DSP LAB PROJECT REPORT

Adaptive noise cancellation using least mean square (LMS)

Sri Sai Harshavardhan Reddy Vippala
M.S Computer Engineering
NYU Tandon
email: ssv315@nyu.edu

Aim: Objective of this project is to filter out the additive noise in a noisy signal i.e. noise reduction by means of adaptive filtering techniques. LMS (least mean square error) based noise cancellation technique can be used on input noisy signal to obtain the de-noised signal as the output. We will combine audio sample of speech (original signal) with real noise to generate noisy signal with a certain signal-to-noise ratio (SNR) using MATLAB.

Introduction

Noise cancellation refers to techniques in signal processing for the removal of unwanted noise (undesired frequencies) from the signal. If these unwanted noises are removed it is known as noise removal and if they are reduced/suppressed it is known as noise cancellation. ANC (Adaptive noise cancellation) is a type of optimal filtering that producing an approximate of the noise by filtering the reference input and then removing this noise estimate from the primary input containing both signal and noise. our aim is to remove unwanted noise from the signals using adaptive filtering techniques. Adaptive filters are self-designing using a recursive algorithm. An adaptive filter will iteratively change its coefficients based on a given criteria, to improve its performance.

The least Mean Square (LMS) algorithm uses the approximation of the gradient in an iterative procedure that modifies the weight vector in the direction of the negative of the gradient vector which results in minimum mean square error. Unlike Finite Impulse Response (FIR) and Infinite Impulse Response (IIR) filters, in adaptive filters coefficients are not determined by a set of desired specifications and are not constant. Adaptive filters are self-adaptable in nature i.e. their specifications are not known and change with time constantly. Adaptive filters have extensive applications in the field of signal processing, some of which are: process control, speech processing echo and noise cancellation.

Theory

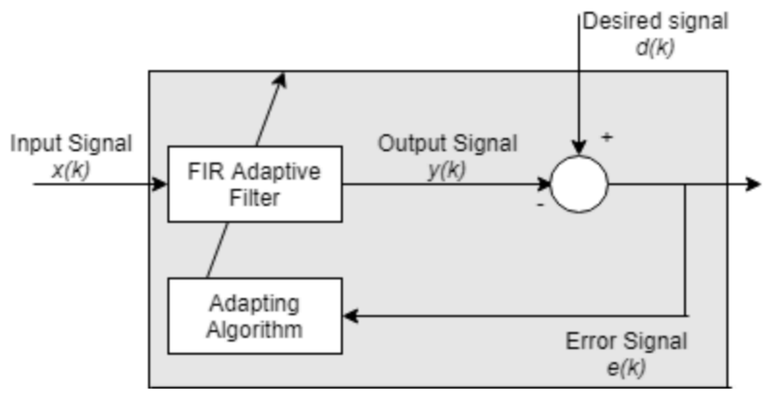
A. Adaptive Filters

Adaptive filters coefficients change iteratively to make the filter converge to an optimal state. The criteria for optimization is a cost function which is usually mean square of the error signal between the output of the adaptive filter and digital signal. The filter iteratively modifies its coefficients which result in mean square error (MSE) to converge to its minimal value. The output of the filter, y(k) is very close to the desired signal d(k). Adaptive filters can adjust their impulse response to filter out the correlated signal in the input. Usually they require little or no prior knowledge of signal and noise characteristics. If the input signal is narrowband(signals which occupy less frequency spectrum and require less transmit power for a given application) and noise broadband(noise that has a continuous spectrum), no prior information is required ,otherwise they require a signal(desired response) that is correlated to the signal to be estimated.

B. Adaptive noise cancellation principles

There are two main components of an ANC, digital filter and adaptive algorithm. Digital filter is used to adjust weights and adaptive algorithm tells us how we can change the weights. We have two inputs into the adaptive filter:

General Adaptive Filter Algorithm



$d(k) = s(k) + n(k)$

where

$s(k)$ = desired signal,

$n(k)$ = noise

$x(k)$ = a measure of contaminating signal which is in some way correlated with noise $n(k)$.

