COMPARATIVE STUDY OF DIFFERENT DENOISING TECHNIQUES

MAJOR PROJECT

Submitted in partial fulfilment of the requirements for the degree of

Bachelor of Technology by

Harshavardhan D - 191EE123
Shashank Shekhar Panda - 191EE248
Karan J Tejas - 191EE126
Aditya Devanand Kaloji- 191EE102

DEPARTMENT OF ELECTRICAL AND ELECTRONICS ENGINEERING NATIONAL INSTITUTE OF TECHNOLOGY KARNATAKA, SURATHKAL SRINIVASNAGAR 575025 KARNATAKA INDIA

July 2022

DECLARATION

by the B. Tech Undergraduate

We hereby declare that the Major Project entitled **“A Comparative study on Denoising techniques”** which is being submitted to the **National Institute of Technology Karnataka, Surathkal** in partial fulfilment of the requirements for the award of the Degree of **Bachelor of Technology** is a bonafide report of the project work carried out by us. The material contained in this Major Project has not been submitted to any University or Institution for the award of any degree.

Harshavardhan D
(Reg. No.: 191EE123)
Department of Electrical and Electronics Engineering

Shashank Shekhar Panda
(Reg. No.: 191EE248)
Department of Electrical and Electronics Engineering

Karan J Tejas
(Reg. No.: 191EE126)
Department of Electrical and Electronics Engineering

Aditya Devanand Kaloji
(Reg. No.: 191EE102)
Department of Electrical and Electronics Engineering

Place: NITK, Surathkal Date:

<u>CERTIFICATE</u>

This is to certify that the Major Project entitled **"A Comparative study on Denoising techniques"** submitted by **Harshavardhan D (Reg. No.: 191EE123), Shashank Shekhar Panda (Reg. No.: 191EE248), Karan J Tejas (Reg. No.: 191EE126) and Aditya Devanand Kaloji (Reg. No.: 191EE102)** as the record of the Major Project carried out by them, is accepted as the Major Project submission in partial fulfilment of the requirements for the award of degree of **Bachelor of Technology.**

Research Guide

Dr. Shubhanga K N

Professor

Department of Electrical and Electronics Engineering

National Institute of Technology Karnataka, Surathkal

ACKNOWLEDGEMENT

We would like to articulate our deep gratitude to our project guide Dr. Shubhanga K N, Professor, Department of Electrical and Electronics Engineering ,National Institute of Technology Karnataka, Surathkal, who has always been a source of motivation for carrying out this project. His constant inspiration and ideas have helped us in shaping this project. We are thankful to him for giving his valuable time despite his busy schedule for completion of our project.

A word of thanks goes to the Head of Department Dr. Dattatraya Narayan Gaonkar, for allowing us to use department facilities and gather the literature for carrying out this project, along with all the Faculty, Staffs and Students of Department of Electrical and Electronics Engineering who have helped us in carrying out our work.

We would like to express our gratitude to all those authors whom we have mentioned in the reference section for giving shape to our thoughts through their path breaking endeavours. Last but not the least we would like to thank our friends and family for supporting us in every possible way while carrying out this project work.

Harshavardhan D
(Reg. No.: 191EE123)
Department of Electrical and Electronics Engineering

Shashank Shekhar Panda
(Reg. No.: 191EE248)
Department of Electrical and Electronics Engineering

Karan J Tejas
(Reg. No.: 191EE126)
Department of Electrical and Electronics Engineering

Aditya Devanand Kaloji
(Reg. No.: 191EE102)
Department of Electrical and Electronics Engineering

ABSTRACT

Signal noise is unwanted interference that degrades the quality of signals. The most frequent and evident issue brought on by signal noise is the distorted interpretation or erroneous display of a process condition by the equipment.

Extreme signal noise, however rare, can cause an apparent signal loss. The majority of contemporary electronic devices have noise filtering built in. This filter won't be enough in excessively noisy surroundings, which can result in the equipment not receiving a signal and no communication at all.

A system that experiences signal noise variations may unintentionally misinterpret the noisy signals, causing relays and alarms to turn on and off at random intervals. An industrial process is improperly controlled in a circumstance like this.

As there are multiple denoising techniques present out there including both traditional methods and learning based methods this project aims to find out the best suitable algorithm given a case.

In this paper we are trying to compare different types of denoising techniques by adding white gaussian noise to a clean signal. The parameter on which different denoising techniques are going to be compared is SNR(Signal to Noise Ratio).

TABLE OF CONTENTS

LIST OF FIGURES

LIST OF TABLES

1. INTRODUCTION

1.1 What is Noise?

The signal is the meaningful information that you're actually trying to detect. The fluctuation or variation that is random, undesirable, which interferes with the original message signal and corrupts the parameters of the message signal is called noise. It is most likely to be entered at the channel or the receiver. Imagine talking on the phone to a friend who is out in the traffic or sitting in a park; aside from your friend's voice (which is regarded as the pure signal in this case), you'll hear a lot of other sounds, such as the sound of vehicles honking, people talking, the traffic police whistling or the birds chirping etc. These sounds can all be categorised as noise in this case because they make it difficult for you to comprehend the signal which you are trying to detect. There are two main ways in which noise appears in any system. One is through some external source while the other is created by an internal source, within the receiver section.

- External Source
  This noise is typically caused by outside factors that could affect the communication channel or medium. It is impossible to entirely stop this noise. Preventing noise from influencing the signal is the best course of action.
  The most prevalent examples of this form of noise are-
  - Atmospheric noise (caused due to irregularities in the atmosphere)
  - Extra-terrestrial noise, such as Cosmic and Solar noise
  - Industrial noise

- Internal Source
  When the receiver is operating, its components make this noise. Due to their continual operation, circuit components may emit a variety of noise. This noise can be measured. The impact of this internal noise may be lessened with an appropriate receiver design.
  Most common examples of this type of noise are −
  - Thermal agitation noise (Johnson noise or Electrical noise).
  - Shot noise (due to the random movement of electrons and holes).
  - Transit-time noise (during transition).
  - Miscellaneous noise is another type of noise which includes flicker, resistance effect and mixer generated noise, etc.

1.2 Why do we need to denoise?

Noise is an inconvenient feature which affects the system performance. Following are the effects of noise. The weakest signal that an amplifier can amplify is indirectly limited by noise. Due to noise, the oscillator in the mixing circuit could have a frequency limit. The functionality of a system is dependent on how well its circuits work. The smallest signal that a receiver can process is constrained by noise. Sensitivity is the smallest quantity of input signal required to produce an output with the desired quality. A receiver system's sensitivity is impacted by noise, which eventually has an impact on the output. Even the process of digitizing an analog signal i.e. the output of an ADC in any circuit, depending on the resolution of the ADC used can result in a processed signal laden with noise. Owing to all these noises the need for techniques to denoise signals so they could be interpreted the way the were meant to be becomes inevitable.

The process of reducing noise from a signal is known as noise reduction or denoising. Both audio and picture noise reduction methods exist. Algorithms for noise reduction may slightly skew the signal. As with common-mode rejection ratio, noise rejection refers to a circuit's capacity to separate an undesirable signal component from the desired signal component.

Numerous researchers have utilised a variety of noise removal strategies recently. We compared the most effective filtering methods and noise-removing algorithms in order to totally remove noise from signals and achieve successful results without compromising the integrity of the signal.

2. LITERATURE SURVEY

Yanxin Wei [1] uses the window function method to select the fast Fourier transform interval, so as to design a FIR digital low-pass filter to achieve the doping in the speech signal in the random noise elimination, and analyze the time domain and frequency domain characteristics of voice signal before and after denoising, compare the quality difference of voice signal before and after filtering.

K. C. Jayashree [2] analyses real time audio signals and try to reduce the noise associated with the message signal under consideration. The main drawback of noise being present in an audio signal, is that it reduces the quality of the signal that is being transmitted within the communication system. For analysis purpose, white gaussian noise(awgn) is concatenated with the audio signal under consideration and the resulting noisy audio signal is subjected to the different filtering techniques like IIR Filter, FIR Filter, Wavelet transform techniques. The noisy audio signal is analyzed with respect to the different filter responses obtained on applying the foresaid methods. A comparative study is done between these techniques to arrive at a technique which would be the most efficient one for audio signal denoising.

Guangyi Chen [3] discusses the denoising of Gaussian additive white noise which is a classical problem in signal and image processing. In this paper, he classifies the most important wavelet denoising methods into different categories and give a brief overview of each method classified. In general, the recently developed block matching and 3D filtering (BM3D) algorithm performs much better than other existing methods published in the literature. We recommend using this method for image denoising because it is currently one of the state-of-the-art denoising methods. The non-local means method and the optimal spatial adaptation (OSA) method are also very successful methods in image denoising.

Anil Chacko [4] discusses the Electrocardiogram (ECG) which shows the electrical activity of the heart and is used by physicians to inspect the heart's condition. Analysis of ECG becomes difficult if noise is embedded with signal during acquisition. In this paper, a denoising technique for ECG signals based on Empirical Mode Decomposition (EMD) is proposed. The noisy ECG signal is initially decomposed into a set of Intrinsic Mode Functions (IMFs) using EMD method. In the proposed technique, the IMFs which are dominated by noise are automatically determined using Spectral Flatness (SF) measure and then filtered using butterworth filters to remove noise. This method is evaluated on ECG signals available in MIT-BIH Arrhythmia database. The experiment results show that the proposed technique performs with better Signal to Noise Ratio (SNR) and lower Root Mean Square Error (RMSE) than the commonly used Wavelet Transform based denoising technique.

3. PROPOSED WORK

3.1 Noise theory

Signal fading, reverberations, echo, multipath reflections, and missing samples are examples of changes in a signal caused by the non-ideal features of the communication channel. The phrase "signal distortion" is frequently used to characterise a systematic undesired change in a signal. A noise process can be further divided into other categories based on its properties in terms of frequency, spectrum, and time:

1. *White noise*: purely random noise has an impulse autocorrelation function and a flat power spectrum. White noise theoretically contains all frequencies in equal power.
2. *Band-limited white noise*: Similar to white noise, this is a noise with a flat power spectrum and a limited bandwidth that usually covers the limited spectrum of the device or the signal of interest. The autocorrelation of this noise is sinc-shaped.
3. *Narrowband noise*: It is a noise process with a narrow bandwidth such as 50/60 Hz from the electricity supply.
4. *Coloured nois*e: It is non-white noise or any wideband noise whose spectrum has a nonflat shape. Examples are pink noise, brown noise and autoregressive noise.
5. *Impulsive noise*: Consists of short-duration pulses of random amplitude, time of occurrence and duration.
6. *Transient noise pulses*: Consist of relatively long duration noise pulses such as clicks, burst noise etc.

Signal to Noise Ratio (SNR): To evaluate how noise affects a signal, people frequently utilise the signal-to-noise ratio (SNR). The quantized signal xq[n] is a superposition of the unquantized, undistorted signal x[n] and the additive quantization error e[n]. This measurement is based on an additive noise model. The SNR is determined by the ratio of the signal powers of x[n] and e[n]. SNR is typically expressed using a logarithmic scale, measured in decibels, to account for the vast range of possible SNR values and the human ear's logarithmic loudness perception (dB).

