Institute of Information Technology (IIT)

Jahangirnagar University

Lab Report: 08

<u>Submitted by:</u>

Name: Md. Shakil Hossain

Roll No: 2023

Lab Date: 30-08-2023

Submission Date: 04-09-2023

EXPERIMENT NO: 08

NAME OF THE EXPERIMENT

Adaptive Filtering Using the Least Mean Squares (LMS) Algorithm

OBJECTIVE

1.Noise Reduction and Signal Recovery.
2.Performance Assessment through Mean Squared Error (MSE).

APPARATUS

1.MATLAB

THEORY

In signal processing, the goal is to clean up noisy signals. If a signal is distorted by unwanted noise, like static on a radio.

To remove this noise, we use a mathematical technique called the LMS (Least Mean Squares) algorithm. It listens to the noisy signal and continuously adjusts itself to reduce the noise while preserving the original signal.

The LMS algorithm is like an automatic cleaner for your audio. It's always working to make the signal clearer by getting rid of the unwanted interference. We can measure its success by checking how close the cleaned-up signal is to the original, noise-free audio. The goal is to make this difference as small as possible, ensuring that the algorithm effectively removes noise and restores the original signal's quality.

PROGRAM

```
clc;
clear all;
close all;
% Generate a time vector from 0 to 1 with a step of 0.001 seconds
t = 0.001:0.001:1;
% Create a clean sinusoidal signal with a frequency of 50 Hz and amplitude of 2
s = 2 * sin(2 * pi * 50 * t);
% Generate a noisy signal by adding Gaussian noise (mean=0, std=0.9) to the clean signal
n = numel(s);
A = s + 0.9 * randn(1, n);
% Initialize filter parameters
m = 25;
w = zeros(1, m);
wi = zeros(1, m);
e = zeros(1, n); % Initialize the error signal as a zero vector
mu = 0.0005;
% Adaptive filtering loop
for i = m:n
e(i) = s(i) - sum(wi .* A(i:-1:i-m+1));
w = w + 2 * mu * e(i) * A(i:-1:i-m+1);
end
% Plot the original signal, noisy signal, and filtered signal
figure;
subplot(3,1,1);
```

```
plot(t, s);
title('Original Signal');
xlabel('Time (s)');
ylabel('Amplitude');
subplot(3,1,2);
plot(t, A);
title('Noisy Signal');
xlabel('Time (s)');
ylabel('Amplitude');
subplot(3,1,3);
plot(t, s - e); % Filtered signal is s - e
title('Filtered Signal');
xlabel('Time (s)');
ylabel('Amplitude');
% Plot the error signal and monitor convergence
figure;
plot(t, e);
title('Error Signal');
xlabel('Time (s)');
ylabel('Amplitude');
% Calculate the Mean Squared Error (MSE) to assess performance
mse = mean(e.^2);
disp(['Mean Squared Error (MSE): ', num2str(mse)]);
```

