



Digital Filters

BM4151 – Biosignal Processing

Nuwan Sriyantha Bandara
180066F

DEPARTMENT OF ELECTRONIC AND TELECOMMUNICATION
ENGINEERING, UNIVERSITY OF MORATUWA, SRI LANKA

November 30, 2022

Contents

1 Smoothing Filters	1
1.1 Moving Average Filters	1
1.1.1 Implementation	3
1.2 Savitzky-Golay Filters	7
2 Ensemble Averaging	9
2.1 MSE variation in ensemble averaging	11
2.2 Improvement of SNR in ensemble averaging	11
2.3 Implementation of ensemble average on signals with repeating patterns .	12
3 FIR derivative filters	15
3.1 Derivative-based filters	15
3.2 FIR derivative filter application	17
4 Designing FIR filters using windows	19
4.1 Improving the windowing function	20
4.2 FIR filter design and application using Kaiser window	21
4.3 Calculation of filter parameters in Kaiser method	22
5 IIR filters	27
5.1 IIR implementation of ECG filters	28
5.2 Comparison between IIR and FIR filters	31

1 Smoothing Filters

Finite impulse response (FIR) filters are a type of digital filter with a finite impulse response in the time domain as the name represents itself. They are non-recursive with no internal feedback (thus, settle down to zero after a sudden impulse) and always stable in operation.

The general formula for FIR filters is:

$$y(n) = \sum_{k=0}^M b_k \times x(n - k) \quad (1)$$

$$H(z) = \sum_{k=0}^M b_k \times z^{-k} \quad (2)$$

where M and b_k are the filter order and the filter coefficients respectively.

It is typically needed to smoothen the data by reducing high-frequency noise from various sources in biomedical signal processing. Two of the most common such smoothing filters are moving average filters (MA(N) when N is the order of the filter) and Savitzky-Golay filters: SG(N,L) where N is the order of the filter and $L' = 2L + 1$ is the length of the filter.

1.1 Moving Average Filters

A MA(N) filter could be conveyed as a moving window across the signal and thus, a point of the filtered signal, $y(n)$ is derived from the average of the windowed points on the input signal $x(n)$. Therefore, $y(n)$ and $x(n)$ have the following relationship in an equally weighted MA filter of order M .

$$y(n) = \frac{1}{M+1} \sum_{k=0}^M x(n - k) \quad (3)$$

$$H(z) = \frac{1}{M+1} \sum_{k=0}^M z^{-k} \quad (4)$$

Here, the MA filter is applied to an electrocardiogram (ECG) signal with the sampling frequency of 500Hz. The typical ECG signal which was given is visualized below in which the characteristic regions of the ECG waveform: P wave, QRS complex and T wave could be apparently observed.

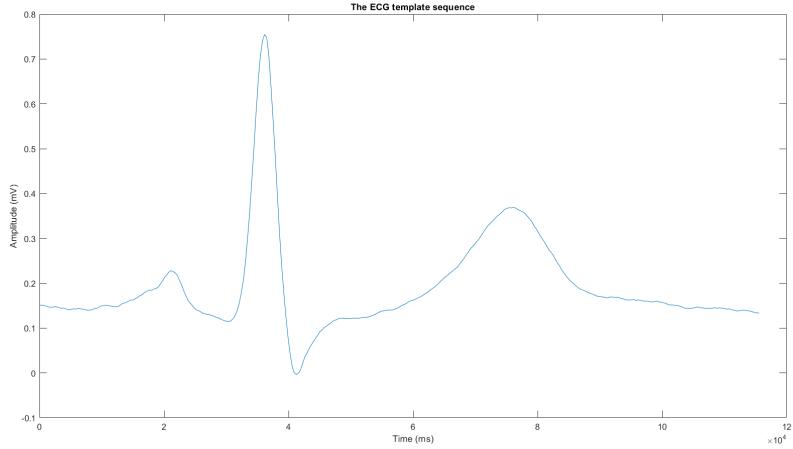


Figure 1: Typical ECG signal which was given in ECG_template.mat

In order to explore the performance of different filters, white Gaussian noise is first added to the sample ECG signal and then, compared the resulting signal (after applying the filter) with the sample signal.

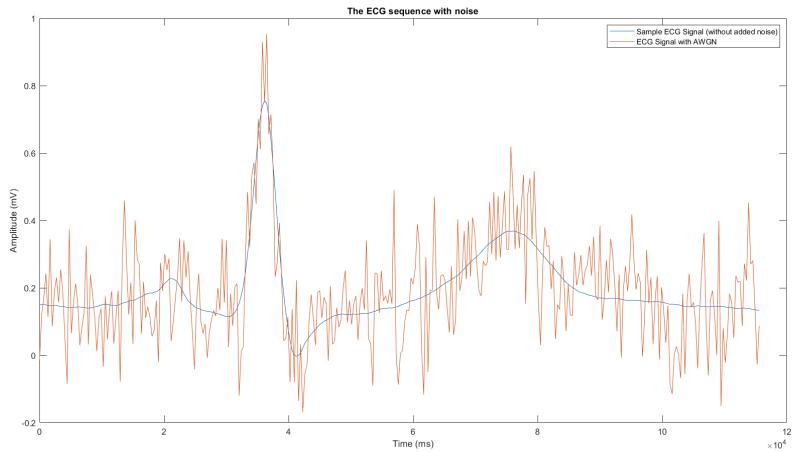


Figure 2: Sample ECG signal with added white Gaussian noise

Power spectral density (PSD) provides a better perspective on the effect of noise on the signal since it is an indicator of the power of the signal with respect to frequencies. The PSD graph is plotted on logarithmic scale in order to obtain a better visualization due to the vast difference of the noise power added (i.e. 5dB).

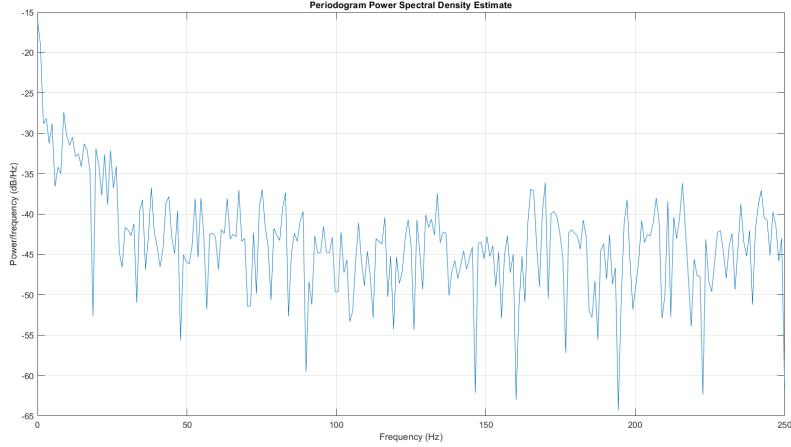


Figure 3: PSD of the noisy signal in linear scale

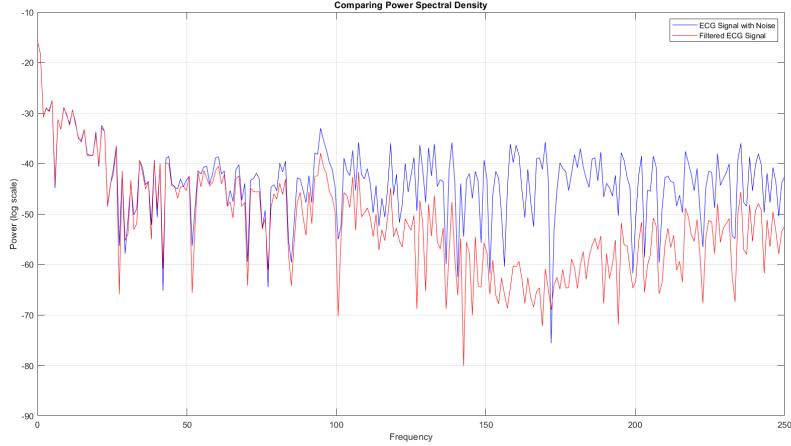


Figure 4: PSD of the noisy signal in logarithmic scale

Through the PSD plot, it is observable that the low frequencies are prominent in the sample signal's power while most of the power in the signal with noise lies within the higher frequencies. Therefore, a more desirable signal could be attained by eliminating the high-frequency power from the signal with noise.