Additive White Gaussian Noise (AWGN): The noise is assumed to be a stationary additive white Gaussian (AWGN) process in classical communication theory. In practise, the noise is frequently time-varying, correlated, and non-Gaussian, despite the fact that for some problems this assumption is true and results in mathematically convenient and usable solutions. In particular, impulsive noise and acoustic noise, which are non-stationary and non-Gaussian and cannot be described using the AWGN assumption, fall under this category. In MATLAB,

y = awgn(x,snr,'measured', seed)

function is used to pollute the vector signal (x) with additive white gaussian noise [11]. The scalar SNR specifies the signal-to-noise ratio per sample, in dB. If x is complex, *awgn* adds complex noise. Here *'measured'* tells *awgn* to measure the power of x before adding noise and finally *seed* specifies a seed value for initializing the normal random number generator that is used to add white Gaussian noise to the input signal.

After adding noise to our input signals using *awgn* next we use different denoising techniques to isolate the original signal from the noisy signal as mentioned in chapter-2.

3.2 Wavelet based denoising

The use of wavelet transforms in mathematics allows for the analysis of data with features that fluctuate over a range of scales. Features for signals can be transients, steadily changing trends, or frequencies that change over time. Edges and textures are examples of features for photographs. The main reason why wavelet transforms were developed was to solve the shortcomings of the Fourier transform.

Wavelet analysis divides signal into scaled and shifted versions of a wavelet, as opposed to Fourier analysis, which separates data into sine waves with predetermined frequencies. A wavelet is a rapidly fading oscillation that resembles a wave, in contrast to a sine wave. As a result, wavelets may now represent data at many scales. Different wavelets can be used, depending on the application.

Wavelet transform method can be employed for analysis of the audio signal with respect to approximation and detailed coefficients. Wavelet transforms can be classified as, continuous wavelet transform and discrete wavelet transform. Here we utilize discrete wavelet transform technique as it is more suitable for denoising of audio signals.

The audio signal which is to be analysed is taken as input, to this *awgn* is added and the resulting noisy audio signal is obtained. White gaussian noise is preferred as it has almost constant PSD (power spectral density) and for easy and precise analysis. This signal is passed through Level filter. The noisy signal is decomposed into two parts, detailed coefficients and approximation coefficients. The number of levels required for decomposition generally depends upon nature of the signal.

Multilevel decomposition is done to repeat the process of decomposition so that many lower resolution components of the signal can be obtained through wavelet decomposition trees. Wavelet thresholding, the final step is to reconstruct the original audio signal without much loss of information. A construction process which involves using the wavelet coefficients and considering the levels of iteration, successful reconstruction of the original audio signal is obtained.

The noisy signal is decomposed into two parts, detailed coefficients and approximation coefficients. The number of levels required for decomposition generally depends upon nature of the signal.

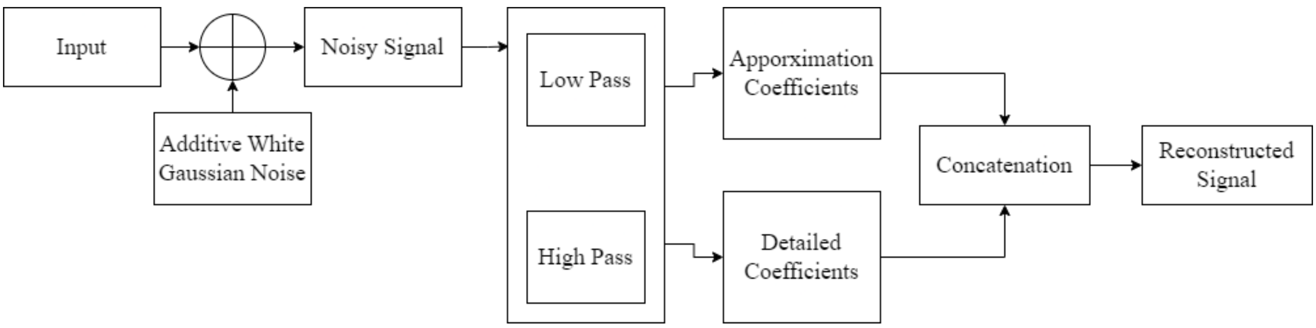

Figure 1 Wavelet transform flow

The wavelets used here is Daubechies (db4) wavelets have highest number N of vanishing moments with the support width 2N-1. db wave solves the problem of signal discontinuities and is applicable for continuous and discrete wavelet transforms.

DWT has two functions wavelet and scaling function

$$\text{Scaling function } \emptyset(t) = \Sigma_{n=0}^{N-1} h[n]\sqrt{2}\emptyset(2t-n)$$

$$\text{Wavelet function } \varphi(t) = \Sigma_{n=0}^{N-1} g[n]\sqrt{2}\emptyset(2t-n)$$

Approximation coefficients:

$$W_{\emptyset}[j_0, k] = \frac{1}{\sqrt{M}} \Sigma_n f[n] \emptyset_{j_{0,k}}[n]$$

Detailed coefficients:

$$W_{\emptyset}[j_0, k] = \frac{1}{\sqrt{M}} \Sigma_n f[n] \varphi_{j_k}[n] \qquad\qquad j \geq j_0$$

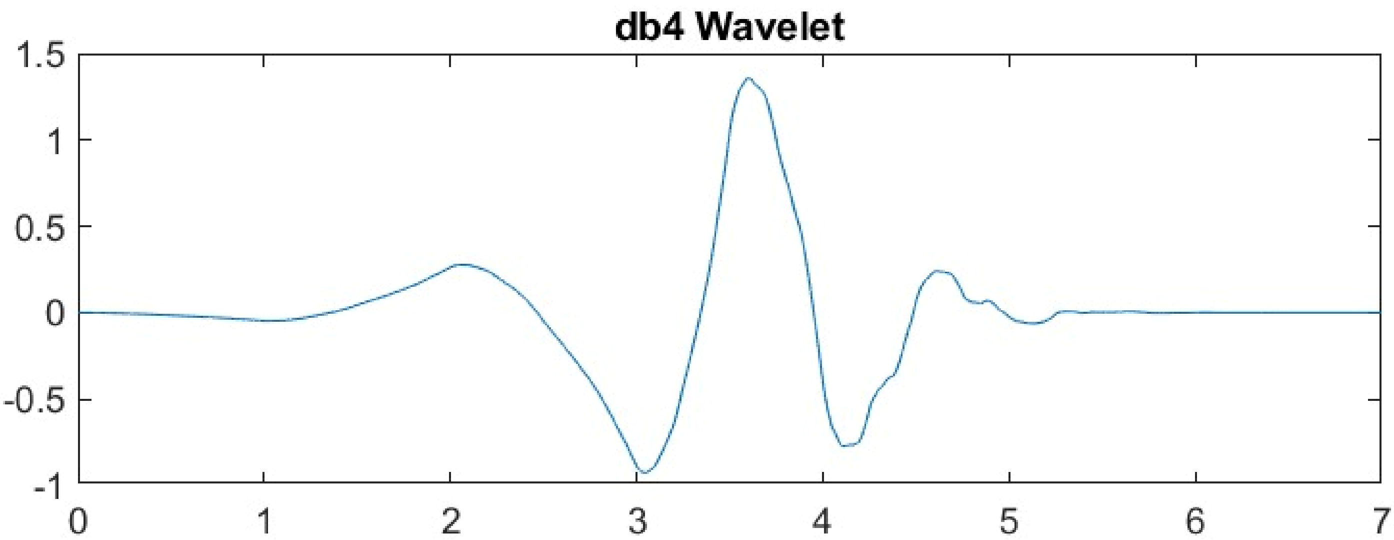

Figure 2 : db4 wavelet

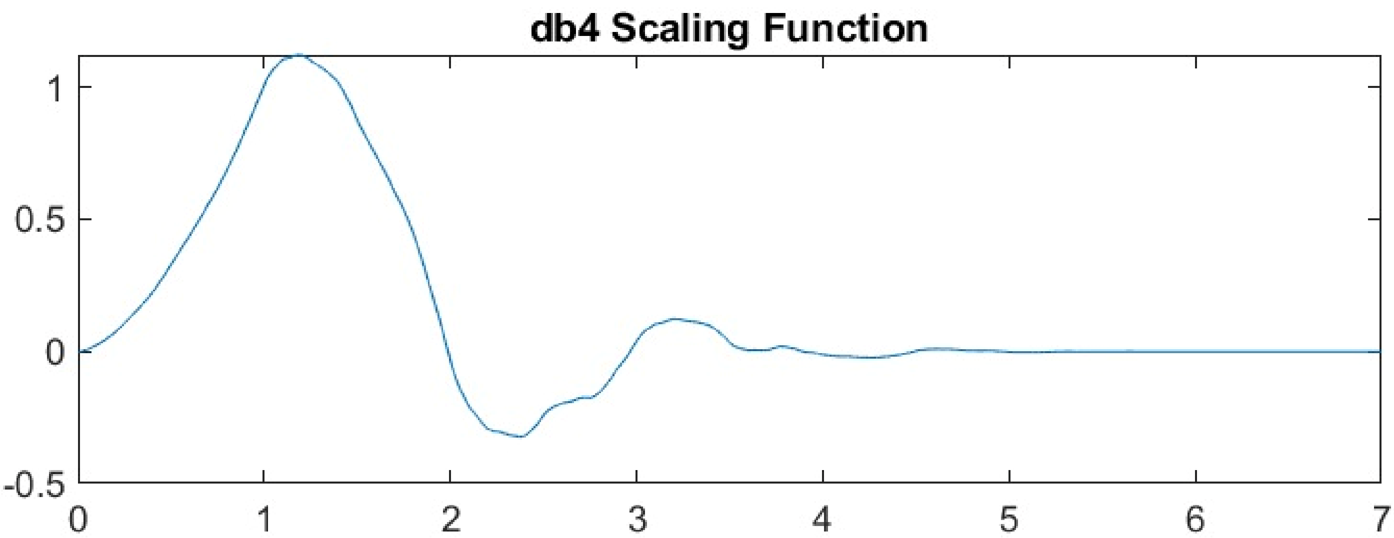

Figure 3 Scaling function

The following MATLAB functions have been utilized for wavelet decomposition and reconstruction to denoise the input signals:

[C, L] = wavedec(x, n, wname, (decomposition function)

This function returns the wavelet decomposition of the 1-D signal x at level n using the wavelet *wname* (db4 in this case) [10]. The output decomposition structure consists of the wavelet decomposition vector c and the bookkeeping vector l, which is used to parse c.

Y = wrcoef(type, C, L, wname, n) (reconstruction function)

This function reconstructs the coefficients at level n vector of type *type* based on the wavelet decomposition structure [C, L] of a 1-D signal using the wavelet specified by *wname* (db4 in this case) [14]. The coefficients at the maximum decomposition level are reconstructed. The length of x is equal to the length of the original 1-D signal.

Now, the parameter n which represents the level of decomposition has its impacts on the output signals. There is no universal value of the level of decomposition which suits all kinds of signals, rather the value of n on which we get good denoising performance depends on the nature of input signal, SNR of input signal (after adding *awgn*) etc. The optimal value of 'n' in these experiments have been found out using trial-and-error method.

We implemented the above algorithm on MATLAB for 4 different input signals and plotted the original signal, the signal after adding white gaussian noise (SNR = 15dB) and the denoised output signal as shown in the figures below.

Input 1 – Frequency readings from a power system taken at 30 samples/second for 180 seconds.