RESULT

```
Mean Squared Error (MSE): 1.952
>>
```

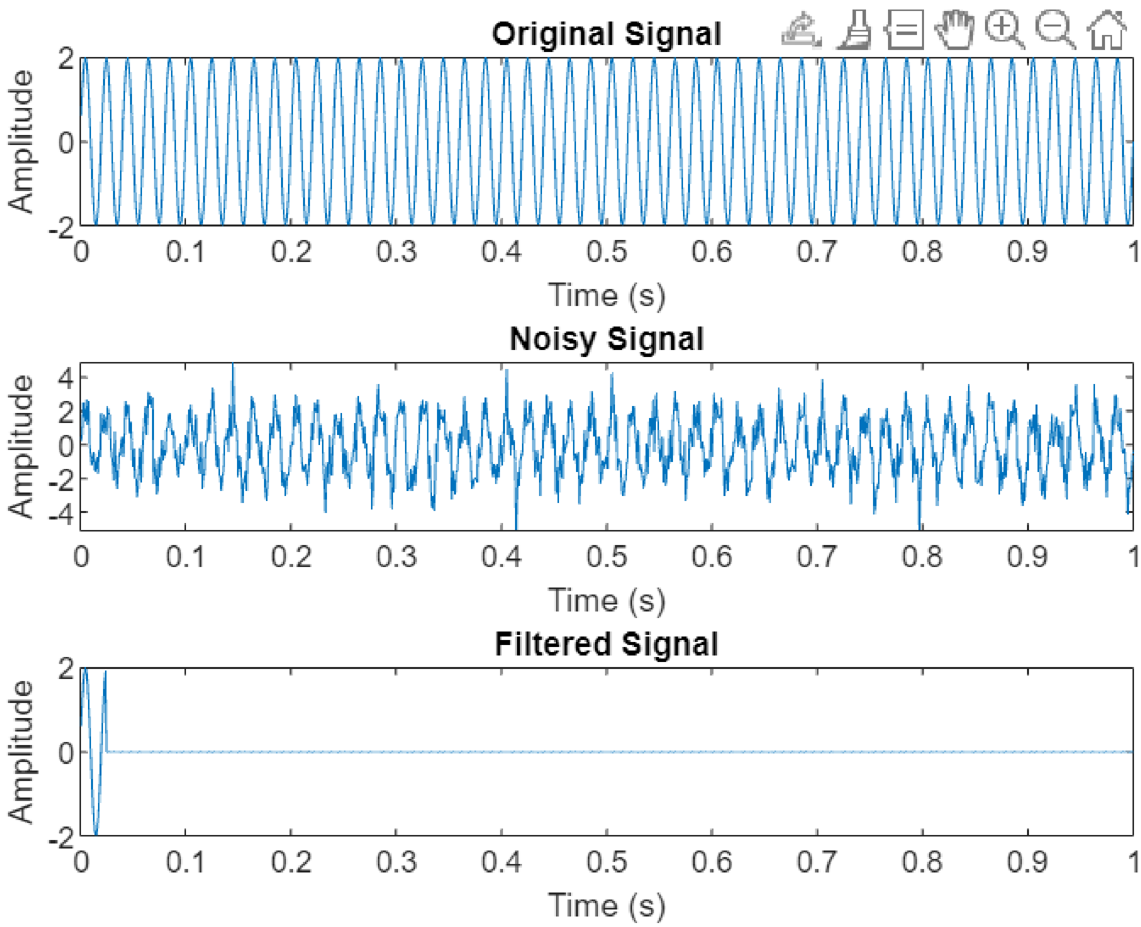

Fig-01: Noisy signal and filtered signal for M=25

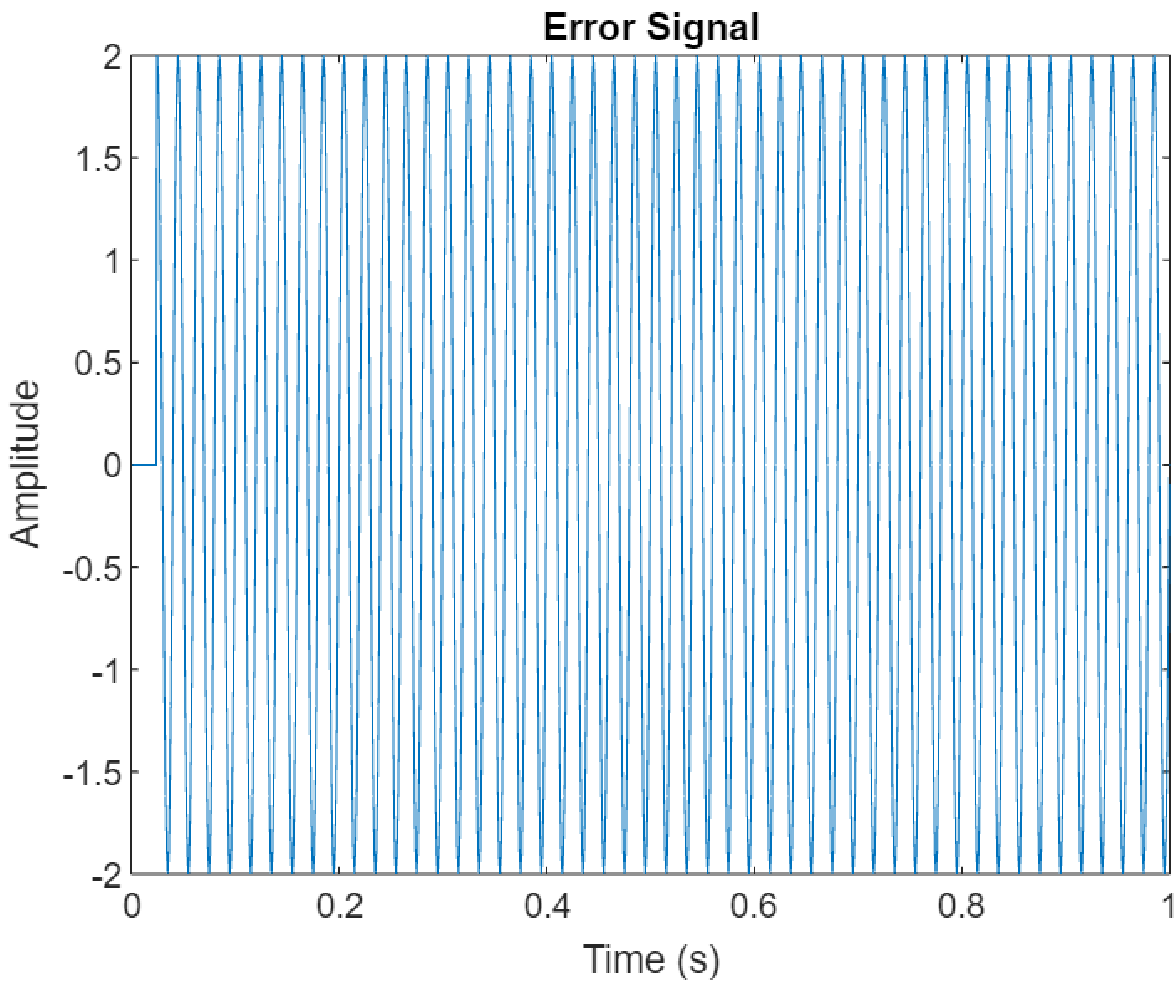

Fig-02: Error signal For M=25

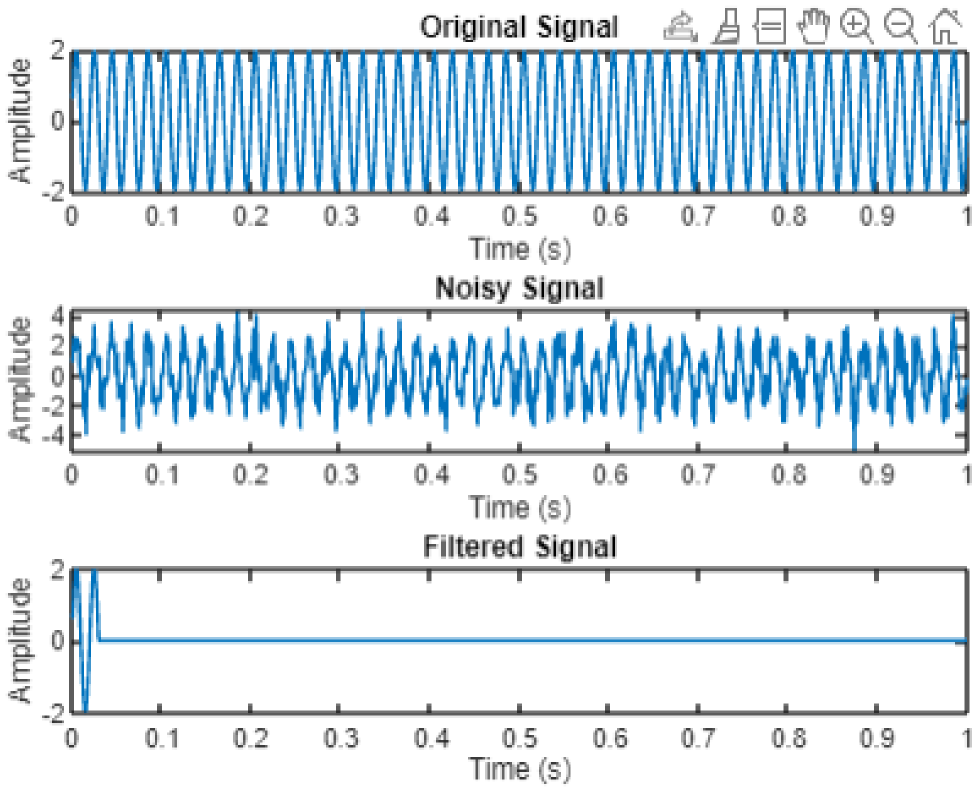

Fig-03: Noisy signal and filtered signal for M=30

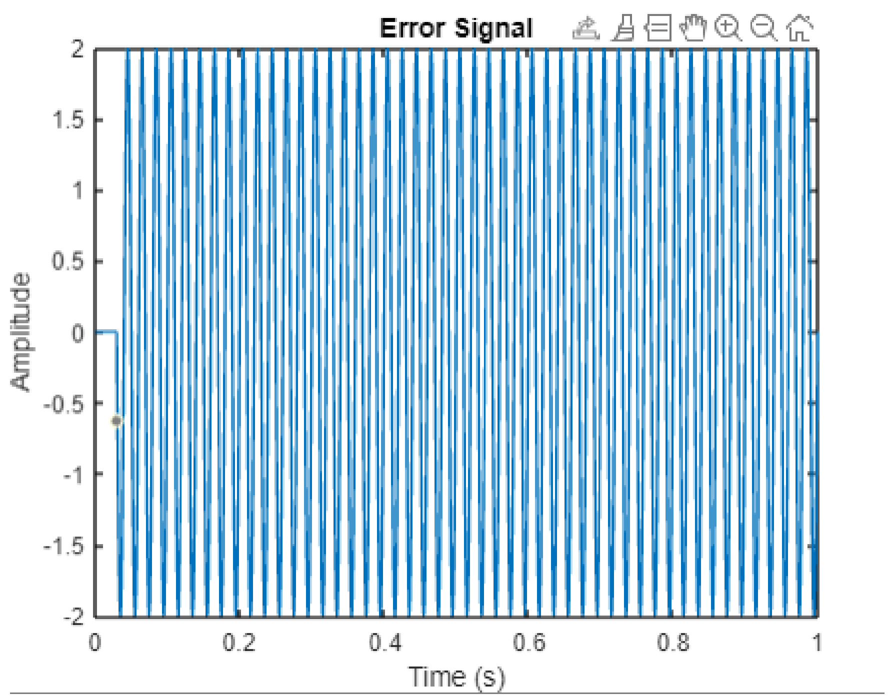

Fig-04: Error signal For M=30

DISCUSSION

In this experiment, we used the LMS algorithm to lower noise and improve signal quality. The program gradually reduced interference by continuously adapting to the noisy input. Monitoring its performance through the Mean Squared Error (MSE) allowed us to gauge its effectiveness. Our discussion centers on the successful noise reduction achieved and the practical applications of adaptive filtering in improving signal quality, including its relevance in fields such as communications, audio processing, and data analysis.

CONCLUSION

In conclusion, our experiment showed how the LMS algorithm for noise reduction may be used in real-world settings. By reducing interference, the program successfully cleaned up cluttered signals, enhancing signal quality. By utilizing the Mean Squared Error (MSE) as a performance metric, we verified the algorithm's success in noise reduction. This experiment highlights the significance of adaptive filtering techniques in various domains where signal quality enhancement is crucial, from audio processing to telecommunications.

REFERENCE

[1]Wikipedia Contributors, “Least mean squares filter,” *Wikipedia*, Available:https://en.wikipedia.org/wiki/Least_mean_squares_filter [Accessed: Aug. 31, 2023]