1.1.1 Implementation

Group delay derivation Through 4, it could be extended that:

$$H(\omega) = \frac{1}{N} \sum_{k=0}^{N-1} e^{-jk\omega} \quad (5)$$

where the order of the filter is N . Thus,

$$H(\omega) = \frac{e^{-j\omega\frac{N-1}{2}}}{N} \sum_{k=-(N-1)/2}^{(N-1)/2} e^{-jk\omega} \quad (6)$$

But, $e^{jk\omega} = \cos(k\omega) + j\sin(k\omega)$, therefore, through 5,

$$H(\omega) = \frac{\cos((N-1)\omega/2 - j\sin((N-1)\omega/2)}{N} \sum_{k=-(N-1)/2}^{(N-1)/2} 2\cos(k\omega) \quad (7)$$

Then, the principle angle,

$$\text{Arg}(H(\omega)) = \tan^{-1}\left(\frac{\text{Im}H(\omega)}{\text{Re}H(\omega)}\right) = \tan^{-1}\frac{\cos((N-1)\omega/2)}{-\sin((N-1)\omega/2)} = \frac{(N-1)\omega}{2} + \pi \quad (8)$$

Then by definition for group delay(τ_g),

$$\tau_g = \frac{\partial \text{Arg}(H(\omega))}{\partial \omega} = \frac{N-1}{2} \quad (9)$$

Since the group delay of the filter is 1 sample, the filtered signal has to be hastened by that number of samples such that the filter and the input signal align as conveyed in the zoomed image. As it is observable, the high-frequency components are reduced in the filtered signal compared to the signal with noise. This reduction could be further verified by analyzing the PSD plot as well.

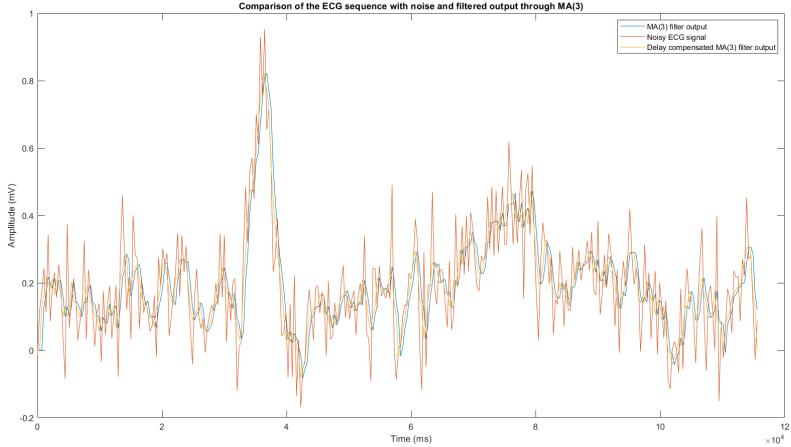


Figure 5: ECG sequence with noise and filtered (with delay compensation) using MA(3)

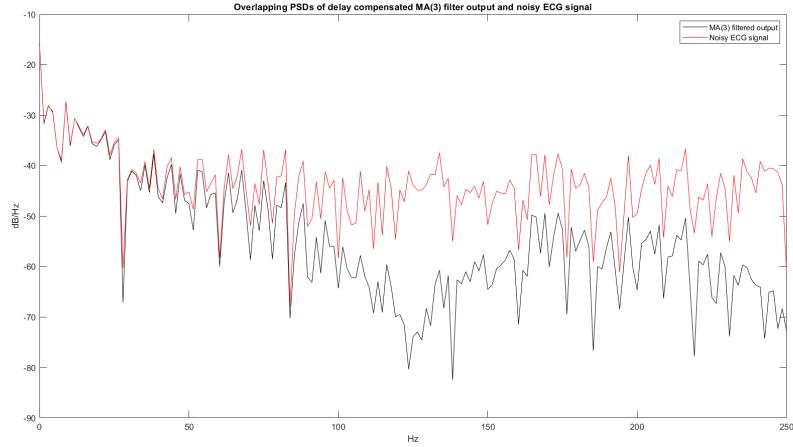


Figure 6: PSD of the ECG sequences with noise and filtered signal using MA(3)

Implementation using MATLAB *filter* In contrast, the *filter* function in MATLAB could be utilized to implement the filter in which the return is a delay-compensated filtered signal. The resultant filtered signal from a MA of order 3 (using *filter* function in MATLAB).

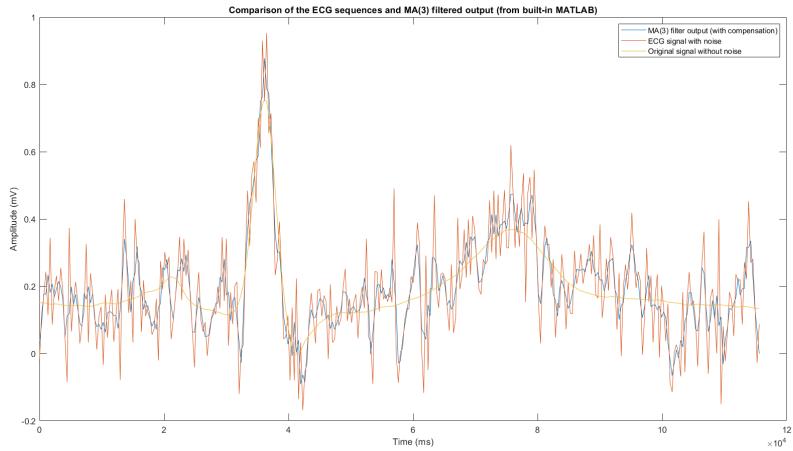


Figure 7: ECG sequence with noise and filtered (with delay compensation) using MATLAB in-built MA(3)

Implementation of a MA filter of order 10 In generic terms, the increment of the order of the filter implies that a larger number of the samples in the sample signal will be averaged, which thus, further redefines any changes that are considered sudden in the input signal. Therefore, more high-frequency noise would be removed as shown in the magnitude response of the filter.