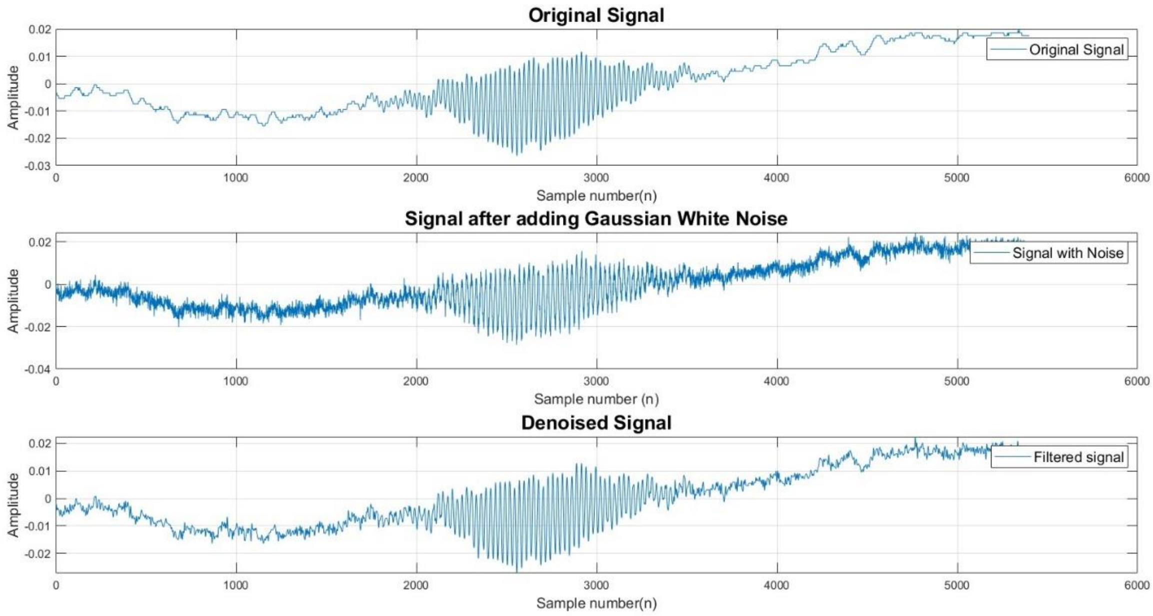

Figure 4 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 20.8425 dB)

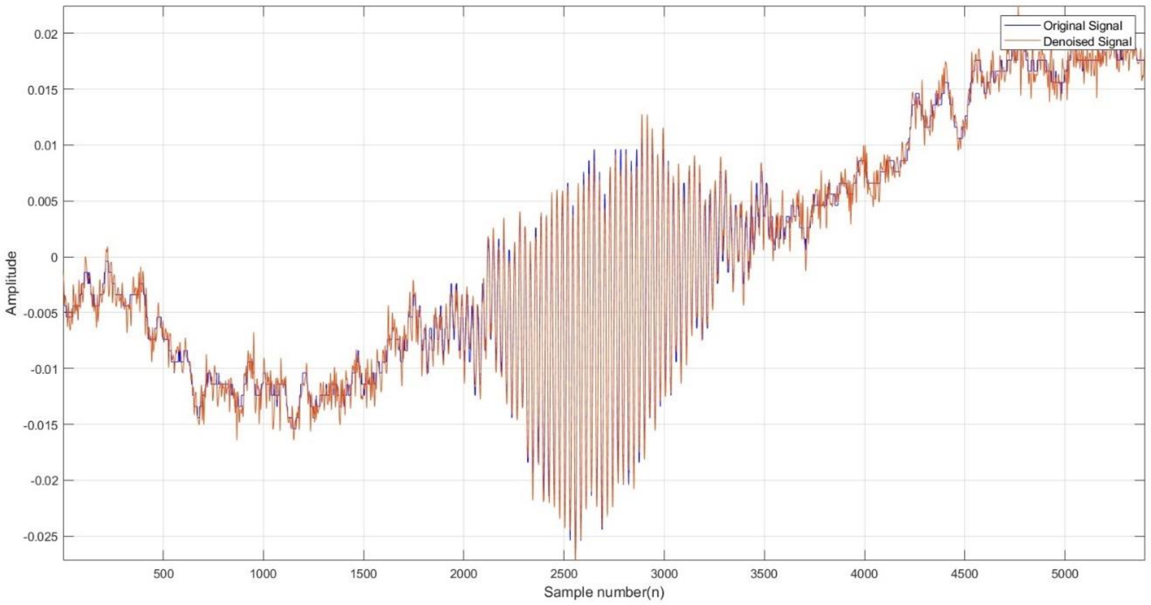

Figure 5 Original signal superimposed with denoised signal

Input 2 – Sine wave of frequency 60Hz

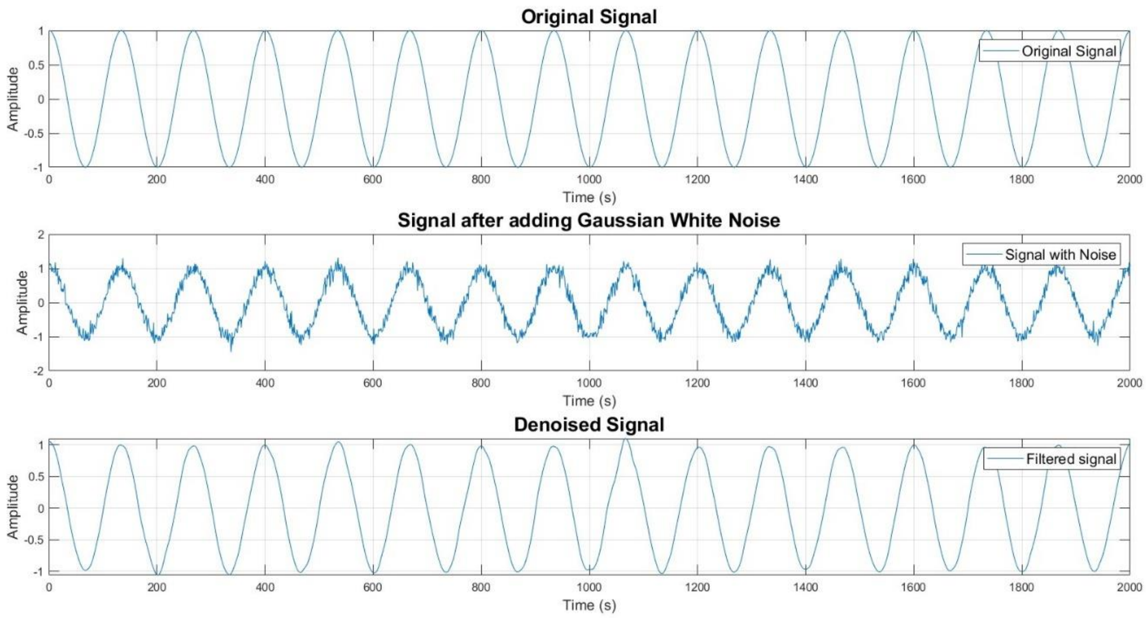

Figure 6 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 27.8395 dB)

Input 3 – Random audio sample

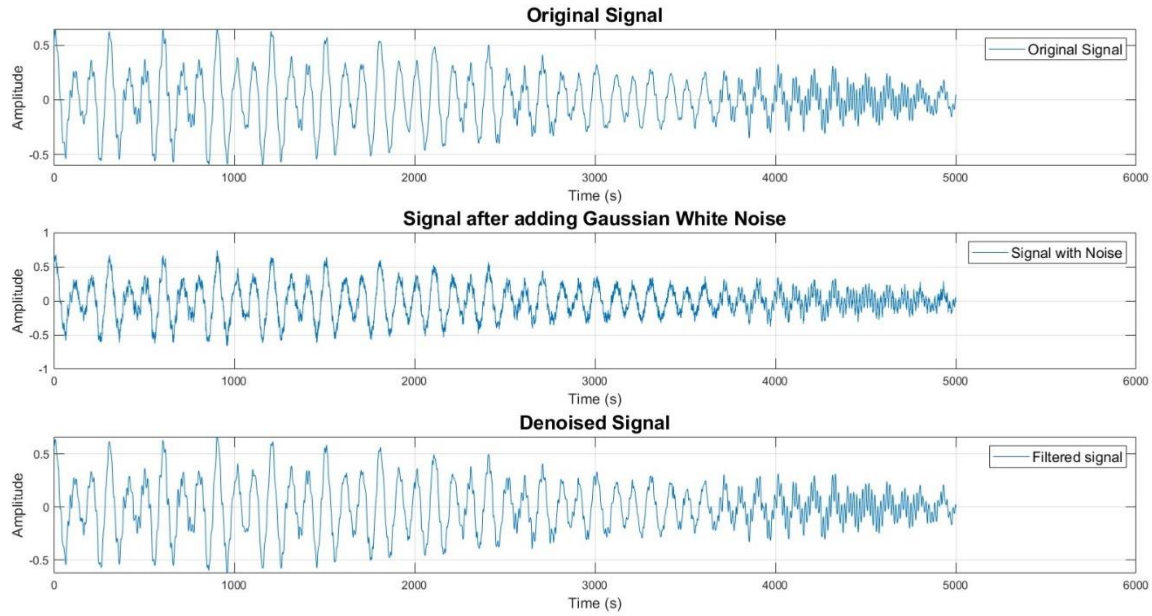

Figure 7 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 20.5533 dB)

Input 4 – Frequency readings from a power system taken at 30 samples/second for 180 seconds.

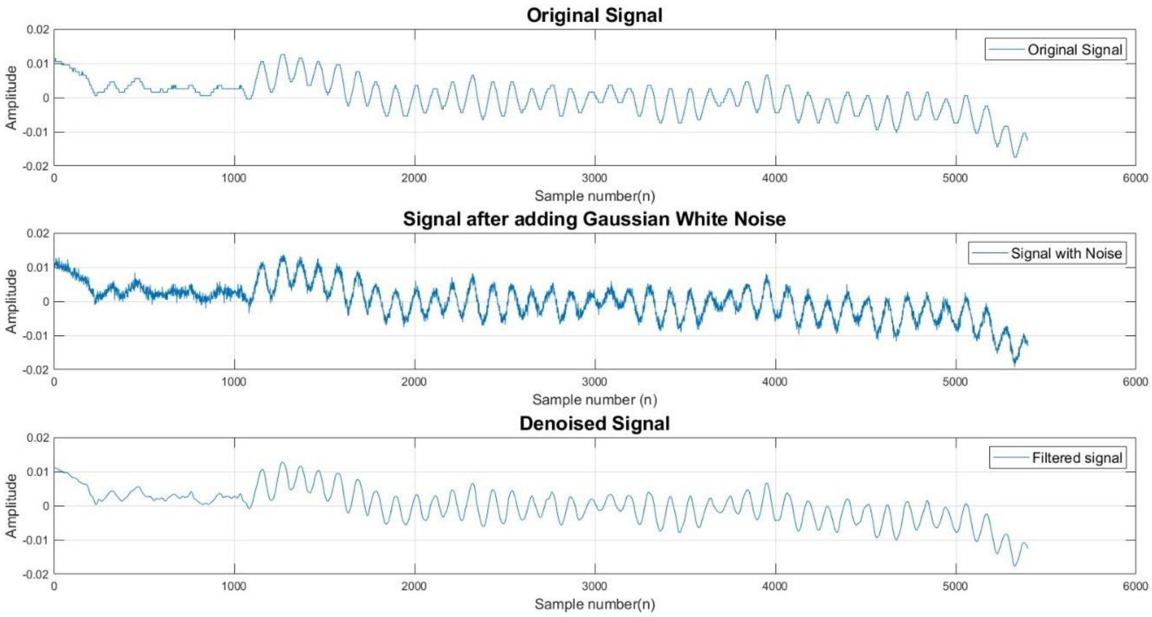

Figure 8 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 22.8226 dB)

3.3 Empirical Mode Decomposition (EMD)

3.3.1 Introduction:

A procedure called the empirical mode decomposition (EMD) approach divides multicomponent signals into a set of amplitude and frequency modulated (AM/FM) zero-mean signals, also known as intrinsic mode functions (IMFs). EMD represents the signal as an expansion of signal-dependent basis functions that are estimated using an iterative process called sifting. This is in contrast to conventional decomposition techniques like wavelets, which perform the analysis by projecting the signal under consideration onto a number of predefined basis vectors [5].