$x(k)$ gives us an estimate of $n(k)$, call it $\grave{n}(k)$. We try to find an estimate of $s(k)$, by subtracting our best estimate of the noise signal. Let $\grave{s}(k)$ be our estimate of the desired signal $s(k)$.

$$\grave{s}(k) = y(k) - \grave{n}(k) = s(k) + n(k) - \grave{n}(k)$$

Our main objective is to produce an optimum $\grave{n}(k)$ such that we have an output very similar to our desired signal.

Adaptive Filters are used from application to prediction. It can calculate a prediction of a random sign al for its present value. Parameters used are, u: input of adaptive filter (delayed version of random signal) y: output of adaptive filter, d: desired response (random speech signal) and e = d - y: estimation of error (system output).

The LMS filter is derived from the steepest descent algorithm. Its gradient is estimated at every iteration and is not required initially for calculations.

LMS Adaptive Filter

$x(\mathrm{k})$ – Input signal to the filter.

$d(\mathrm{k})$ – Reference to the adaptive filter.

$e(\mathrm{k})$ – Error signal, i.e., represents the difference between $d(n)$ & $y(n)$.
Finite response filter (FIR) or Infinite response filter (IIR) can be used as linear filters. We have taken a FIR filter for this example. LMS algorithm (the adaptive algorithm in our case) iteratively modifies the coefficients of the FIR filter to minimize the power of $e(n)$.

When we pass the input signal $x(\mathrm{k})$ through the linear filter (FIR in this case), output signal $y(\mathrm{k})$ is calculated:

$$y(k) = u(k)^{\mathrm{T}} * w(k)$$

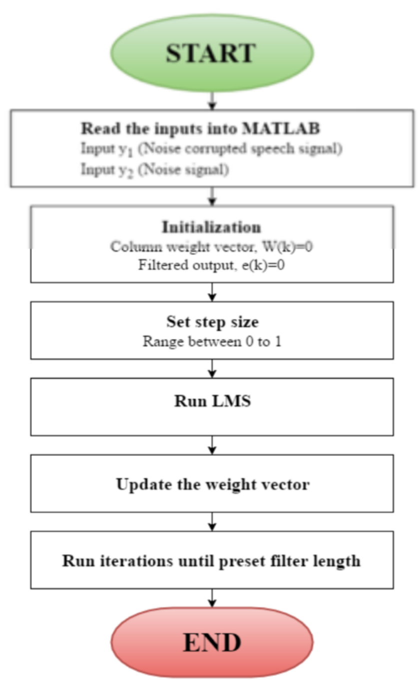
where,
$u(k)$ is input filter vector & $u(k) = [x(k)x(k-1)x(k-2) \dots \dots \dots . . x(k-N+1)]^{\mathrm{T}}$
$w(k)$ is filter coefficient vector & $w(k) = [w_0\,(k) w_1(k) w_2(k) \dots \dots \dots . w_{N-1}(k)]^{\mathrm{T}}$
Error signal, $e(\mathrm{k})$ is calculated:

$$e(\mathrm{k}) = d(\mathrm{k}) - y(\mathrm{k})$$

Based on the error signal, filter coefficients are updated:

$$w(k+1) = (1-\mu).w(k) + \mu.e(k).u(k)$$

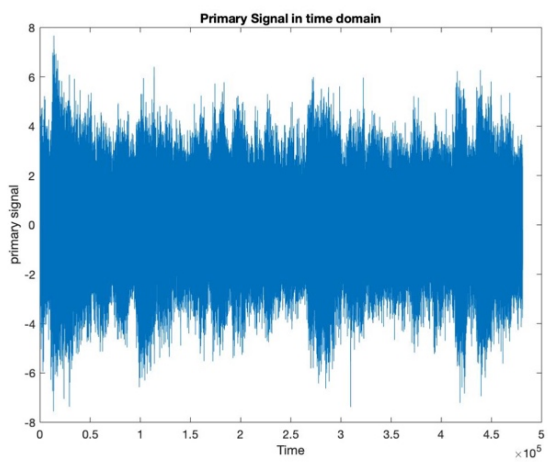
μ is the fixed step size of the adaptive filter. We have taken the value of μ as 0.003 in the demo. LMS provides fast convergence speed but provides a suboptimal solution in low SNR (signal to noise ratio) environment.
Adaptive filters have a wide range of applications. It can be used to find the response of an unknown system, determining inverse response of an unknown system, to remove noise from unknown system, predicting future values of a continuous signal. Larger values for step size will increases the adaptation rate (faster adaptation), but this will increase the residual mean-squared error.