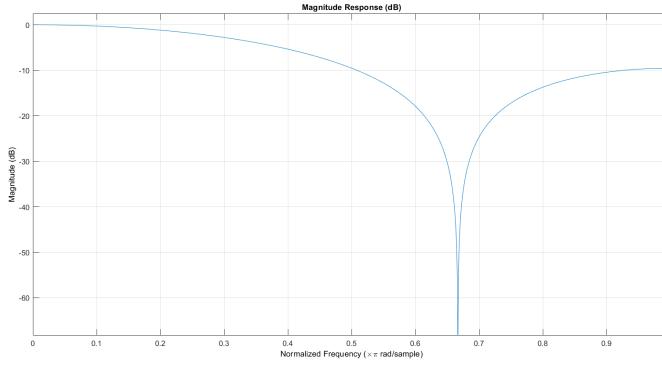


Figure 8: Magnitude response of MA(3)

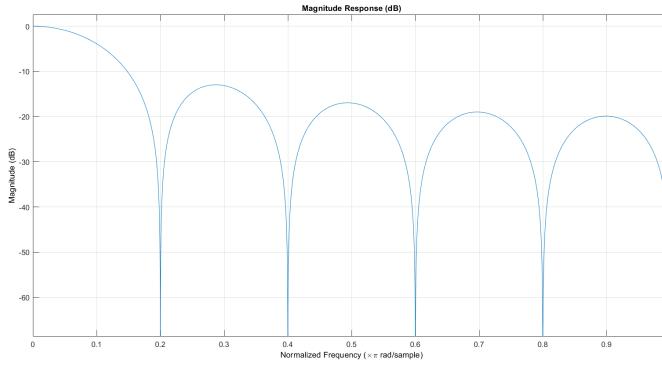


Figure 9: Magnitude response of MA(10)

In observation, the $-3dB$ half-power magnitude threshold is crossed at around 0.3 of the normalized frequency in the MA filter of order 3 while the respective crossing point for the MA filter of order 10 is at around 0.03 of the normalized frequency.

When plotting the noisy signal (nECG), the output signal from the MA filter of order 3 ($ma3ECG_2$), the output signal from the MA filter of order 10 ($ma10ECG$) and the expected signal, it can clearly show that the increment of the order helps in high-frequency noise removal as expected. However, higher-order filter also smooths out the peaks of the expected signal (due to their sudden behaviour) even though they are expected to be preserved.

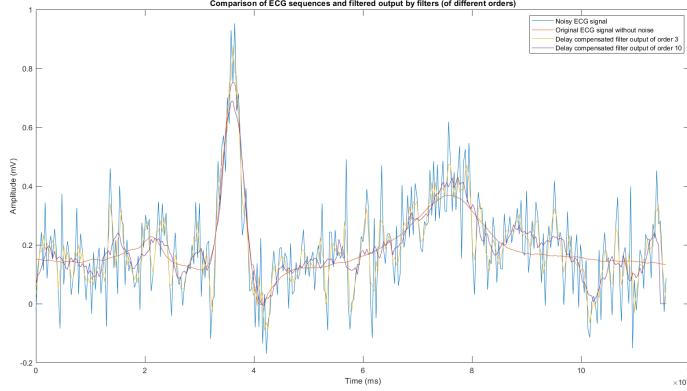


Figure 10: Comparison of ECG sequences and filtered output from MA filters of different orders

Optimum filters Since lower-order MA filters reserve a significant amount of high-frequency noise while the higher-order MA filters smooth out the expected signal characteristics as well, it is needed to find the best possible filter order in between to determine the filter order which gives the least error (i.e. best-expected output signal in this case). For that determination, the variation of error (i.e. mean squared error) against filter order is plotted. As per the data points in the plot, the minimum MSE of 0.00161 is obtained when a moving average filter of order 11 is used.

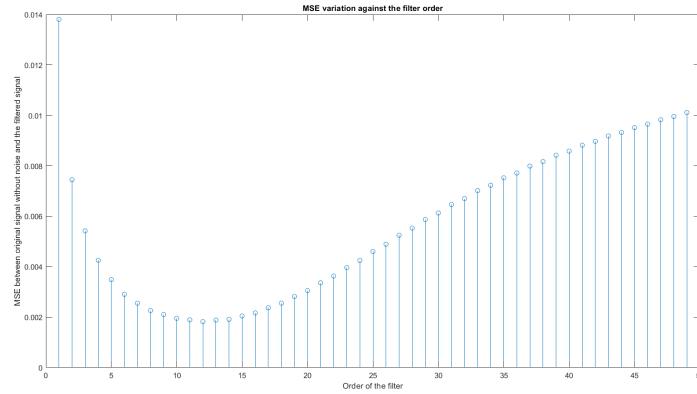


Figure 11: MSE variation against the filter order of MA

1.2 Savitzky-Golay Filters

Savitzky-Golay (SG) filters, which are also known as least-square polynomial estimation filters, are also set smoothing filters which operate by fitting a polynomial function to a set of data points in which the order of the polynomial function is the order of the filter (N) while the length of the filter is L (which defines the odd number of data points to be $2L + 1$). If the filter order is zero (i.e. $N = 0$), SG filter acts as a MA filter.

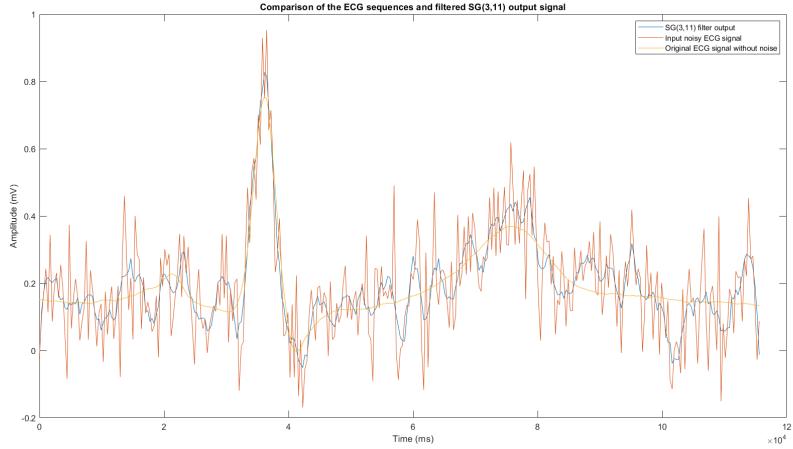


Figure 12: ECG sequences with respect to filtered SG(3,11) output

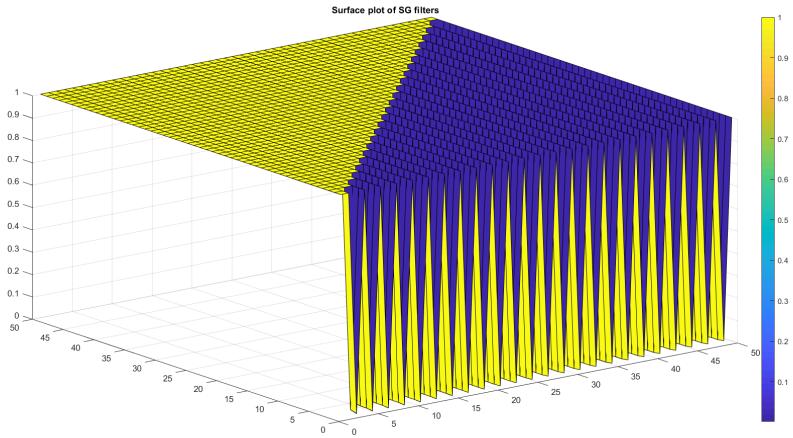


Figure 13: Surface plot of SG filters

The implementation functions (*sgolayfilt*) of MATLAB to calculate the filter coefficients resulted in having unstable values when the filter order is higher. Thus, a maximum order of the filter is experimentally selected to be 30, such that the obtained results are ensured to have meaningful context. In addition, as the signal must have at least $2L + 1$ data points to be filtered (given the filter length is L which is an odd number), the filter length is experimentally limited to 150. As a note, since nECG has only 340 samples and if the maximum possible value is selected to be 169, none of the results would be meaningful. As per the calculations for optimum filter parameters, the minimum error occurred when $N = 4$ and $L = 17$.