Despite several efforts to better understand how EMD functions or to improve its efficiency (see, for instance), EMD still lacks a good mathematical theory and is mostly explained by an algorithm. However, it has found a wide range of various applications including biomedical, watermarking, and audio processing to mention a few, partly because it is easily and directly applicable and partly because it frequently produces interesting and valuable decomposition outputs.

3.3.2 Theoretical Background

Huang et al [6] established empirical mode decomposition (EMD) for breaking down a given signal x(t) into a limited number of subcomponents known as Intrinsic Mode Functions (IMFs). The IMFs depict a system's oscillatory behaviour.

A specific signal is acquired through a methodical procedure known as sifting. An IMF must meet the next two requirements [4].

1) A maximum of one should separate the number of extrema from the number of zero crossings.
2) The mean of the envelopes produced by the maxima and minima at any given place should be 0.

Below is the algorithm that will be used to perform the next iteration of sifting on the signal x(t) that has been provided.

The algorithm for performing sifting on a given signal x(t) is given below.

i. Identify all the maximas and minimas of x(t).
ii. Interpolate between minima, ending up with a signal $x_{min}(t)$ and similarly between maximas to give $x_{max}(t)$
iii. Calculate the average between those two envelopes:
    a. $x_{avg}(t) = (x_{max}(t) + x_{min}(t))/2$ (1)
iv. Extract the detail:
    a. $d1(t) = x(t) - x_{avg}(t)$. d1(t) is given as input to the next iteration of sifting.

In order to guarantee that the IMF component retains enough physical sense of both amplitude and frequency modulation, a stopping limit to the number of sifting rounds is used. This is accomplished by reducing the Standard Deviation (SD) between the outcomes of two successive iterations of filtering. The SD is given by if k number of sifting iterations are made:

$$SD = \sum_{t=0}^{L-1} \frac{|d_{k-1}(t) - d_k(t)|^2}{d_{k-1}^2(t)}$$

Typically, the value of SD is set between 0.2 and 0.3. Once dk(t) is accepted as first IMF, h1(t), the residue is calculated as,

$$r_1(t) = x(t) - d_k(t)$$

$$h_1(t) = d_k(t)$$

The input for the subsequent round of filtering, which will extract the second IMF, is r1(t). When the residue r(t) turns into a monotone function from which no more IMF can be derived, the EMD process can be terminated.

If N rounds of sifting process is performed on the given signal x(t), it will be decomposed to a set of N IMFs and a residue signal which can be denoted as,

$$x(t) = \sum_{k=1}^{N} h_k(t) + r_N(t)$$

The equation above demonstrates how an EMD-decomposed signal can be easily rebuilt by simply adding the IMF components $h_k(t)$ and the residue signal $r_N(t)$. Below is an illustration of how a signal is broken down using EMD.

A sample noisy sinewave of input SNR =15dB has been decomposed using the EMD algorithm whose individual IMF’s and residue signal are as shown in the figure below

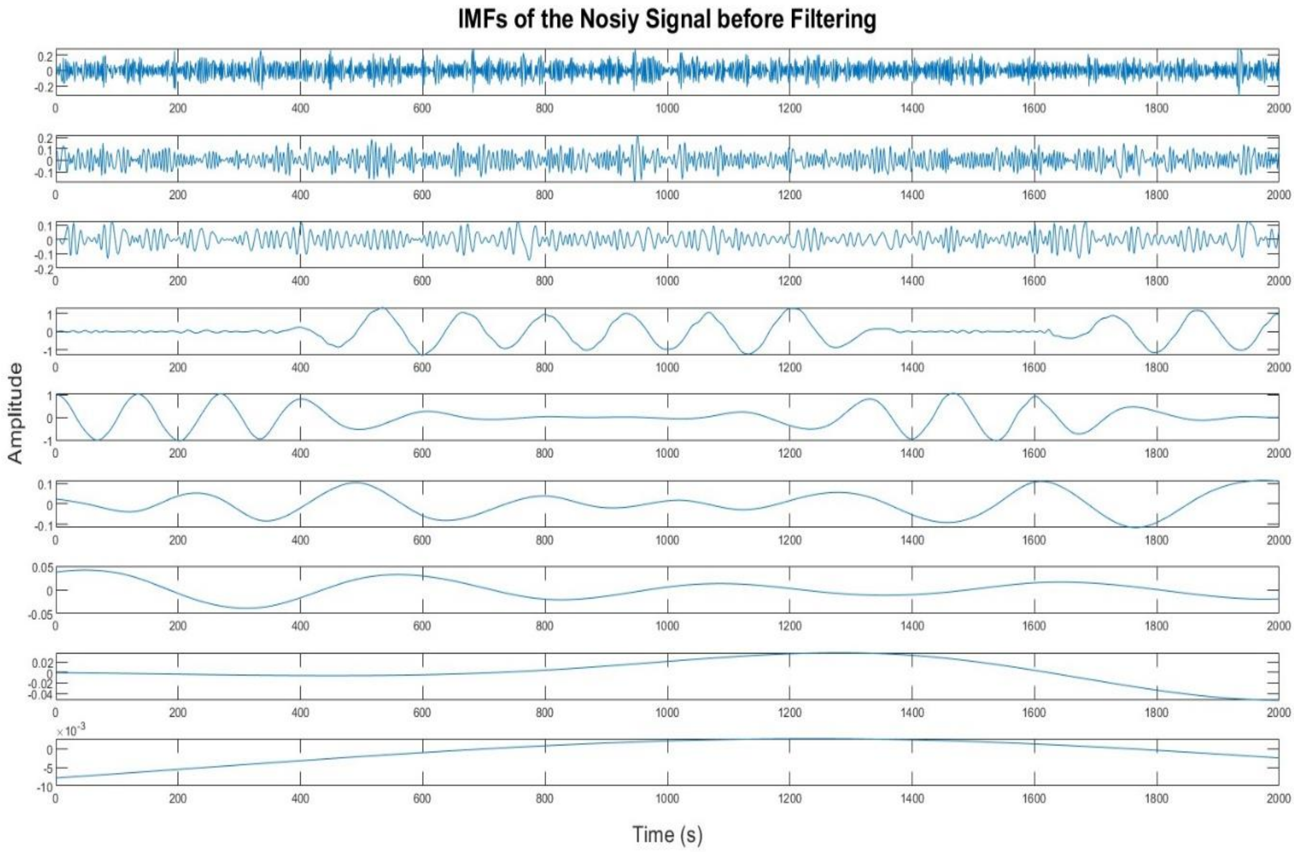

Figure 9 IMF's of the Noisy Signal before Filtering

3.3.3 Proposed method for EMD Subtraction based approach

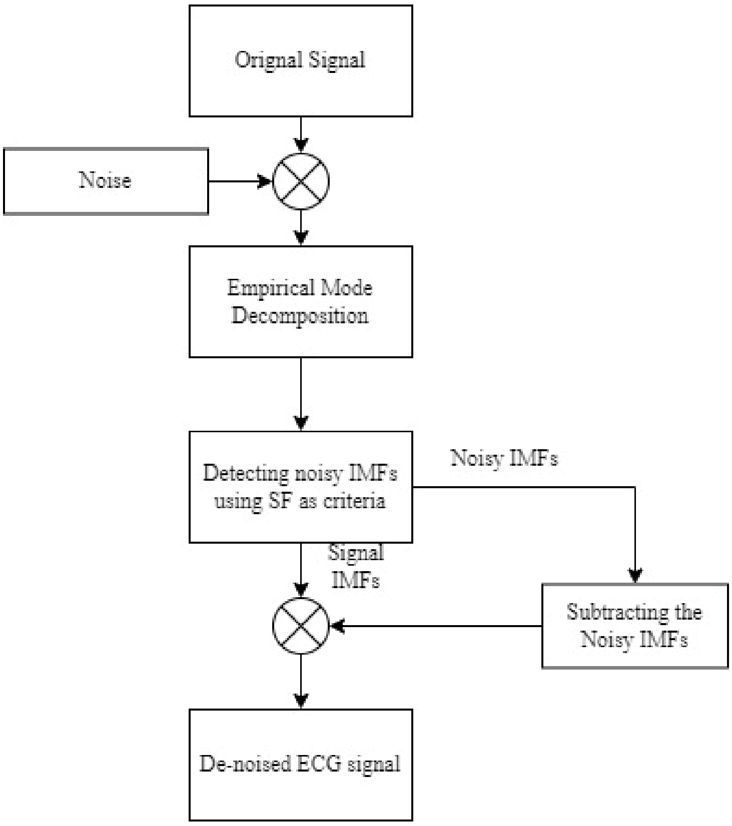

Figure 10 Flowchart depicting The EMD subtraction based denoising algorithm

Step 1: The signals are obtained from a power system (or) a random audio sample (or) a sinewave

S(t) = x(t) + n(t), where x(t) is the original ECG and n(t) is the noise signal, yields the noisy signal.

Step 2: Using the EMD approach, the noisy signal is divided into IMFs.

Step 3: By comparing the Spectral Flatness (SF) of each IMF to a threshold T, it is possible to determine the number of noisy IMFs, n. The geometric mean of the power spectrum divided by its arithmetic mean is used to compute the spectral flatness. It is given by,

$$Spectral\ Flatness = \frac{\sqrt[L]{\prod_{n=0}^{L-1} H(n)}}{\frac{\sum_{n=0}^{L-1} H(n)}{L}}$$

The first n IMFs are regarded as noisy IMFs if their Spectral Flatness is higher than the threshold T. The illustration below explains this procedure. Depending on the input signal chosen, we iterate through multiple values of possible spectral flatness values to arrive at the most optimal threshold for a given input SNR.

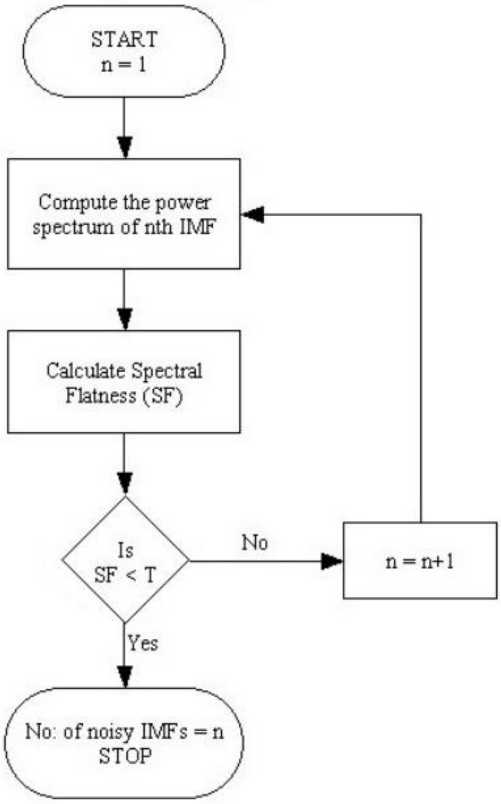

Figure 11 Flowchart depicting EMD thresholding algorithm

Step 4: The first few IMFs containing noise can be removed using the spectral flatness criteria as per the flowchart shown above.

