Lab Homework: Adaptive Noise Cancellation
Digital Signal Processing ELEC 6601

Jamil Reza Chowdhury
Electrical and Computer Engineering.
Concordia Unversity

Montreal Canada

j_chowd@encs.concordia.ca

Abstract— we aim at cancelling the additive noise in a noisy audio signal, namely noise reduction, by means of adaptive filtering techniques. Here we combined the audio samples of a speech that is original signal with the real noise to generate noisy signal. We recorded different speech signals and added noisy signal with different amounts of signal-to-noise (SNR) ranging from -10dB to +10dB directly in MATLAB. Finally, we used LMS based noise cancelling techniques on the input noisy signal to obtain the de-noised enhanced signal as the output. The noise signal generated directly from MATLAB was used as the reference signal to calculate the LMS filter weights. The de-noised signal had the same length as the desired signals.

Keywords—adaptive filter, noise cancellation, least mean square error, variance, signal to noise ratio

I. Introduction

Noise-cancelling or noise removal are techniques in signal processing that perform the removal of certain undesired audible frequencies. If the undesired frequencies are removed, we call it noise removal and if they are suppressed, we call it noise cancellation. Adaptive Noise Cancellation is a variation of optimal filtering that involves producing an estimate of the noise by filtering the reference input and then subtracting this noise estimate from the primary input containing both signal and noise. Active noise cancellation (ANC) that employs a process of using a microphone to monitor environmental noise and creating anti-noise that is then mixed in with audio playback to cancel noise entering the user's ears. So, we aim at cancelling the noise in a noisy signal by means of adaptive filtering techniques. Filters may be used for filtering, smoothing, prediction. Adaptive filters are self-designing using a recursive algorithm. An adaptive filter will change its coefficients based on a given criteria, to improve its performance. For example, the least Mean Square (LMS) algorithm uses the estimates of the gradient in an iterative procedure that makes corrections to the weight vector in the direction of the negative of the gradient vector which eventually leads to the minimum mean square error. Adaptive filters differ from other filters such as Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters in the sense that the coefficients are not determined by a set of desired specifications and the coefficients are not fixed. With adaptive filters the specifications are not known and change with time. They are self-adaptable i.e. recursive in nature. Major applications include: process control, speech processing echo and noise cancellation channel equalization

II. Theory

A. Adaptive filters

Adaptive filtering involves the changing of filter coefficients over time, to adapt to changing signal characteristics [5].These have the ability to adjust their impulse response to filter out the correlated signal in the input. They require little or no a priori knowledge of the signal and noise characteristics. However, if the signal is narrowband and noise broadband, which is usually the case, or vice versa, no a priori information is needed; otherwise they require a signal (desired response) that is correlated in some sense to the signal to be estimated. Moreover adaptive filters have the capability of adaptively tracking the signal under non-stationary conditions.[1] Over the past three decades, real-time adaptive filtering algorithms are quickly becoming practical and essential for the future of communications, both wired and wireless.

B. Adaptive Noise cancellation Principles

As shown in the figure, an Adaptive Noise Canceller (ANC) has two inputs – primary and reference. The primary input receives a signal d from the signal source that is corrupted by the presence of noise n uncorrelated with the signal. The reference input receives a noise u uncorrelated with the signal but correlated in some way with the noise n. The noise no passes through a filter to produce an output y

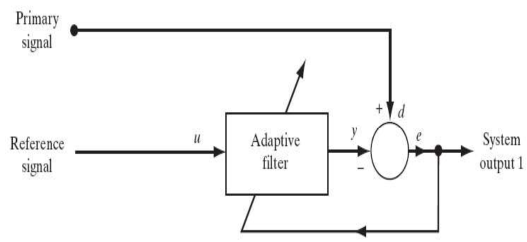

Fig. 1: Adaptive noise cancellation

that is a close estimate of primary input noise. This noise estimate is subtracted from the corrupted signal to produce an estimate of the signal d at e, the ANC system output. Adaptive Filters can be used from application to prediction. It provides a prediction of the present value of a random signal. Parameters used are: u: input of adaptive filter (delayed version of random signal) y: output of adaptive filter d: desired response (random speech signal) e = d – y: estimation of error (system output). Here we have used the most popular adaptive filter. The Least-Mean-Square (LMS) filter which is derived from the steepest descent algorithm. It does not require the gradient to be known rather it is estimated at every iteration. The Least-Mean-Square (LMS) algorithm is accomplished in two steps:

XXX-X-XXXX-XXXX-X/XX/$XX.00 ©20XX IEEE

1. Filtering: First we calculate the output of FIR filter by convolving reference signal (pure noise signal) and taps.

2. Adaptation process: The recursive adjustment of filter tap-weights based on the estimation error. We now consider this adaptive noise cancelation: The primary signal d is the desired signal corrupted with noise. Let, the reference signal u is the pure noise signal.