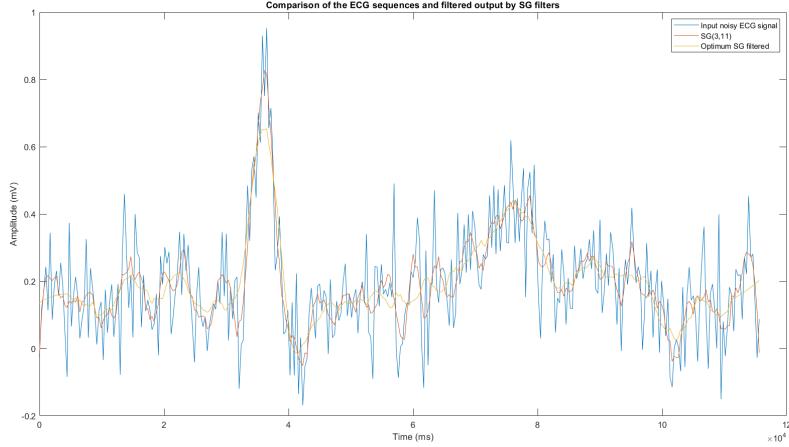


Figure 14: ECG sequences with respect to filtered outputs by SG filters

Comparison between MA and SG filters When compared,

- MA filters: Computationally inexpensive since of the constant filter coefficients, Remove high-frequency components of the expected signal as well along with the smoothing action
- SG filters: Computationally expensive since filter coefficients have to be computed for each set of data points, Comparatively better in high-frequency noise removal than the MA filters

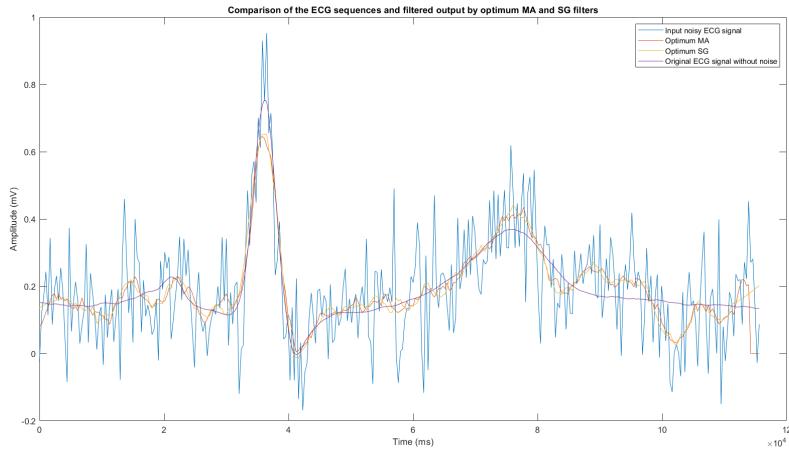


Figure 15: ECG sequences with respect to filtered outputs by optimum MA and SG filters

2 Ensemble Averaging

In the case of overlapping signal and noise spectra, synchronized averaging is effective and simple for noise removal with minimal signal distortions. However, the input data should have multiple measurements or repetitive patterns in order for synchronized averaging to be applied and consequently, when averaged over a large enough number of measurements, the average noise at each point would be zero since the noise is expected

to be random. This ensemble averaging is typically useful in situations where the noise power is comparable or greater to the power of the signal.

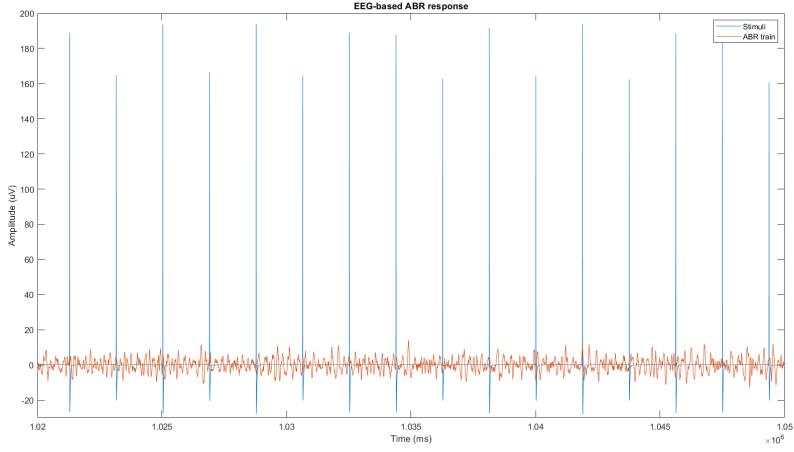


Figure 16: EEG-based ABR responses

Through auditory brainstem response (ABR) in the range of μV from EEG signals which are overshadowed by the ongoing EEG noise in the range of mV , the utilization of ensemble averaging could be demonstrated.

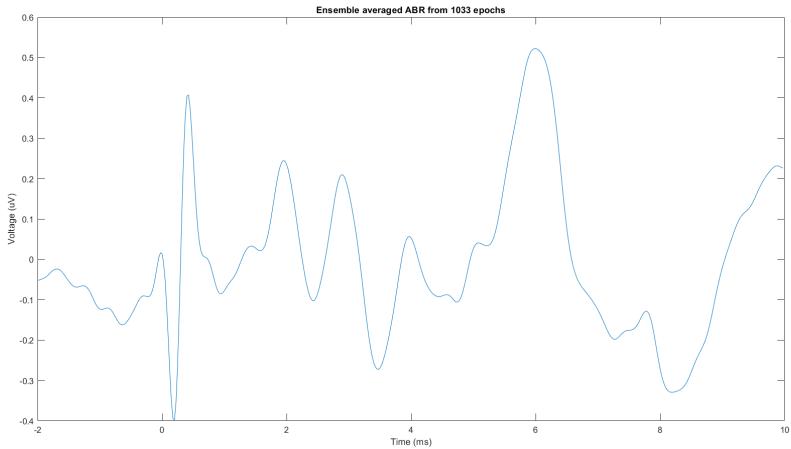


Figure 17: Ensemble averaged ABR response

As per the zoomed image, any distinguishable pattern could not be noticed. However, as ABR are elicited by the auditory stimuli, it is reasonable to separate ABR train into repeated patterns using the stimuli starting points. The exact time window for separation and extraction of ABR epochs would be $12ms$ such that $-2ms$ to $+10ms$ with respect to stimulus starting point. The resultant ensemble average of the extracted epochs are shown in the figure 17.

2.1 MSE variation in ensemble averaging

If the ensemble average which is calculated earlier is to be considered as the desired signal, the MSE against the number of epochs could be plotted to get insights on the error variation.