Step 5: The signal can be reconstructed after combining the noise free IMF's. The reconstructed signal is given by

$$\hat{x}(t) = \sum_{k=1}^{n} \tilde{h}_k(t) + \sum_{k=n+1}^{N} h_k(t) + r_N(t)$$

where $\tilde{h}_k$(t) is the filtered version of $h_k(t)$

We implemented the above algorithm on MATLAB for 4 different input signals as in the previous case and plotted the original signal, the signal after adding white gaussian noise (SNR = 15dB) and the denoised output signal as shown in the figures below.

Input 1 – Frequency readings from a power system taken at 30 samples/second for 180 seconds.

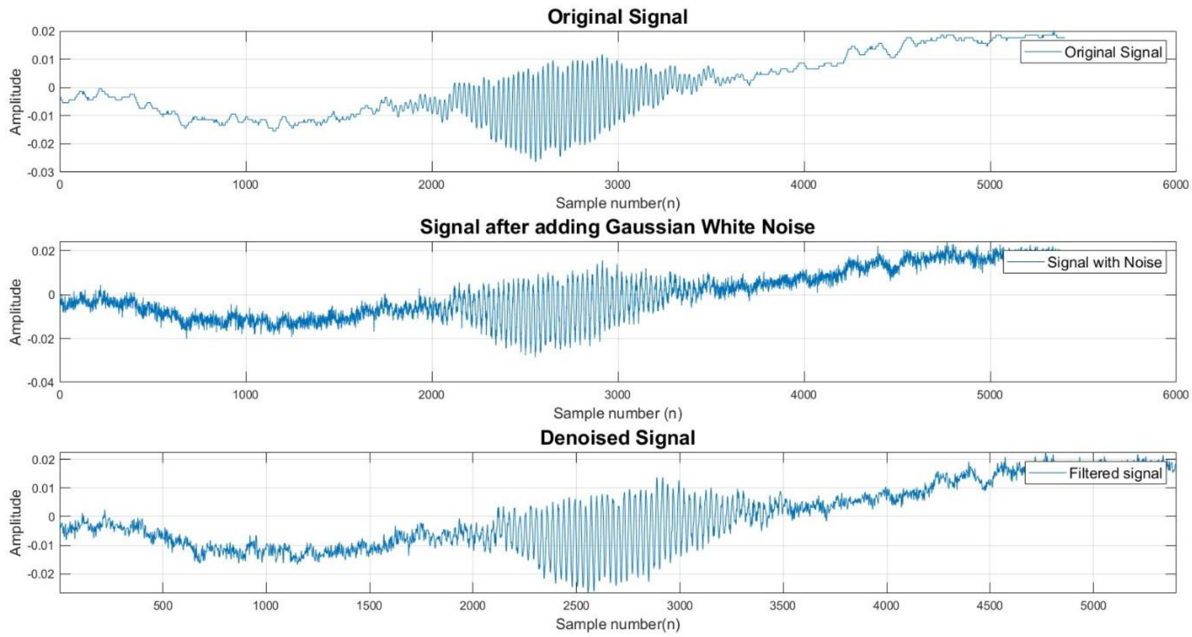

Figure 12 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 18.3184 dB)

Input 2 – Sinewave of frequency 60Hz

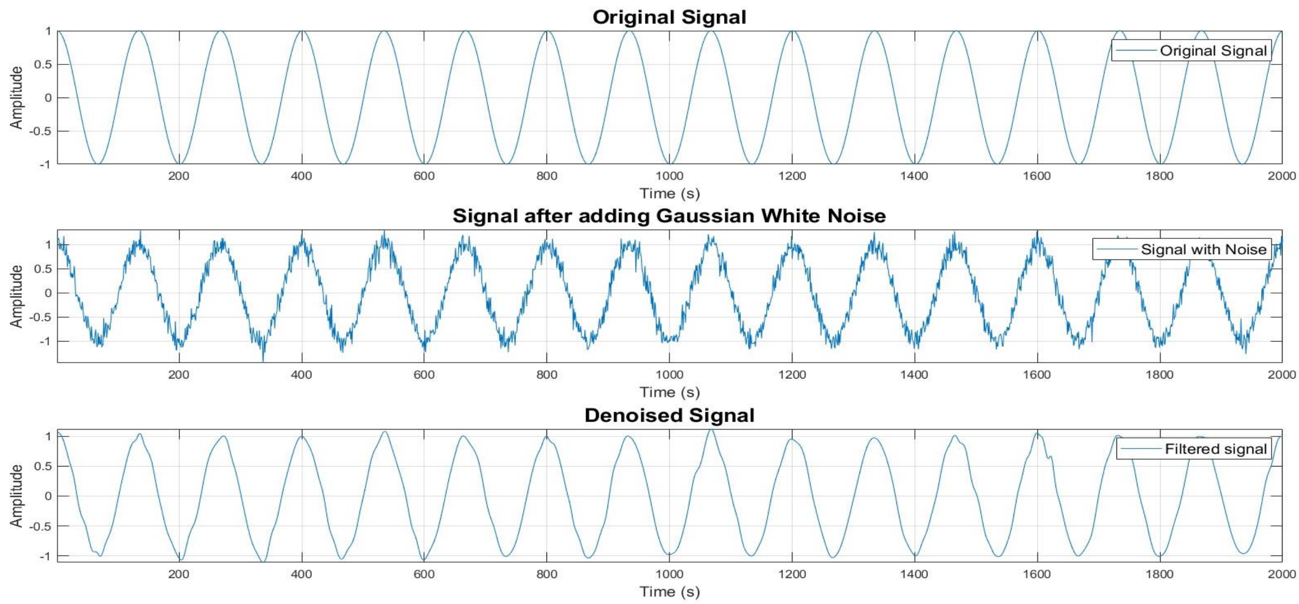

Figure 13 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 24.8402)

Input 3 – Random audio signal

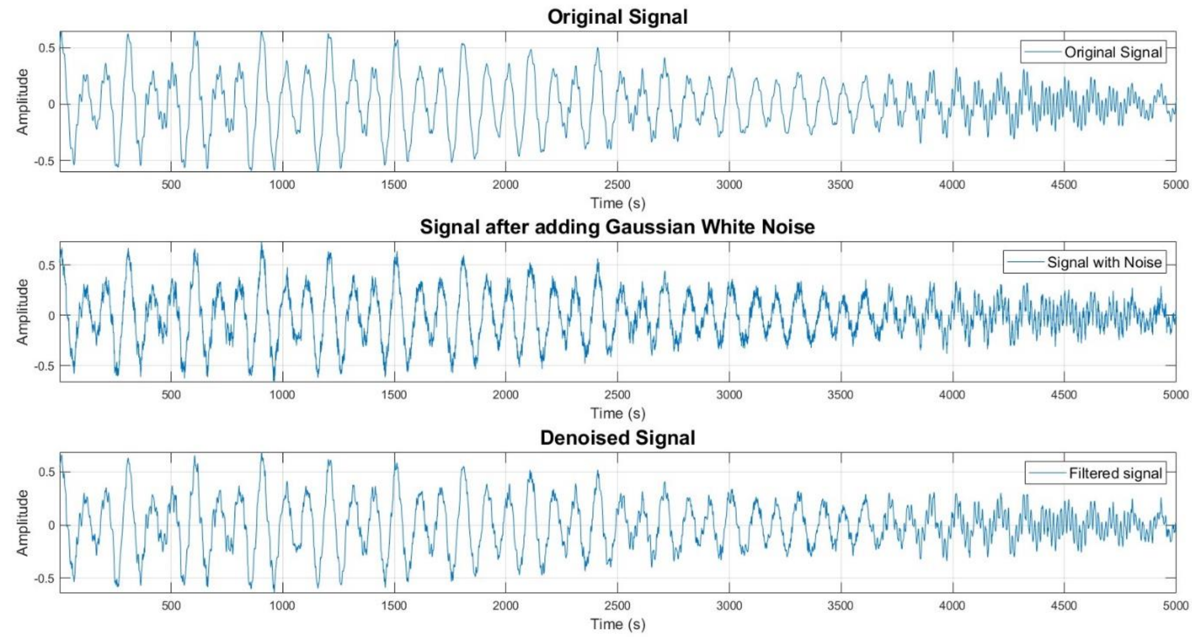

Figure 14 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 18.2948 dB)

Input 4 – Frequency readings from a power system taken at 30 samples/second for 180 seconds.

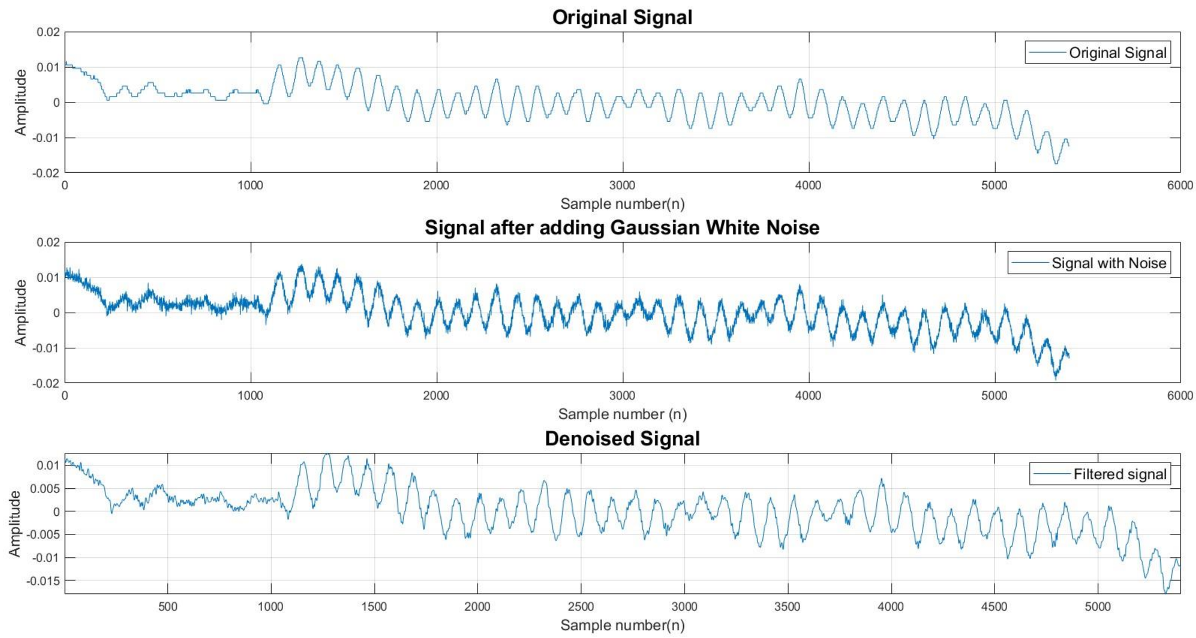

Figure 15 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 20.0316 dB)

3.3.4 Proposed Method for EMD Filter based approach

When a noisy signal is decomposed using EMD, the initial IMFs are primarily where the noise components are found. In this study, the Spectral Flatness (SF) measure is used to determine whether or not a specific IMF is noise-dominated. In contrast to signal IMFs, the spectrum of noisy IMFs will be rather flat.

