMS) Algorithm, Adaptive filtering and Normalized Least Mean Square (NLMS).

I. INTRODUCTION

Acoustic noise problems becomes more pronounce as increase in number of industrial equipment such as engines, transformers, compressors and blowers are in use. The traditional approach to acoustic noise cancellation uses passive techniques such as enclosures, barriers and silencers to remove the unwanted noise signal [1][2]. Silencers are important for noise cancellation over broad frequency range but ineffective and costly at low frequencies. Mechanical vibration is a type of noise that creates problems in all areas of communication and electronic appliances. Signals are carriers of information, both useful and unwanted. Extracting or enhancing the useful information from a mix of conflicting information is a simplest form of signal processing. Signal processing is an operation designed for extracting, enhancing, storing, and transmitting useful information. Hence signal processing tends to be application dependent. In contrast to the conventional filter design techniques, adaptive filters do not have constant filter coefficients and no priori information is known. Such a filter with adjustable parameters is called an adaptive filter. Adaptive filter adjust their coefficients to minimize an error signal and can be realized as finite impulse response (FIR), infinite impulse response (IIR), lattice and transform domain filter [4]. The most common form of adaptive filter is the transversal filter using Least Mean Square (LMS) algorithm and Normalized Least Mean Square (NLMS) algorithm. In this paper, noise is defined as any kind of undesirable signal, whether it is borne by electrical, acoustic, vibration or any other kind of media. In this paper, adaptive algorithms are applied to different kind of noise. The basic idea of an adaptive noise cancellation algorithm is to pass the corrupted signal through a filter that tends to suppress the noise while leaving the signal unchanged. This is an adaptive process, which means it does not require a priori knowledge of signal or noise characteristics. Adaptive noise cancellation (ANC) efficiently attenuates low frequency noise for which passive methods are ineffective. Although both FIR and IIR filters can be used for adaptive filtering, the FIR filter is by far the most practical and widely used. The reason being that FIR has adjustable zeros, and hence it is free of stability problems associated with adaptive IIR filters that have adjustable poles as well as zeros.

II. LEAST MEAN SQUARE ALGORITHM

To make exact measurements of the gradient vector ∇J(n) at each iteration n, and if the step-size parameter is suitably chosen then the tap-weight vector computed by using the steepest descent algorithm would converge to the optimum wiener solution. The exact measurements of the gradient vector are not possible and since that would require prior knowledge of both the auto-correlation matrix R of the tap inputs and the cross correlation vector p between the tap inputs and the desired response, the optimum wiener solution could not be reached [3]. Consequently, the gradient vector must be estimated from the available data when we operate in an unknown environment. After estimating the gradient vector we get a relation by which we can update the tap weight vector recursively as:

$$w(n + 1) = w(n) + \mu u(n)[d^*(n) - u\ H(n)w(n)] \qquad (1)$$

Where μ= step size parameter
$u^H(n)$ = Hermit of a matrix **u**
$d^*(n)$ = Complex conjugate of d(n)

We may write the result in the form of three basic relations as follows:

1.Filter output:

$$y(n) = wH(n)u(n) \qquad (2)$$

2.Estimation error or error signal:

$$e(n) = d(n) – y(n) \qquad (3)$$

3.Tap weight adaptation:

$$w(n+1) = w(n) + \mu u(n)e^*(n) \qquad (4)$$

Equations (2) and (3) define the estimation error e(n), the computation of which is based on the current estimate of the tap weight vector w(n). Note that the second term, u(n)e*(n) on the right hand side of equation (4) represents the adjustments that are applied to the current estimate of the tap weight vector w(n). The iterative procedure is started with an initial guess w(0).The algorithm described by equations (2) and (3) is the complex form of the adaptive least mean square (LMS) algorithm. At each iteration or time update, this algorithm requires knowledge of the most recent values u(n), d(n)w(n). The LMS algorithm is a member of the family of stochastic gradient algorithms. In particular, when the LMS algorithm operates on stochastic inputs, the allowed set of directions along which we “step” from one iteration to the next is quite random and therefore cannot be thought of as consisting of true gradient directions.

III. NOISE CANCELLATION

Fig. 1 shows the basic problem and the adaptive noise cancelling solution to it. A signal s is transmitted over a channel to a sensor that also receives a noise n0 uncorrelated with the signal. The primary input to the canceller is combination of both signal and noise s + n0 . A second sensor receives a noise n1 uncorrelated with the signal but correlated with the noise n0. This sensor provides the reference input to the canceller. This noise n1 is filtered to produce an output y that is as close a replica of n0. This output of adaptive filter is subtracted from the primary input s + n0 to produce the system output z = s n0 –y.