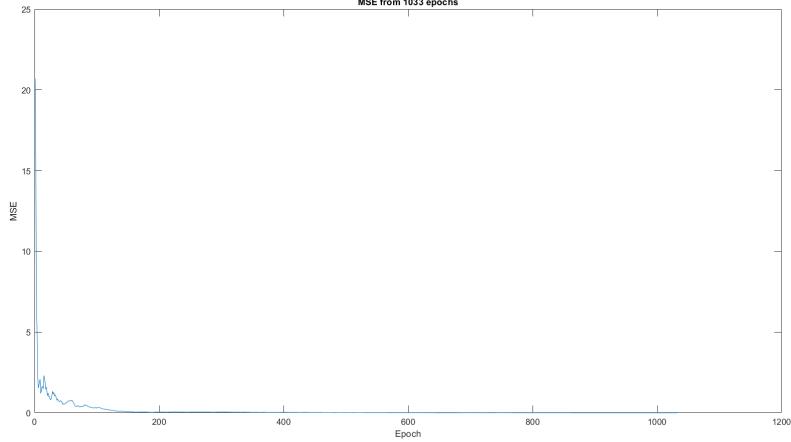


Figure 18: MSE variation through epochs

It is clear that the error is significantly reduced when the number of epochs is higher, especially when it is plotted in the logarithmic scale, in which the error reaches negative infinity (i.e. when the error reaches to zero). The insights of the error variation could be further highlighted by observing the signal-to-noise ratio (SNR).

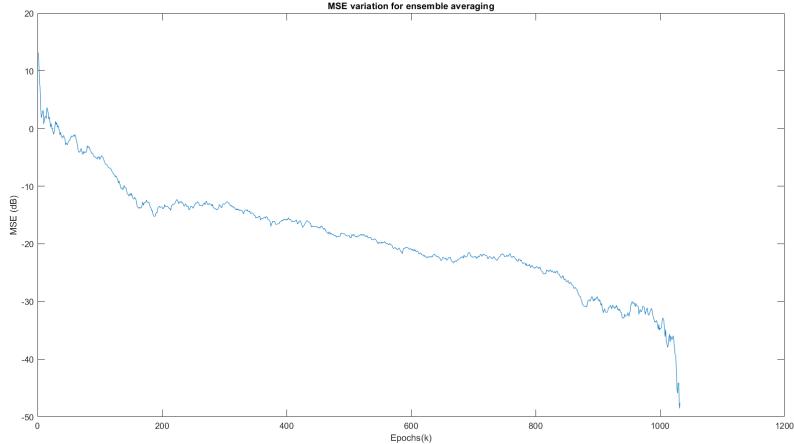


Figure 19: MSE variation through epochs in logarithmic scale

2.2 Improvement of SNR in ensemble averaging

If k epochs are considered to attain the ensemble average, the total signal (S_k) could be calculated as:

$$S_k = \sum_{i=0}^k S_i = k S_i \quad (10)$$

where S_i is the same signal for each epoch. Since noise is considered to be random and assuming the acquired signal is formed by adding or subtracting elements from the expected signal and noise signal (while the noise in the summed signal follows same argument), it could be deduced that the resultant acquired signal will have a variance that is the sum of two initial signal populations. Thus, the variance of the summed signal (i.e. noise) would be:

$$\sigma_k^2 = \sum_{i=0}^k \sigma_i^2 = k\sigma_i^2 \quad (11)$$

where σ_i^2 is the variance of the noise. Thus, the standard deviation of noise in the acquired signal is $\sqrt{k}\sigma_i$ and therefore, the SNR is, from 10 and 11,

$$SNR = \frac{S_k}{\sigma_k} = \sqrt{k} \frac{S_i}{\sigma_i} = \sqrt{k} SNR_i \quad (12)$$

Thus, the SNR improvement is expected to be proportional to \sqrt{k} where k is the number of epochs. The resultant linear variation (in logarithmic scale) would be,

$$SNR_k = \frac{-20\log(k)}{2} + SNR_i \quad (13)$$

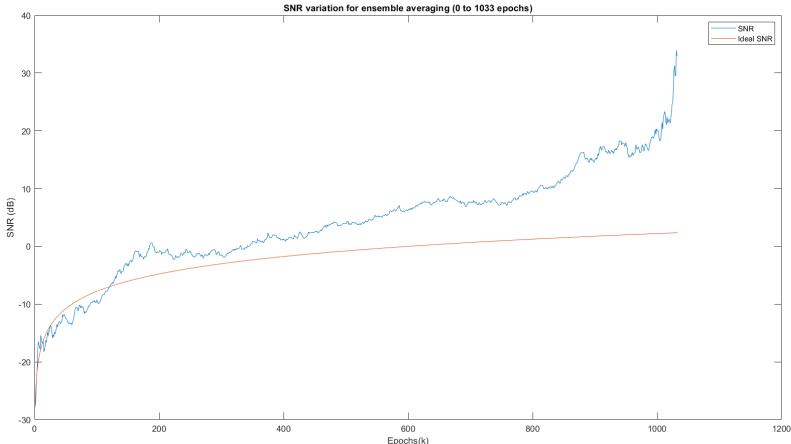


Figure 20: SNR variation for ensemble averaging in logarithmic scale

2.3 Implementation of ensemble average on signals with repeating patterns

When applying of ensemble average on signals which are having repetitive patterns, it is needed to observe the expected waveform after extracting a single ECG pulse.