Let the input noise be $U = [U_0\ U_1\ U_2\ \ldots\ U_M]$ and the tap-weight vector that is the coefficients of the filter be $W = [W_0\ W_1\ W_2\ \ldots\ W_M]$. Therefore, the output filtered signal is $Y = U^T W = W^T U$. The error is $e = d - Y = d - U^T W$. The square error is $e^2 = d^2 + W^T U U^T W - 2dU^T W$, the expected mean square is $E[e^2] = E[d^2] + W^T E[UU^T]W - 2E[dU^T]W \ldots\ \ldots\ \ldots\ \ldots\ \ldots(1)$

In the above equation, we consider autocorrelation R and cross-correlation P as follows: $R = E[UU^T]$ and $P = E[dU^T]$. Now we can write the Mean Square Error as $J = MSE \triangleq E[e^2] = E[d^2] + W^T RW - 2PW$ [4]. Therefore, the gradient will be calculated as follows $\nabla = \partial J/\partial W = 2RW - 2P$. The Gradient descent is based on the observation that if the multi-variable function (here for example Mean Square Error, $J(n)$ at time n is defined and differentiable with respect to a parameter (here for example tap-weight vector, Wn at time n) then that multi-variable function $J(n)$ will decrease fastest if one goes in the direction of its negative gradient $(-\nabla(n))$. Therefore, if we consider: $W(n+1) = W(n) + 1/2\ \mu(-\nabla(n))$ the Mean Square Error J(n) will decrease fastest to reach its local minimum, with μ as the step-size parameter or weighting constant. Since $\nabla = \partial J/\partial W = 2RW - 2P$, we can write $W(n+1) = W(n) + \mu(P - RW(n))$. The instantaneous estimates of the auto-and cross-correlation are: $R \cong U(n)UT(n)$ and $P \cong U(n)dT(n)$.
$W(n+1) = W(n) + \mu(U(n)dT(n) - U(n)UT(n)W(n))$is
Simplified to $W(n+1) = W(n) + \mu U(n)(dT(n) - UT(n)W(n)$
And further simplified to

$W(n+1) = W(n) + \mu U(n)e(n)$ …… ……. ……. (2)

The equation (2) is the weight adaptation algorithm that plays a major role in adaptive filters and is implemented in LMS adaptive filtering. The basic premise of the LMS algorithm is the use of the instantaneous estimates of the gradient in the steepest descent algorithm:

$Wk[n+1] = Wk[n] + \mu\nabla n,k$[4]

It has been shown that: e[n]= d[n] –y[n] is the error signal $\nabla n,k = e[n]u[n-k]$ Finally $wk[n+1] = wk[n] + \mu e[n]u[n-k]$ [4] μ: step size parameter $\nabla n,k$: gradient vector that makes W[n] approach the optimal value W[opt]. Stability of LMS: The LMS algorithm is convergent in the mean square if and only if the step-size parameter 'mu' is satisfied $0 < \mu < 2/\lambda max$, Here, λ (max) is the largest eigenvalue of the correlation matrix of the input data. However, more practical test for stability is $0 < \mu < 2/$ (input signal power). Larger values for step size will increases the adaptation rate (faster adaptation). But this will increase the residual mean-squared error. We also used the filter order of 11 to calculate the filter taps for the adaptive filter coefficients. We faced the challenge of choosing the value of μ. At first we used a default value 0.005 but later we calculated the value of μ using the stability criteria to solve the problem.

III. Results and Discussion

We used five different speech signals. Noise was added from MATLAB directly using the built-in function called the white Gaussian noise. The third parameter of this function is the noise power in decibel. We have plotted three test signals in both time and frequency domain here. We then, calculated the variance and signal to noise ratio of the desired signal and output (de-noised) signal.