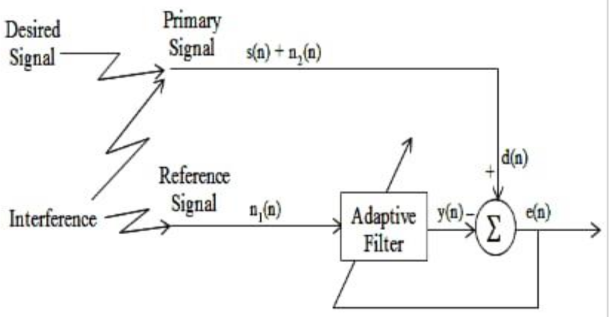

Fig 1. Adaptive noise cancellation concept.

If we know the characteristics of the channels over which the noise and signal was transmitted to the primary and reference sensors, it would theoretically be possible to design a fixed filter. The filter output could then be subtracted from the primary input, and the system output would be signal alone. But the characteristics of the transmission paths are unknown and are not of a fixed nature, due to this use of a fixed filter are not feasible.

In Fig. 1 the reference input is processed by an adaptive filter. An adaptive filter is that which automatically adjusts its own impulse response. Adjustment is accomplished through an algorithm. The filter can operate under changing conditions and can readjust itself continuously to minimize the error signal. In noise cancelling systems the practical objective is to produce a system output z = s + n 0 - y that is a best fit in the least squares sense to the signal s. This objective is accomplished by feeding the system output back to the adaptive filter and adjusting the filter through an LMS adaptive algorithm to minimize total system output power. In an adaptive noise cancelling system, the system output serves as the error signal for the adaptive process.

The prior knowledge of the signal s or of the noises n0 and n1 would be necessary before the filter could be designed to produce the noise cancelling signal y. Assume that s, n1 , n0 and y are statistically stationary and have zero means. Assume that s is uncorrelated with n0 and n1 , and suppose that n1 is correlated with n0 .The output z is

$$z = s + n0 - y \qquad (1)$$

Squaring, we obtain

$$z2 = s2 +(n0 - y)2 + 2s(n0 - y) \qquad (2)$$

Taking expectations both side of equation (2)

$$E[z2] = E[s2] + E[(n0 - y)2] + 2E[s(n0 - y)]$$

Realizing that s is uncorrelated with n0 that output y is

$$E[z2] = E[s2] + E[(n0 - y)2] \qquad (3)$$

The signal power E[s2] will be unaffected as the filter is adjusted to minimize E[z2]. Accordingly, the minimum output power is min E[z2] = E[s2] +min E[(n0 - y)2] (4)

When the filter is adjusted so that E[z2] is minimized, therefore E[(n0 - y)2] is, also minimized. The filter output y is then a best least squares estimate of the primary noise n0. Moreover, when E[(n0 - y)2] is minimized, E[(z-s)2] is also minimized, since, from (1),

$$z - s = n0 – y \qquad (5)$$

Adapting the filter to minimize the total output power is thus causing the output z to be a best least squares estimate of the signal s. The output z will contain the signal s plus noise. From (l), the output noise is given by (n0 - y). Since minimizing E[z2] minimizes E[(n0 - y)2] minimizing the total output power minimizes the output noise power. Since the signal in the output remains constant, minimizing the total output power maximizes the output signal to noise ratio. From (3) the smallest possible output power is

$$E[z2] = E[s2] \qquad (6)$$

When E[(n0 - y)2] = 0

At y=n0 and z=s .

Minimizing the output power causes the output signal to be perfectly noise free.

IV. EXPERIMENTAL RESULTS

In this section we compare the performance of the LMS and NLMS algorithms as noise canceller. The algorithms are implemented according to the steps. Figure 3 shows that, the Input sinusoidal signal and random noise signal. Figure 4 shows that, the noise present in the sinusoidal signal and is eliminated using LMS algorithm.

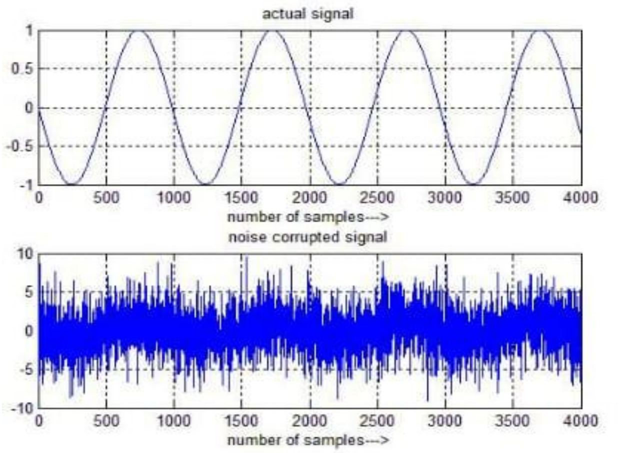

Fig 3. Input and noise signal.