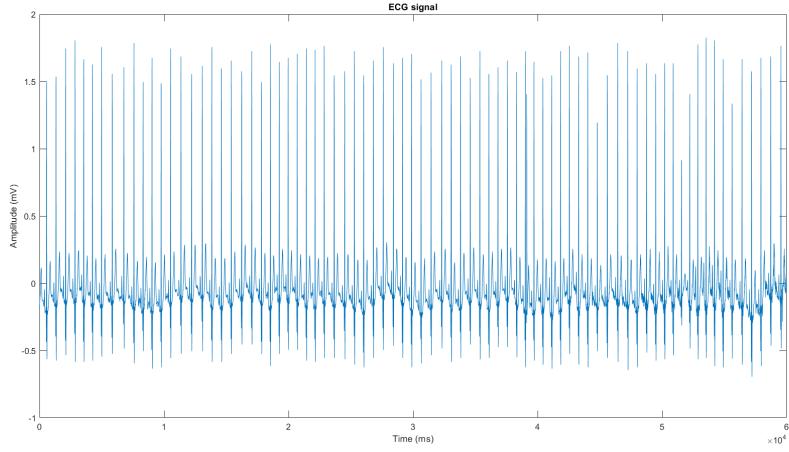


Figure 21: ECG signal sequence

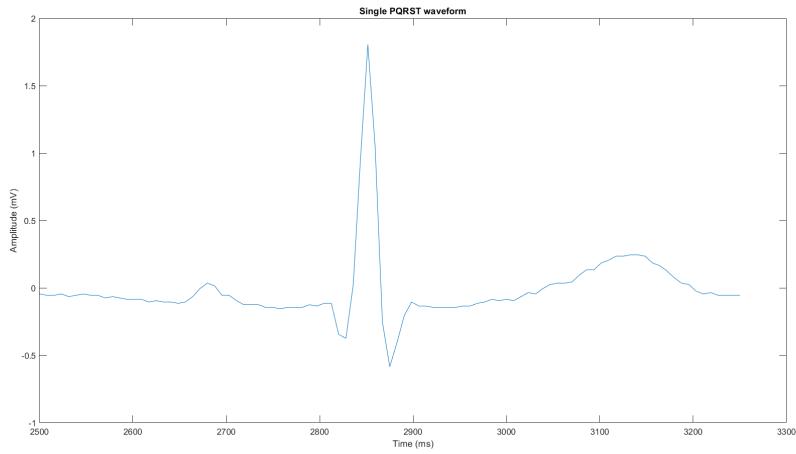


Figure 22: Single PQRST waveform

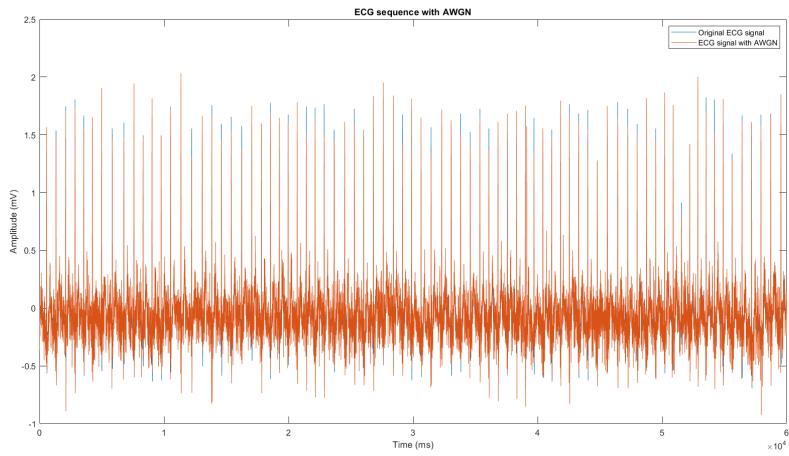


Figure 23: ECG pulse train with added AWGN

Since the ECG template is available, the points of maximum correlation between that template and noisy ECG pulse train is utilized to segment the ECG pulse train into

separate epochs (rather than merely detecting R waves in the pulse train). Through this approach, more accurate segmented positions could be identified which gives the best overlap using a defined cross-correlation threshold score.

The normalized cross-correlation between ECG template and noisy ECG signal would be:

$$\theta_{xy}(k) = \frac{\sum_{n=1}^N ([x(n) - \bar{x}][y(n - k) - \bar{y}])}{\sqrt{\sum_{n=1}^N (x(n) - \bar{x})^2 \sum_{n=1}^N (y(n - k) - \bar{y})^2}} \quad (14)$$

Through observation, the normalized cross-correlation score threshold is set to be 0.08 and that is used to extract the pulse starting points. Through the SNR(dB) vs the number of pulses in the ensemble average plot, it could be observed that when the number of pulses increases, the SNR(dB) is typically increasing as well. Further, with respect to ECG amplitude vs the number of samples plot, it is observable that the ensemble average approaches the ECG template (or rather the ideal ECG waveform) when the number of measurements within the ensemble advances.

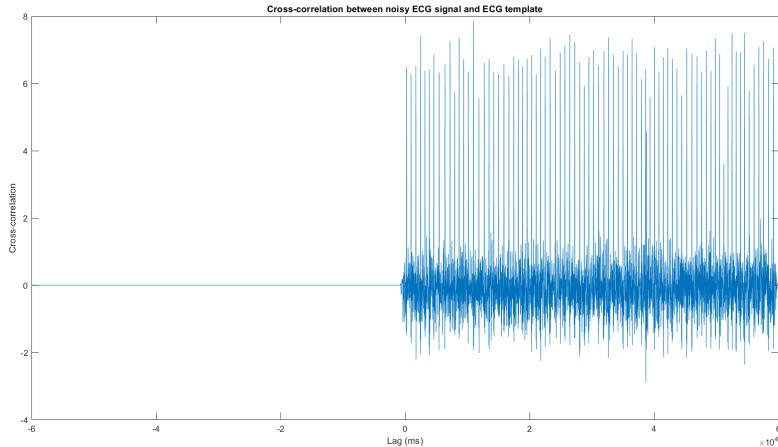


Figure 24: Cross-correlation between noisy ECG signal and ECG template

The superior performance of cross-correlation over mere peak detection Due to the slight variations in the time window, the cross-correlation is more reliable in identifying different wave-forms and reject abnormal waves rather than merely relying upon the peaks. Further, the noise in the noisy ECG signal may have altered the peak characteristics of the ECG waveform which thus negatively affects the identification process of the R peaks in the ensembling approach.

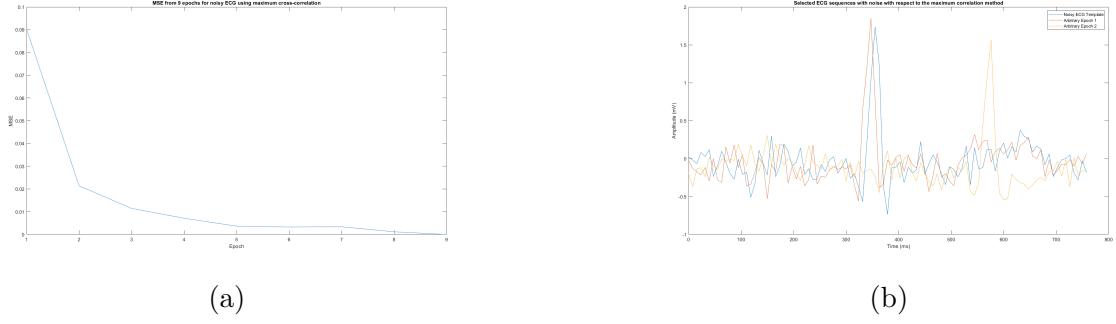


Figure 25: MSE variation and results visualization using cross-correlation method

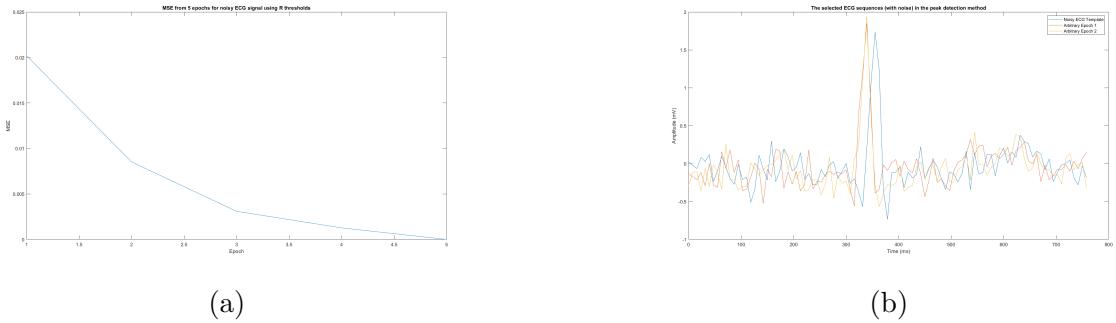


Figure 26: MSE variation and results visualization using R peaks method

3 FIR derivative filters

3.1 Derivative-based filters

Even though the moving average filters have the same filter coefficients for multiplying all time-shifted samples, it is possible to attain different FIR filters, such as derivative-based filters, by altering their respective filter coefficients.