The proposed noise removal method using EMD is illustrated below,

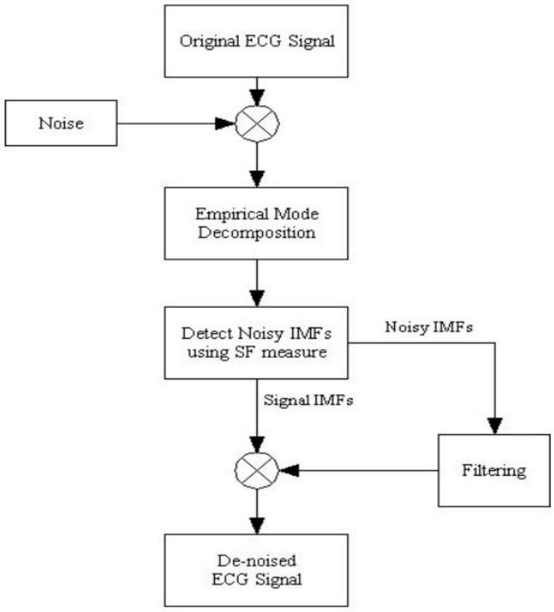

Figure 16 Flowchart depicting EMD filter based denoising algorithm

Step 1: The signals are obtained from a power system (or) a random audio sample (or) a sinewave

S(t) = x(t) + n(t), where x(t) is the original signal and n(t) is the noise signal, yields the noisy signal.

Step 2: Using the EMD approach, the noisy signal is divided into IMFs.

Step 3: By comparing the Spectral Flatness (SF) of each IMF to a threshold T, it is possible to determine the number of noisy IMFs, n. The geometric mean of the power spectrum divided by its arithmetic mean is used to compute the spectral flatness. It is given by,

$$Spectral\ Flatness = \frac{\sqrt[L]{\prod_{n=0}^{L-1} H(n)}}{\frac{\sum_{n=0}^{L-1} H(n)}{L}}$$

The first n IMFs are regarded as noisy IMFs if their Spectral Flatness is higher than the threshold T. The illustration below explains this procedure. Depending on the input signal chosen, we iterate through multiple values of possible spectral flatness values to arrive at the most optimal threshold for a given input SNR.

Step 4: The IMFs that do not satisfy the threshold (spectral flatness criterion) is filtered using a lowpass butterworth filter of order 5 with a cut-off frequency of suitable value obtained through ad-hoc analysis of the input signal.

The figure below shows the sample sine wave that was decomposed into its individual IMF’s and residue signals as seen in figure 5 filtered with a IIR butterworth low pass filter of cut-off frequency 60Hz to depict the filtering of IMF’s that don’t satisfy the threshold criterion.

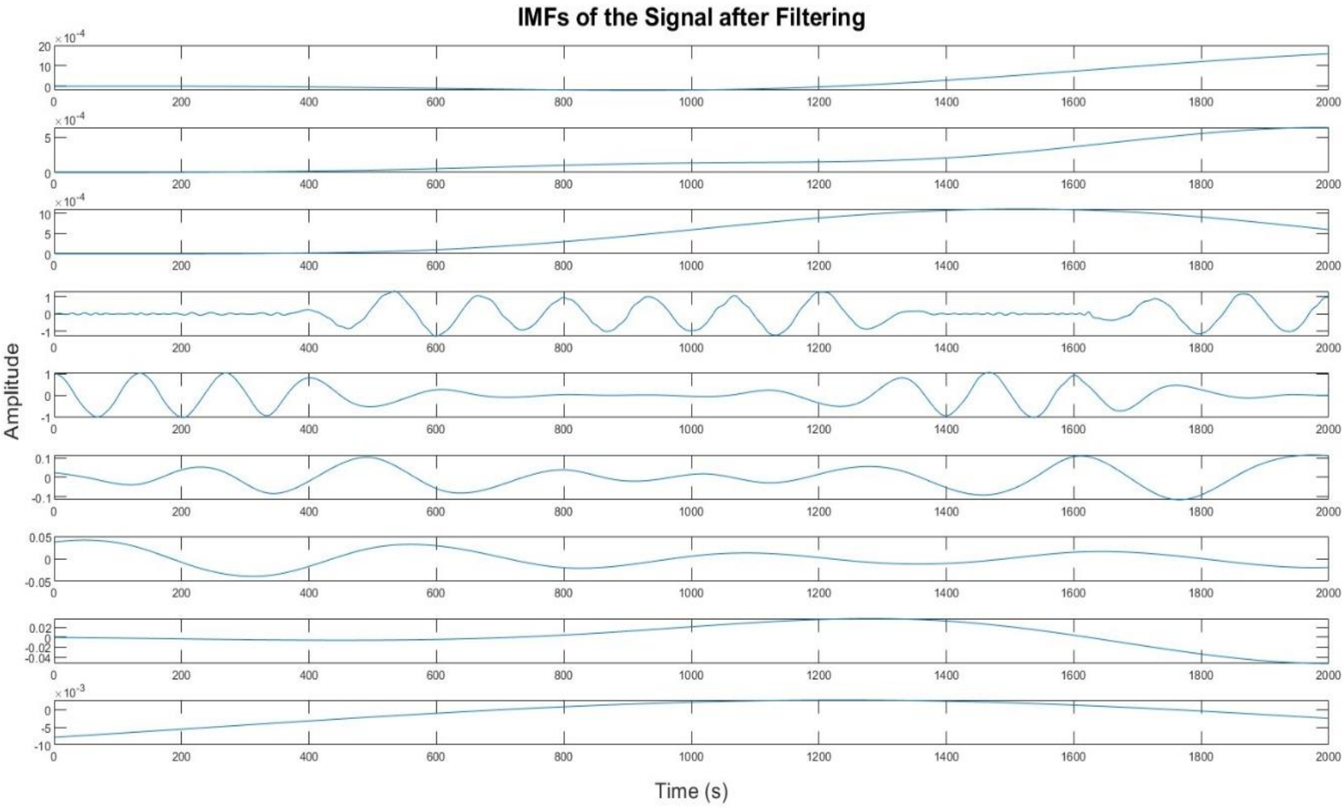

Figure 17 IMF's of the signal after filtering

Here we notice that the top 3 IMF's that failed to meet the thresholding criterion have been passed through the filter to remove the noisy content.

Step 5: the filtered IMFs and the remaining signal IMFs are combined to reconstitute the signal. The reconstructed signal is obtained by the equation

$$\hat{x}(t) = \sum_{k=1}^{n} \tilde{h}_k(t) + \sum_{k=n+1}^{N} h_k(t) + r_N(t)$$

where $\tilde{h}_k$(t) is the filtered version of $h_k(t)$

The following MATLAB functions have been utilized for EMD based decomposition and reconstruction to denoise the input signals:

imf= emd(a); (EMD function)

This function performs empirical mode decomposition of the given signal which is *a* in our case and will store the obtained IMFs into the variable *imf* [13].

[y, x] = butter(5, F/(Fs/2), 'low');(butterworth function)

This function designs a lowpass butterworth filter as it has been mentioned as a parameter . The resulting bandpass and bandstop designs are of order 2n [12].

new_imf(:,i) = filter(y, x, imf(:,i)')';

This function filters the input signal that is *imf(:,i)'* using a rational transfer function [9].

We implemented the above algorithm on MATLAB for 4 different input signals as in the previous case and plotted the original signal, the signal after adding white gaussian noise (SNR = 15dB) and the denoised output signal as shown in the figures below.

Input 1 – Frequency readings from a power system taken at 30 samples/second for 180 seconds.

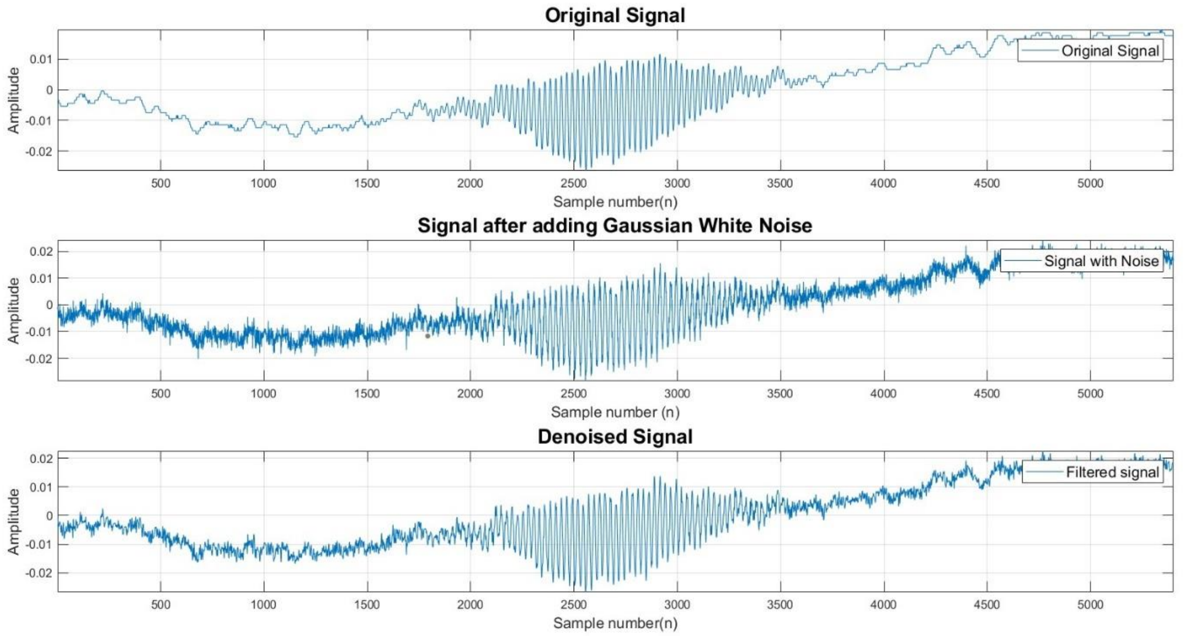

Figure 18 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 18.3182 dB)