A. Figures and Tables

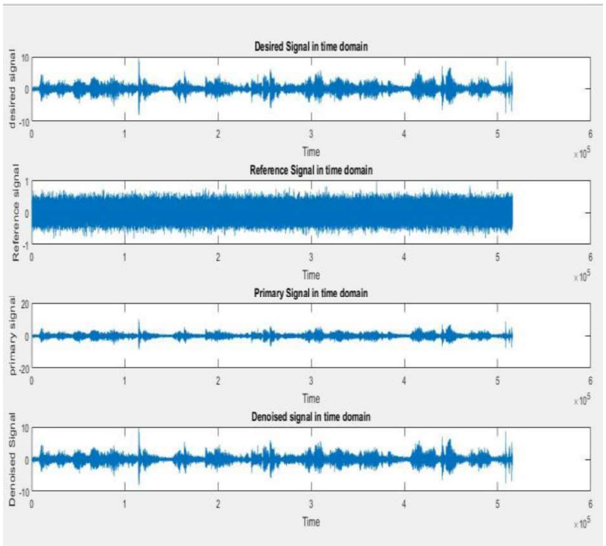

Fig. 2. Output of the signals in Time domain for test signal 1

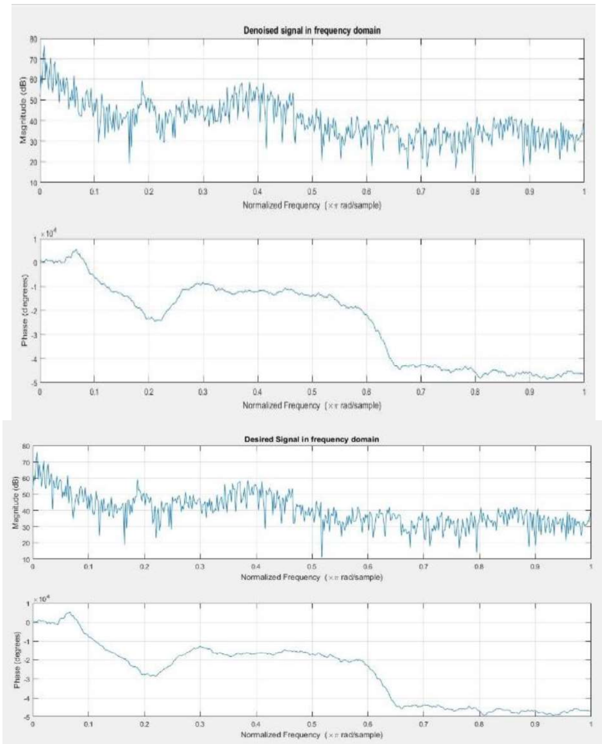

Fig. 3. Desired signal and de-noised signal in frequency domain for test signal 1