The numerical definition of a derivative is, as the forward difference equation:

$$y(n) = \frac{d(f(n))}{dt} = \frac{f(n) - f(n - \Delta t)}{\Delta t} \quad (15)$$

It is possible to extend the above definition for a digital signal with sampling period of T as:

$$y[n] = \frac{x[n] - x[n - 1]}{T} \quad (16)$$

If $b_0 = 1$, $b_1 = -1$, $b_k = 0$ for $k > 1$ of a FIR filter, the transfer function of that *first order* filter would be:

$$H(z) = \frac{1}{T}(1 - z^{-1}) \quad (17)$$

This is similar to the utilization of backward difference and central difference formula to obtain derivation-based filters. In addition, since it is not concerned about the time

shifting (including direction), the backward difference would be conceptualized same as the forward difference in terms of the time shifting. However, the central difference follows a different result in operation since:

$$y(n) = \frac{d(f(n))}{dt} = \frac{f(n - \Delta t) - f(n + \Delta t)}{2\Delta t} \quad (18)$$

Similarly, for a digital signal with sampling period T using 21:

$$y[n] = \frac{x[n + 1] - x[n - 1]}{2T} \quad (19)$$

Given that it is acceptable to ignore the time shift of the filter output in the above equation, especially for causal systems,

$$y[n] = \frac{x[n] - x[n - 2]}{2T} \quad (20)$$

Thus, $b_0 = 1$, $b_1 = 0$, $b_2 = -1$ and $b_k = 0$ for $k > 2$ of a FIR filter, the transfer function would be:

$$H(z) = \frac{1}{2T}(1 - z^{-2}) \quad (21)$$

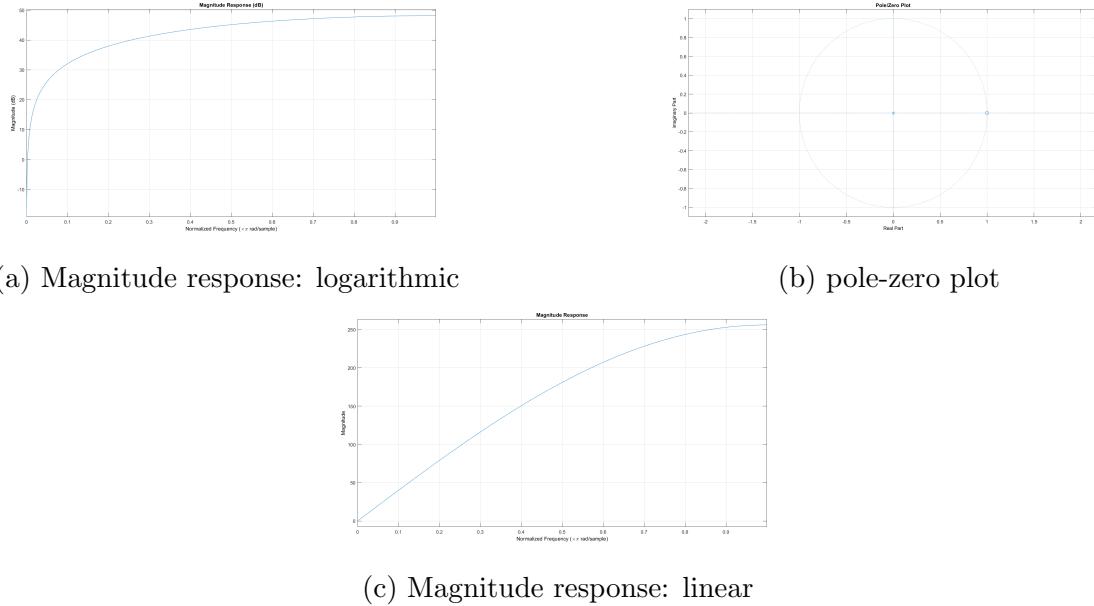


Figure 27: First-order derivative filter

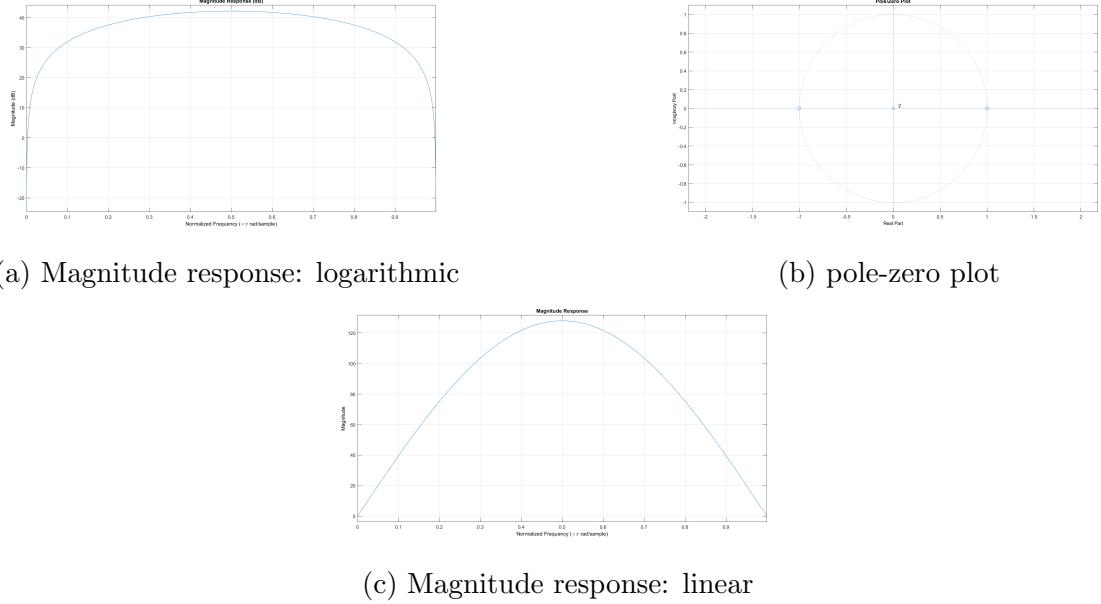


Figure 28: Three-point central difference filter

As observed in the magnitude plots of the derivative filters, the amplification present in the passband frequencies (i.e. $gain > 1$) creates distortions in the output. Thus, it is required to apply a scaling factor (G) to prevent the gain from exceeding 1 in the passband. The value of G could be obtained by setting the maximum possible gain as 1.

$$g = \frac{1}{|H(z)|_{max}} \quad (22)$$

- Therefore, for first order derivative filter,

– From 17,

$$|H(z)| = \frac{1}{T}(1 - z^{-1}) = \frac{|Z - 1|}{T} \quad (23)$$

- Thus, the amplitude is maximum when $Z = -1 \rightarrow |H(z)|_{max} = \frac{2}{T}$
- Hence, $G = \frac{T}{2}$ and $y[n] = \frac{1}{2}(x[n] - x[n - 1])$

- For central difference derivative filter,

– From 17,

$$|H(z)| = \frac{1}{2T}(1 - z^{-2}) = \frac{|Z^2 - 1|}{2T} \quad (24)$$

- Thus, the amplitude is maximum when $Z^2 = -1$ or $Z = j \rightarrow |H(z)|_{max} = \frac{\sqrt{2}}{2T} = \frac{1}{T}$
- Hence, $G = T$ and $y[n] = \frac{1}{2}(x[n] - x[n - 2])$

- The added benefit of this approach is that the filter output is not affected by the sampling period

3.2 FIR derivative filter application

In this section, the above derivative FIR filters are utilized to filter ECG signals which are corrupted by EMG signals and high-frequency noise signals.