Input 2 – Sinewave of frequency 60Hz

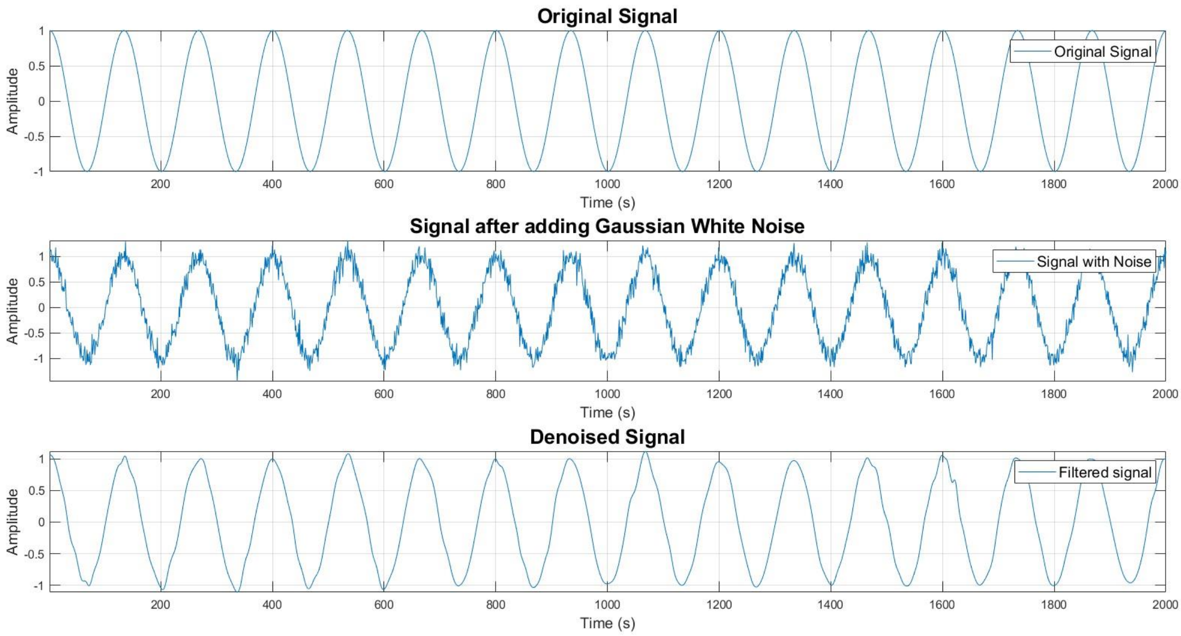

Figure 19 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 24.8620 dB)

Input 3 – Random audio signal

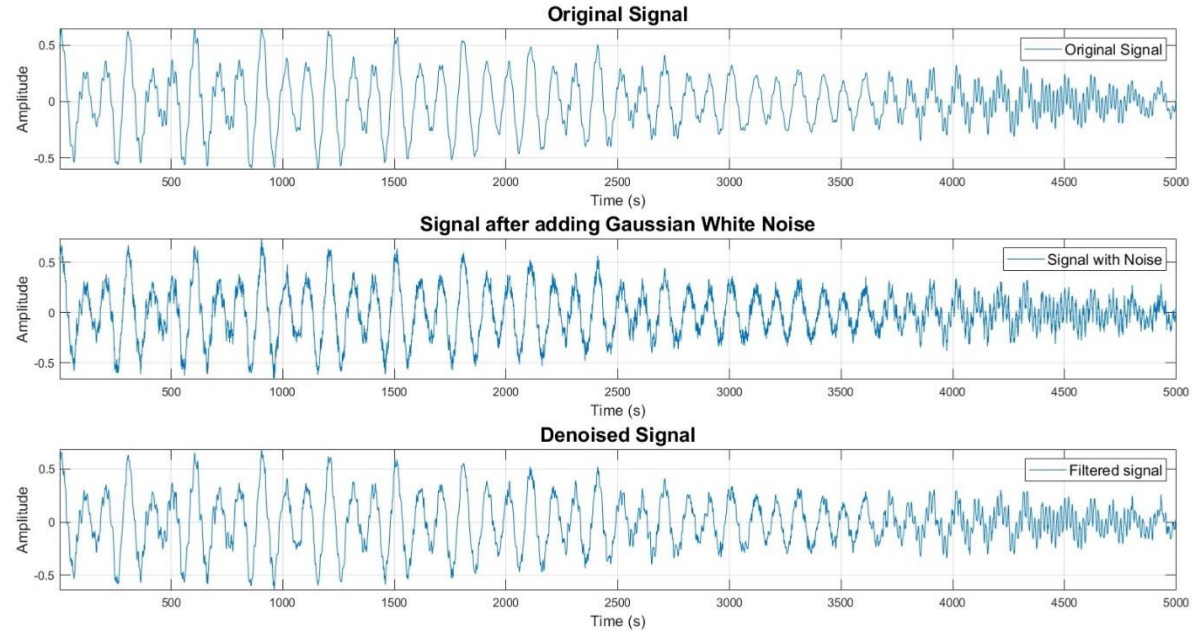

Figure 20 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 18.2944 dB)

Input 4 – Frequency readings from a power system taken at 30 samples/second for 180 seconds.

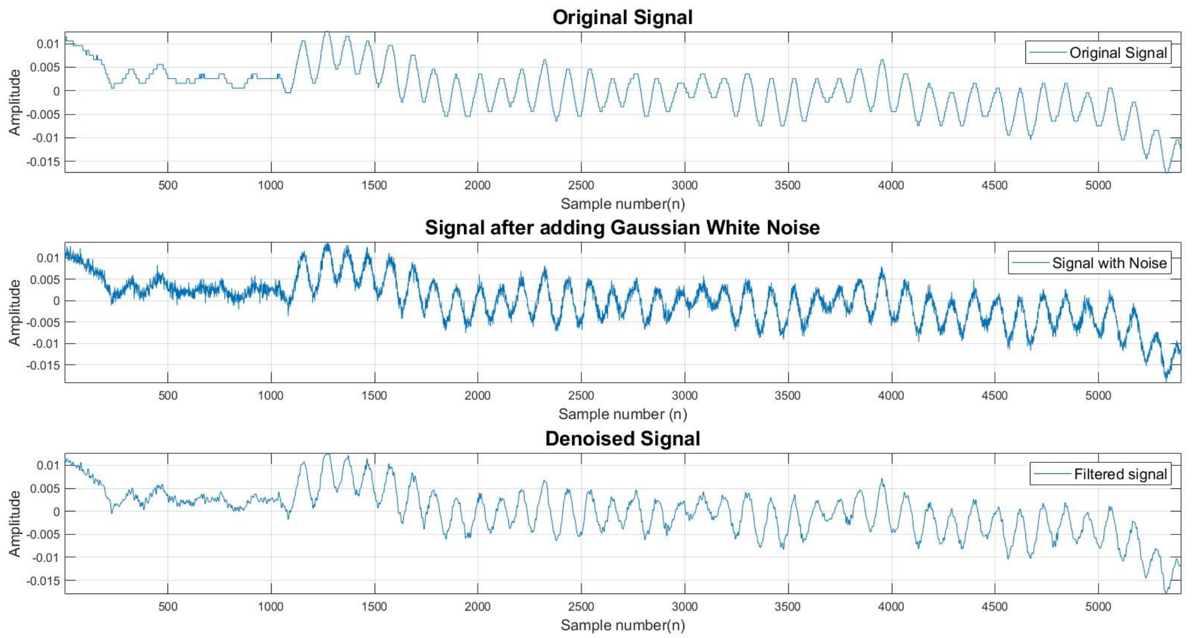

Figure 21 Plots depicting original signal, white gaussian noise added to original signal and denoised output signal respectively (Output SNR: 20.0259 dB)

4. OBSERVATIONS AND RESULTS

Based on the Signal to Noise Ratio (SNR) the effectiveness of the above-mentioned algorithms for denoising have been assessed. The following is a representation of the SNR:

where x(t) is the signal and y(t) is the noise.

$$SNR_{in} = 10log_{10}\frac{\sum_{t=1}^{T} x(t)^2}{\sum_{t=1}^{T}(y(t)-x(t))^2}$$

For SNR_{output} :

$$SNR_{out} = 10log_{10}\frac{\sum_{t=1}^{T} x(t)^2}{\sum_{t=1}^{T}(x(t)-\bar{x}(t))^2}$$

Here $\overline{x(t)}$ is reconstructed signal.

Given below are tables depicting the performance of each of the 3 algorithms for various values of input SNR values in dB scale for each of the 4 input signal cases.

Input sample 1: f_case3_fps_30_180s

Table 1 Performance of 3 algorithms for Input 1

	EMD subtraction based	EMD filter based	Wavelet based
Input SNR value (dB)	Output SNR value (dB)	Output SNR value (dB)	Output SNR value (dB)
5	11.1582	11.1559	13.3733
10	15.8903	15.8895	17.3718
15	18.3184	18.3182	20.8425
25	27.7387	27.7368	29.3126
35	35.0678	35.0678	34.6033
45	45.0678	45.0678	36.8406

Note: Input sample 2: f_case1_fps_30_180s

Table 2 Performance of 3 algorithms for Input 2

	EMD subtraction based	EMD filter based	Wavelet based
Input SNR value (dB)	Output SNR value (dB)	Output SNR value (dB)	Output SNR value (dB)
5	11.2279	11.2590	16.4111
10	18.1147	18.1018	20.2425
15	20.0316	20.0259	22.8226
25	25.7388	25.7378	26.0104
35	35.0678	35.0678	29.5427
45	45.0678	45.0678	30.1253

Note: Input sample 3: sinewave (F = 60Hz)

Table 3 Performance of 3 algorithms for Input 3

	EMD subtraction based	EMD filter based	Wavelet based
Input SNR value (dB)	Output SNR value (dB)	Output SNR value (dB)	Output SNR value (dB)
5	18.9494	18.9028	18.2565
10	20.2040	20.2114	22.9292
15	24.8402	24.8620	27.8395

25	34.3362	34.3477	36.7675
35	41.6016	41.6005	44.2299
45	48.6556	48.6554	53.7922

Note: Input sample 4: Random Audio sample

Table 4 Performance of 3 algorithms for Input 4

	EMD subtraction based	EMD filter based	Wavelet based
Input SNR value (dB)	Output SNR value (dB)	Output SNR value (dB)	Output SNR value (dB)
5	11.2534	11.2541	12.1875
10	15.6131	15.6035	14.9244
15	18.2948	18.2944	20.5533
25	28.0803	28.0801	27.9659
35	35.0176	35.0176	37.1497
45	45.0176	45.0176	42.8150

4.1 General observations and comparisons

The above tables illustrate how the each of the algorithms perform for varying input SNR values for each of the 4 different input signals. In case of the wavelet approach we notice that it performs considerably better than its EMD counterpart for the lower input SNR values and doesn’t should much improvement rather the quality deteriorates as and when we increase the input SNR values further. The reason for this is mentioned later.

However, in case of the EMD based denoising approach we can observe that although it performs comparatively worse than the wavelet at lower input SNR values, but it is able to maintain the signal integrity at higher SNR values unlike the wavelet.

A possible reason for this could be that, as and when we increase the value of input SNR to values greater than 35 dB, the ratio of signal to noise in the linear scale is in the order of 104 meaning the amount of noise compared to that of the original signal is very negligible because of which the algorithm chooses not to remove or filter any of the IMF's during the thresholding process.

One another observation, is that the difference between the output SNR values of the two EMD based approaches namely the EMD subtraction based & EMD filter based is very minimal. This is because, the IMF's being filtered or removed in each respective case are the same as seen in the previous section. As seen in fig we notice that upon filtering the higher noisy IMF's, the remaining signal is just a simple curve which even removal (subtraction), doesn't alter the original signal structure to a significant extent.