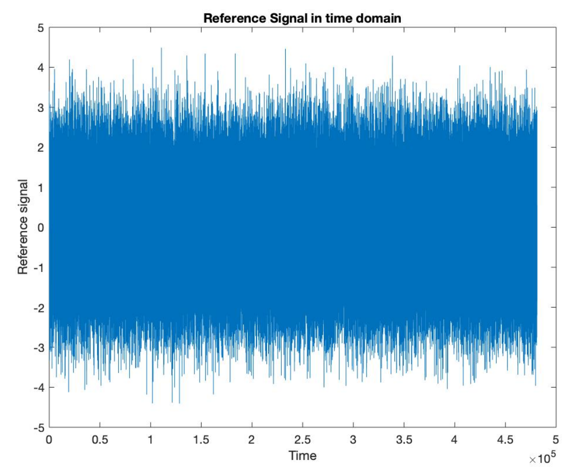


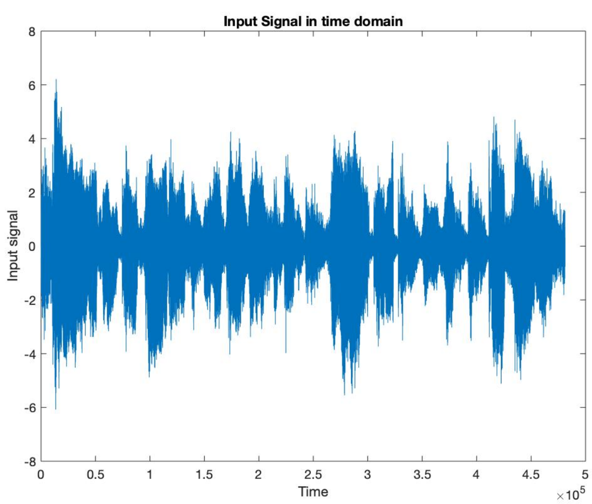
Results and discussion

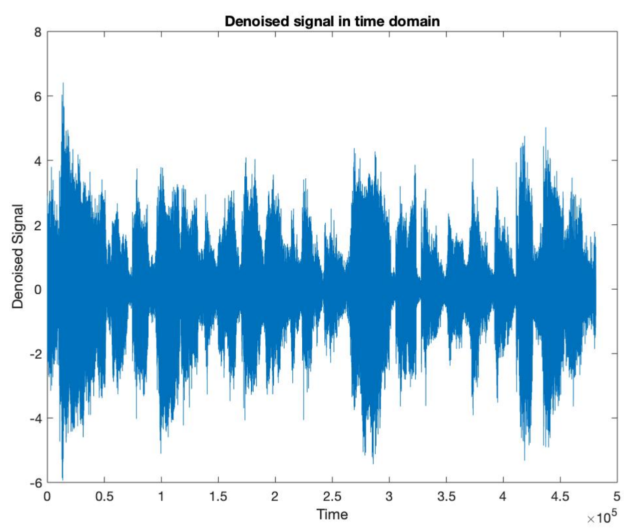
We have taken an input test signal in .wav format. Noise is added in MATLAB using wgn function. White gaussian noise is added to the test signal with a power of 8 dB (noise power). Then we have plotted our input signal, reference(noise) signal, primary signal (Signal + noise) and Output signal in time domain. We have also plotted the Input and Output signals in frequency domain. SNR (signal to noise ratio) of both Input and Output signals is calculated to assess the extent of noise cancellation in our output signal. Variance of both the signals is also calculated.