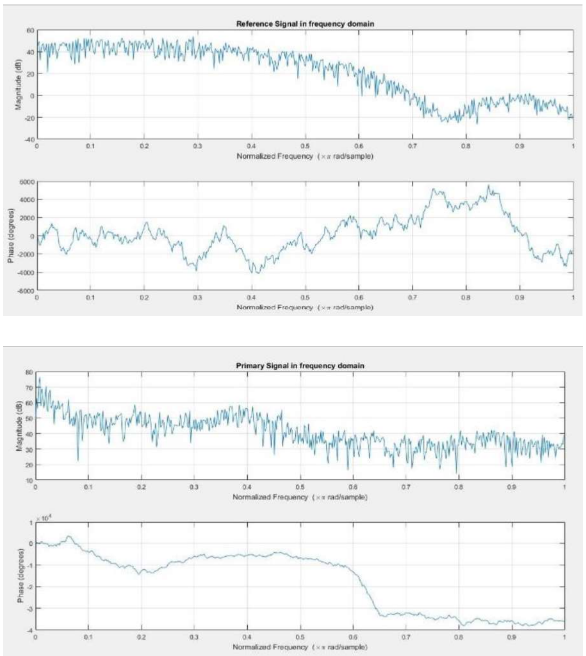

Fig. 4. Reference and primary signal in frequency domain for test signal 1

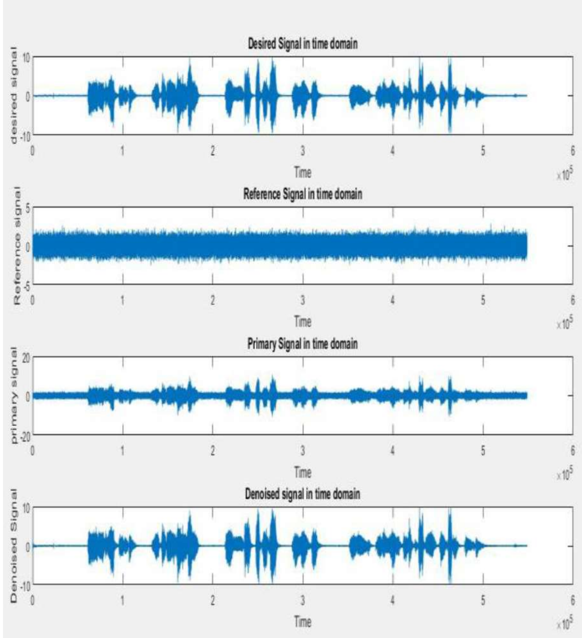

Fig. 5. Output of the signal in time domain for test signal 2

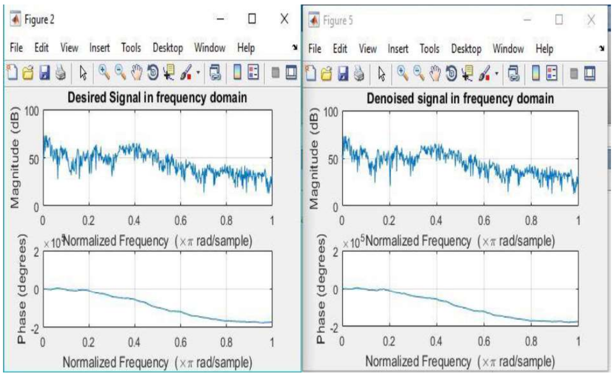

Fig. 6 Desired and de-noised signal in frequency domain for test signal 2

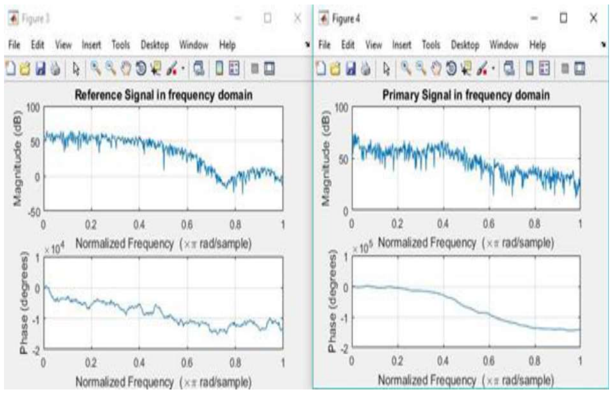

Fig. 7. Reference and primary signal in frequency domain for test signal 2

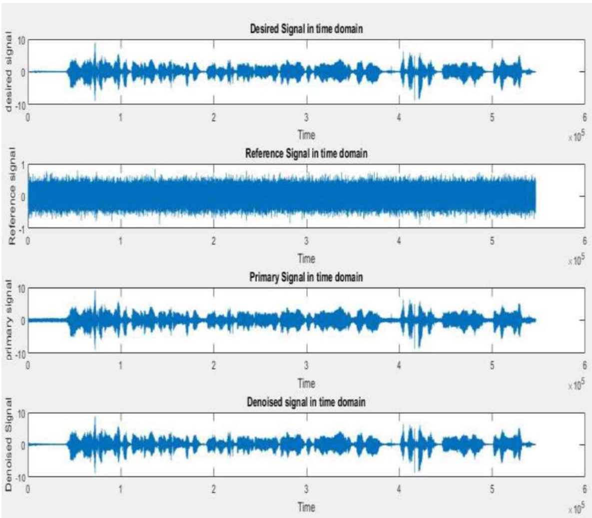

Fig. 8. Output of the signals in time domain for test signal 3

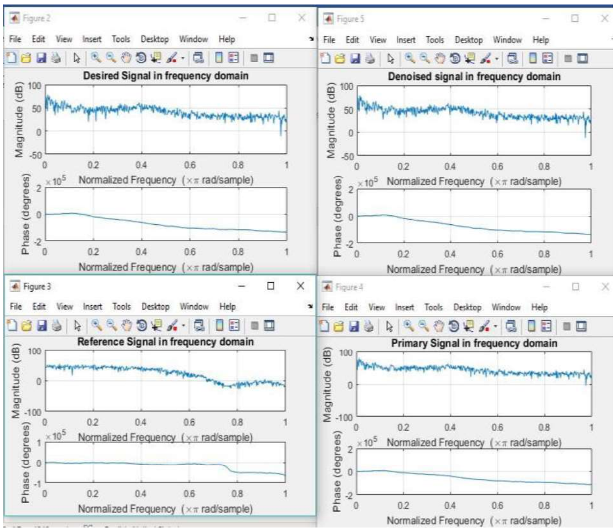

Fig. 9 Output of the signals in frequency domain for test signal 3

We see from the graphs that for different signals the algorithm almost gives correct wave shapes for both the desired and de-noised signal. Even the outputs of the magnitude and phase response are also similar. The plots of the noise signals are also shown for better clarification. In the table we have shown the variance, signal to noise ratio and run-time of the algorithm. The run time is quite good and can be implemented in real time. The table shows the variance value and it is seen that as noise power gets high the variance value degrades a little bit. We see from the table beside that five different signals are tested and for different values of the noise power different results of variances and run time has been calculated. Besides, that we have used a threshold value of 0.4 in designing the filter for filtering the noise signal u. There we have used the window-based filter design technique. We utilized the hamming window for the design of an 11^{th} order low pass filter having angular frequency 0.4 for better results. However, for testing purposes we can change this threshold value. This threshold value along with the value of step size were difficult part to determine in this lab homework. On the other hand, looking into the values of the table it is quite satisfactory that the results are almost correct for five different test signals.

TABLE I, TABLE FOR COMPARISON OF THE RESULTS OF 5 TEST SIGNALS