The plot below depicts how beyond a particular input SNR value the performance of the wavelet drops as compared to the EMD based approach

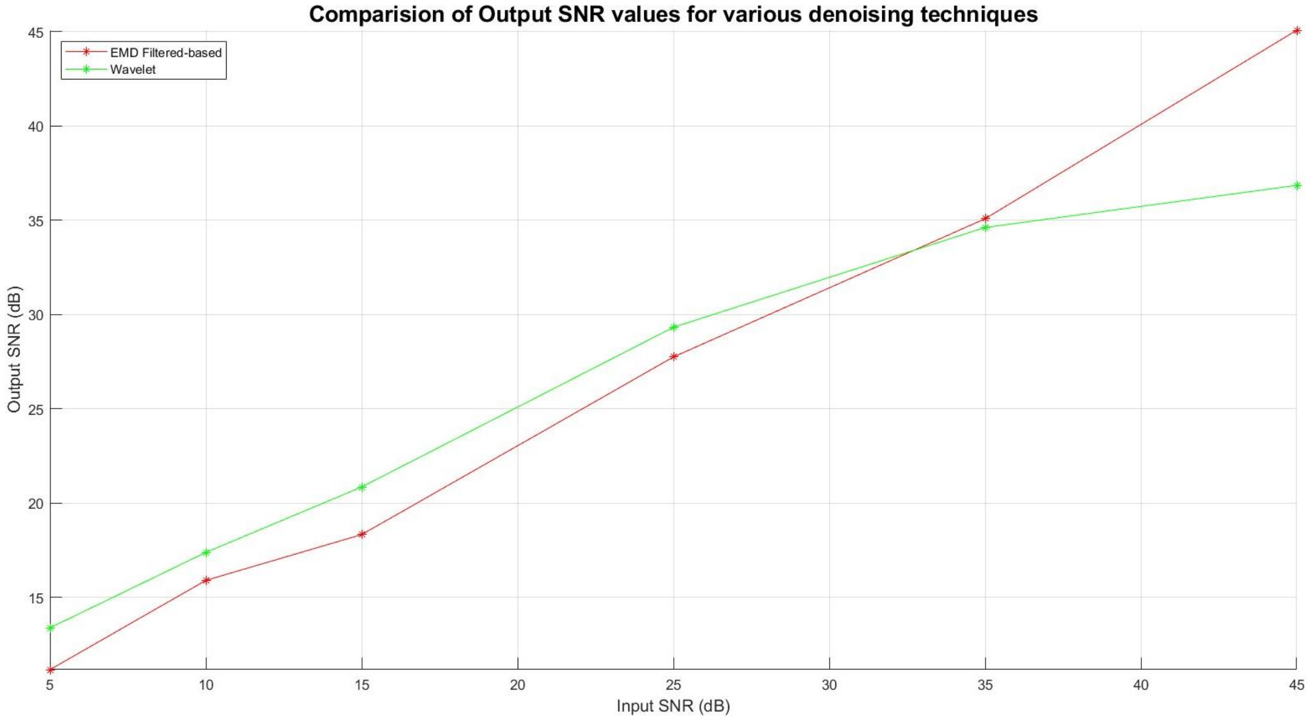

Figure 22 Comparison of Output SNR values for various denoising techniques

4.2 Observations for wavelet

Following the application of wavelet-based denoising to each of the four input signals and considering SNR as a metric, it can be seen that, when the nature of the input signal is known, wavelet transform is a very effective method for denoising the signal. Additionally, wavelet smoothens the input signal somewhat more effectively than other denoising techniques like EMD.

Other intriguing findings include the fact that wavelet-based denoising tends to perform comparatively worse when the input signal has a high SNR value (> 25dB).

[8] also mentions that the maximum level to apply the wavelet transform depends on how many data points contain in a data set, since there is a down-sampling by 2 operation from one level to the next one. In

their experience, one factor that affects the number of levels we can reach to achieve the satisfactory noise removal results is the signal-to-noise ratio (SNR) in the original signal. Generally, the measured signals from the PZT sensors have higher SNR than that of the measured signals from fiber optic sensors. So to process the PZT data, they need more level of wavelet transform (e.g. 12) to remove most of its noise. For the fiber optic sensor data, they could only go up to 4 or 5 level otherwise they would end up removing much of the base signal, therefore the base signal spikes wouldn't be captured. This is true for all similar decomposition approaches since some of the original information is lost when noise is filtered out of a signal that was originally clean.

In the wavelet transform, the output SNR is dependent on the level of decomposition. Increasing the levels of decomposition generally increases the computational complexity of the wavelet denoising algorithm and this does not give any reasonable improvement in signal quality too. This paper [7]. also mentions that there is a limit to extend the number of stages or level of decomposition for the transform. After a certain limit the performance of the system degrades. Therefore, optimal value for number of levels should be selected while removing noise from noisy signals. This can be seen in the table and supporting graph below, which shows that for a given SNR, the output SNR grows up to a maximum level before beginning to decline.

The following graph and table illustrate these observations,

Input sample: f_case3_fps_30_180s

Table 5 Shows variation of output SNR w.r.t level of decomposition

Level of Decomposition	**SNR (dB)**		
	5	15	35
1	8.0918	18.0438	34.6033
2	11.0299	20.8425	33.3256
3	13.3733	20.1935	22.5505
4	9.0377	9.6985	9.7777
5	8.6454	8.9029	8.9320
6	8.6838	8.7927	8.8049
7	8.5523	8.6099	8.6170
8	8.4824	8.5236	8.5297

Table showing output SNR values of denoised signal for different levels of decomposition (SNR of Noisy wave = 5dB, 15dB & 35dB)

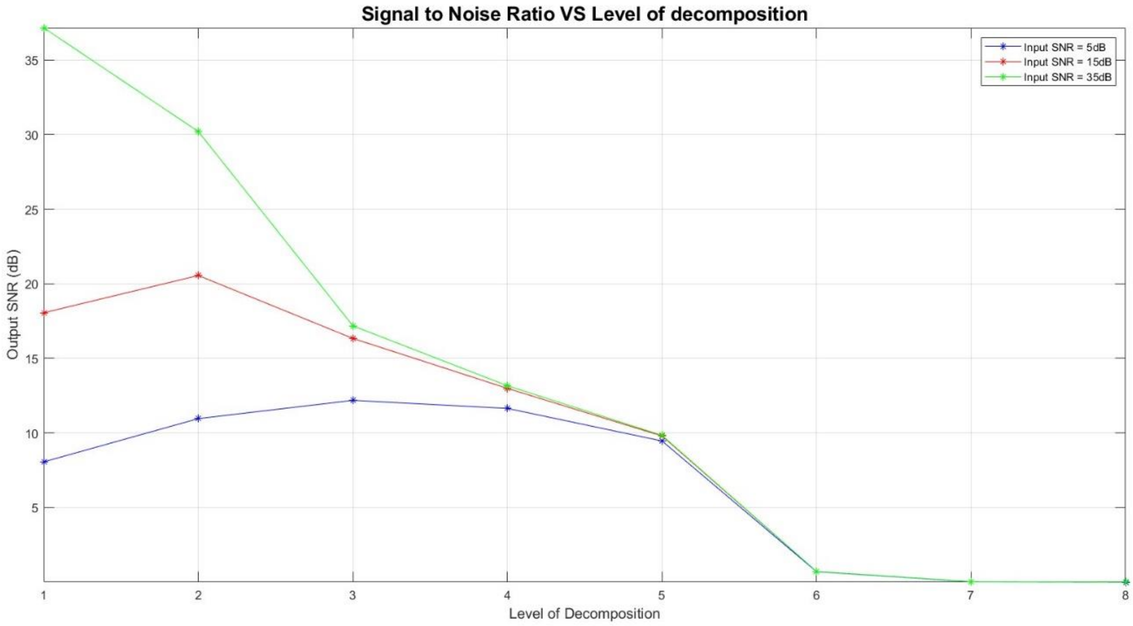

Figure 23 Performance of wavelet with varying level of decomposition

5. FUTURE SCOPE

In the coming semester we plan on working on FFT & IFFT based approach for denoising. Using this method we plan on taking FFT of noisy signal, examine it's amplitudes in frequency domain and identifying to the noise part using an ad-hoc approach.

We also plan to use a few machine learning based techniques for denoising real time signals. This will be done using feature extraction for comparison between the noisy and denoised signal. We also plan on incorporating neural networks for the same.

Perform a comparative analysis of the various algorithms implemented so far and yet to be implemented on the basis of computational speed.

Examine the occurrence of the different types of coloured noises and their real-tie denoising applications

Examine the reconstruction of the decomposed signal after wavelet transform to understand on what basis are the coefficients retained or removed during reconstruction. After looking into the forward process of decomposition of wavelet into two coefficients using MATLAB, we plan to understand the thresholding process and the process behind reconstruction functions used in MATLAB. Hence, we plan to explore the underlying code and understand the reconstruction process which might shed some light as to why wavelet performs inadequately for higher SNR values.

6. REFERENCES

[1] Y. Wei and X. Fan, ‘De-noising the Speech Signal With FIR Filter Based on Matlab’, in *Proceedings of the 2016 4th International Conference on Electrical & Electronics Engineering and Computer Science (ICEEECS 2016)*, Jinan, China, 2016. doi: 10.2991/iceeecs-16.2016.205.

[2] S. Saxena, ‘REVA UNIVERSITY Bangalore , India sugandhasxn@gmail.com’, p. 7.

[3] G. Chen, W. Xie, and Y. Zhao, ‘Wavelet-based denoising: A brief review’, in *2013 Fourth International Conference on Intelligent Control and Information Processing (ICICIP)*, Jun. 2013, pp. 570–574. doi: 10.1109/ICICIP.2013.6568140.

[4] A. Chacko and S. Ari, ‘Denoising of ECG signals using Empirical Mode Decomposition based technique’, in *IEEE-International Conference On Advances In Engineering, Science And Management (ICAESM -2012)*, Mar. 2012, pp. 6–9.

[5] Y. Kopsinis and S. McLaughlin, ‘Development of EMD-Based Denoising Methods Inspired by Wavelet Thresholding’, *IEEE Trans. Signal Process.*, vol. 57, no. 4, pp. 1351–1362, Apr. 2009, doi: 10.1109/TSP.2009.2013885.

[6] A. J. Quinn, V. Lopes-Dos-Santos, D. Dupret, A. C. Nobre, and M. W. Woolrich, ‘EMD: Empirical Mode Decomposition and Hilbert-Huang Spectral Analyses in Python’, *J. Open Source Softw.*, vol. 6, no. 59, p. 2977, Mar. 2021, doi: 10.21105/joss.02977.

[7] ‘11-184-144136129024-29.pdf’. Accessed: Nov. 17, 2022. [Online]. Available: http://pep.ijieee.org.in/journal_pdf/11-184-144136129024-29.pdf
[8] ‘Chapter4.pdf’. Accessed: Nov. 17, 2022. [Online]. Available: https://vtechworks.lib.vt.edu/bitstream/handle/10919/35983/Chapter4.pdf
[9] ‘1-D digital filter - MATLAB filter - MathWorks India’. https://in.mathworks.com/help/matlab/ref/filter.html (accessed Nov. 17, 2022).
[10] ‘1-D wavelet decomposition - MATLAB wavedec - MathWorks India’. https://in.mathworks.com/help/wavelet/ref/wavedec.html (accessed Nov. 17, 2022).
[11] ‘Add white Gaussian noise to signal - MATLAB awgn’. https://www.mathworks.com/help/comm/ref/awgn.html (accessed Nov. 17, 2022).
[12] ‘Butterworth filter design - MATLAB butter - MathWorks India’. https://in.mathworks.com/help/signal/ref/butter.html (accessed Nov. 17, 2022).
[13] ‘Empirical mode decomposition - MATLAB emd - MathWorks India’. https://in.mathworks.com/help/signal/ref/emd.html (accessed Nov. 17, 2022).
[14] ‘Reconstruct single branch from 1-D wavelet coefficients - MATLAB wrcoef - MathWorks India’. https://in.mathworks.com/help/wavelet/ref/wrcoef.html (accessed Nov. 17, 2022).