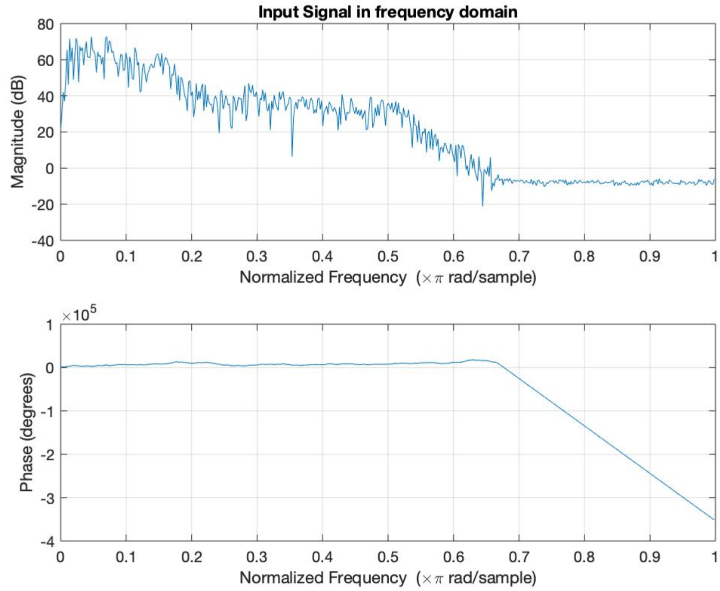
Plots for signals in time domain:

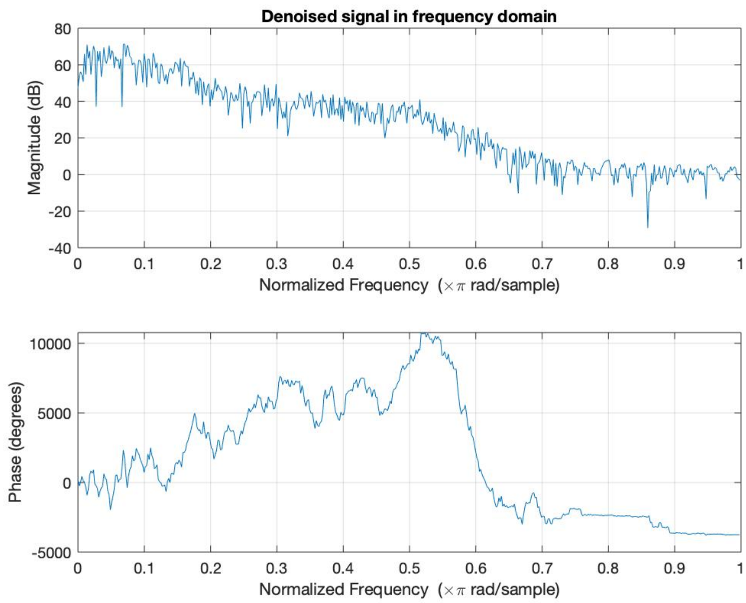












SNR and variance of two input signals is shown in the table:

	SNR	Variance
Input	-18.9825	1.0000
Output	-19.1242	1.0049

We can see from the plots that the algorithm gives almost similar wave shapes for both Input and Output signals. Even the plots of frequency domain are very similar. We have similar SNR values for both Input and Output signals. We can also hear the filtered output to notice the difference between the signal mixed with noise and the output signal.

Conclusion

Adaptive noise cancellation can be used for noise cancellation in a signal corrupted with noise. The main feature of the algorithm is its ability to adapt and change its weight coefficients to produce low output noise with less distortion. LMS is used in to filter the input signal in this case, but there have been several developments in the area of adaptive filters and various new algorithms have been introduced which have higher precision, less distortion and are computationally easier. VS-LMS (variable step size-LMS), RLS (Recursive least square), NLMS (Normalized-LMS) are some of the other adaptive filters.

However, according to a research observation comparing various VS-LMS algorithms found that no VS-LMS algorithm is as adaptable, robust and convenient for real-time applications as the NLMS (Normalized least mean square method) which still dominates adaptive solutions in signal processing.