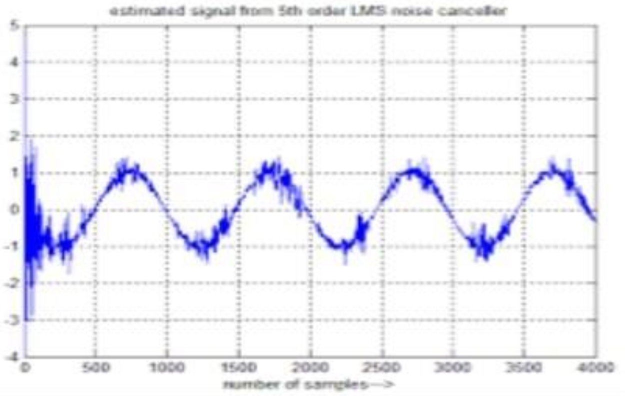

Fig 4. LMS filter output

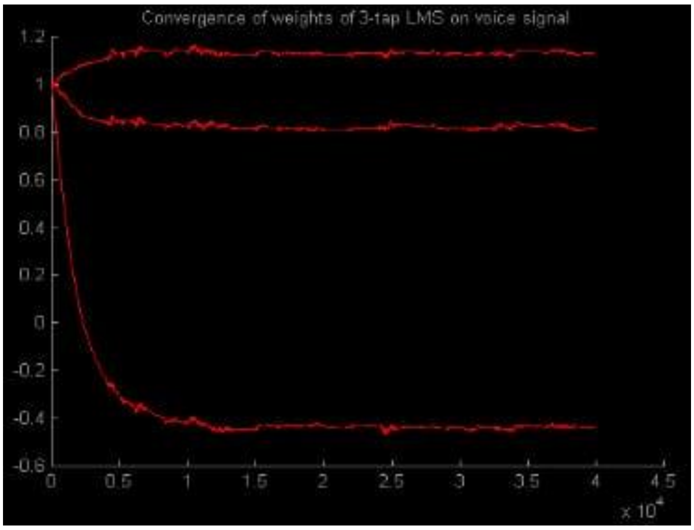

Fig 5. Convergence of weights

V. CONCLUSION.

This paper has described an application in which the use of an LMS adaptive filter is particularly appropriate. The main goal of this paper is to investigate the application of an algorithm based on adaptive filtering in noise cancellation problem. The LMS algorithm has been shown to produce good results in a noise cancellation problem.

V1. ACKNOWLEDGEMENT.

I take this opportunity to express my regards and sincere thanks to my advisor and guide Prof. Ashish Vanmali, without whose support, this project would not have been possible. His constant encouragement and moral support gave me the motivation to carry out the project successfully. I am also indebted to him for his valuable and timely guidance. The discussions with him helped a lot in developing an in-depth understanding of the topics involved. Also, I would like to thank the Lab In charge, who helped me with the lab facilities whenever I needed them.

VII. REFERENCE

[1] Adaptive Filter Theory by Simen Haykin: 3rd edition,Pearson Education Asia.LPE.

[2] Adaptive Signal Processing by John G Proakis, 3rd edition, Perntice Hall of India.

[3] B. Widow,"Adaptive noise canceling: principles and applications", Proceedings of the IEEE, vol. 63, pp.1692-1716, 1975.

[4] Adaptive Signal Processing by Bernard Widrow and Samuel D.Stearns; Pearson Education Asia, LPE.

APPENDIX

MATLAB CODE

```
clc;
close all;
clear all;
order=3;

NN=10000;

[y1,Fs]=wavread('a1');
y = (y1(1:NN))';
time=(1/22050)*200;
N=length(y);
xlabel('time (sec)');
ylabel('relative signal strength')
subplot(5,1,1); plot(y);

n1 = randn(1,NN);
n1 = n1/max(abs(n1));
subplot(5,1,2); plot(n1);

x = y + 0.1*n1;
subplot(5,1,3); plot(x);

ref=x+.1*rand; %noisy noise
subplot(5,1,4)
```

```
plot(ref)
title('reference  (noisy noise)   (input2)');
wall = [];
w=0.1*zeros(order,1);
mu=0.04;
for i=1:N-order
   buffer = ref(i:i+order-1); %current 32 points of reference
   desired(i) = x(i)-buffer*w;                    %dot product reference and coeffs
   w=w+(buffer.*mu*desired(i)/norm(buffer))';     %update coeffs
   wall = [wall,w];
end

subplot(5,1,5)
plot(desired)
title('Adaptive output  (hopefully it''s close to "voice")')

color = ['b','r','m','k','g','c','y','b','r','m','k','g','c','y'];
figure;
for pp = 1 : order
    plot(wall(pp,:),color(pp));
    hold on;
end
```