Test No	Type of signal	Two parameters		Noise variation and runtime	
		Signal to Noise Ratio (SNR) in dB	Variance	Noise power (dB)	Run-time(seconds)
1	Desired	-21.8907	0.9999	-10	5.390869
	De-noised	-21.9402	0.9998	-10	5.390869
2	Desired	-11.7884	1.0000	-5	5.644972
	De-noised	-11.8008	0.9985	-5	5.644972
3	Desired	-13.5354	0.9999	0	6.032574
	De-noised	-13.6654	0.9935	0	6.032574
4	Desired	-12.0934	0.9999	5	5.501564
	De-noised	-12.2775	0.9848	5	5.501564
5	Desired	-12.0934	0.9999	10	5.551959
	De-noised	-12.7210	0.9736	10	5.551959

IV. Conclusion

Adaptive Noise Cancellation is an alternative way of cancelling noise present in a corrupted signal [1]. The principal advantage of the method are its adaptive capability, its low output noise, and its low signal distortion. The adaptive capability allows the processing of inputs whose properties are unknown. Output noise and signal distortion are generally lower than can be achieved with conventional optimal filter configurations. The simulation results verify the advantages of adaptive noise cancellation. In each instance cancelling was accomplished with little signal distortion even though the frequencies of the signal and interference overlapped. In this lab homework, only the Least-Mean-Squares Algorithm has been used. Other adaptive algorithms can be studied and their suitability for application to Adaptive Noise Cancellation compared. Some other algorithms that can be used are: Recursive Least Squares, Normalized LMS, Variable Step-size algorithm etc [1]. In future, we can use these other techniques to improve the complexity of the algorithm to have a better run time.

References

[1] Adaptive noise cancellation: https://www.cs.cmu.edu/~aarti/pubs/ANC.pdf

[2] Adaptive Filter Theory by Simon Haykin, 4ed,2002 Prentice Hall

[3] B. Widrow and S.D. Stearns, Adaptive Signal Processing, Prentice Hall, Englewood Cliffs, NJ, 1985.

[4] https://users.encs.concordia.ca/~amer/teach/elec6601/src/notes/Ch16_adaptiveFilters.pdf

[5] https://www.mathworks.com/help/dsp/ug/overview-of-adaptive-filters-and-applications.html