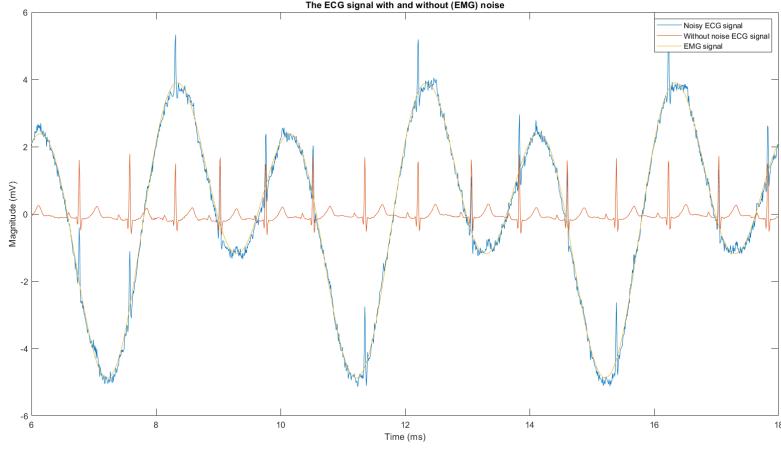


Figure 29: ECG signal with and without EMG noise

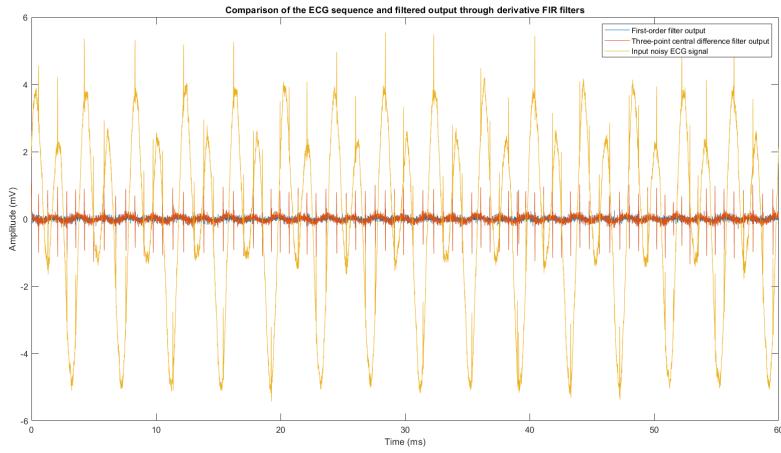


Figure 30: ECG sequences and filtered outputs through derivative filters

Through the results, it could be shown that the derivative filters are sufficiently capable of removing low-frequency EMG noises and less capable of removing high-frequency noises. In visual comparison, the implemented first-order filter performs better noise attenuation than the three-point central difference filter.

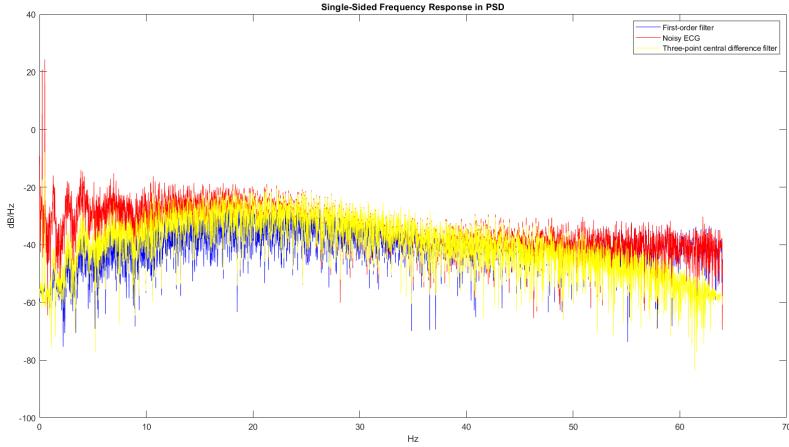


Figure 31: PSD; single-sided frequency response

4 Designing FIR filters using windows

The windowing method is aimed to obtain a finite approximation of the ideal filter through windowing such ideal filter, $h[n]$, using a window function $w[n]$. Typically, an FIR filter designed in this manner sufficiently approximates the ideal filter when the window length increases.

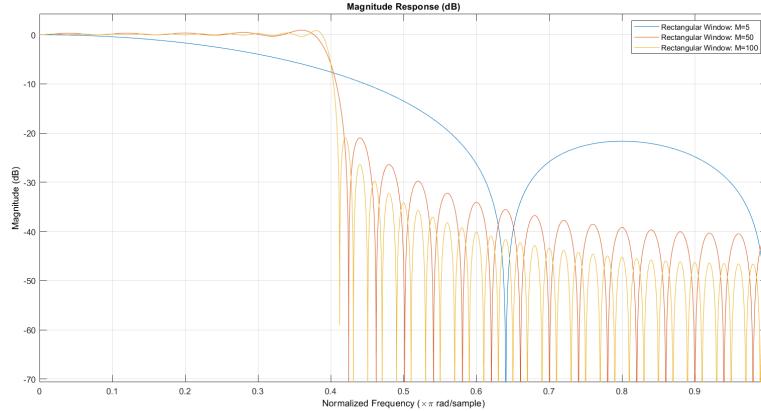


Figure 32: Magnitude response of rectangular window FIR filters

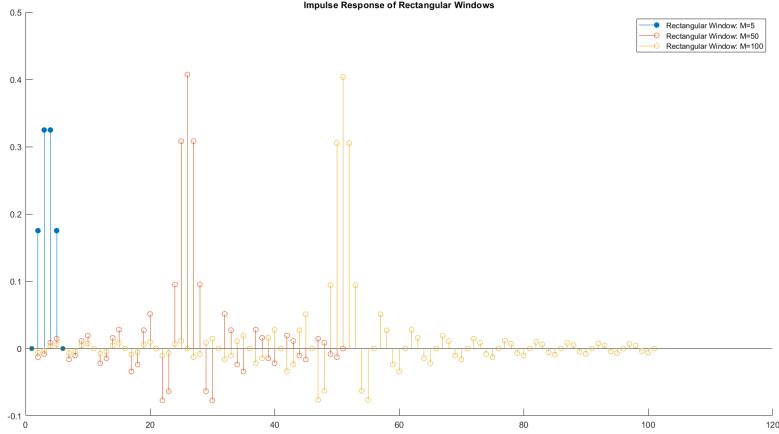


Figure 33: Impulse response of rectangular window FIR filters

Through the impulse response and magnitude response plots, it is observable that:

- Pass band ripples reduce
- Higher attenuation in the stop band could be achieved
- Transition band narrows
- Filter has a sharper cut-off

when the filter order increases.

4.1 Improving the windowing function

The rectangular window simply truncates the impulse response and thereby, manipulates a discontinuity in the time domain and hence, results in having ripples in the passband. This ripple effect raises undesirable effects in the passband, especially in the applications such as audio filtering and engineering. This effect could be minimized by eliminating sudden transitions in the window, apparently by making the transition smoother through different windowing functions.

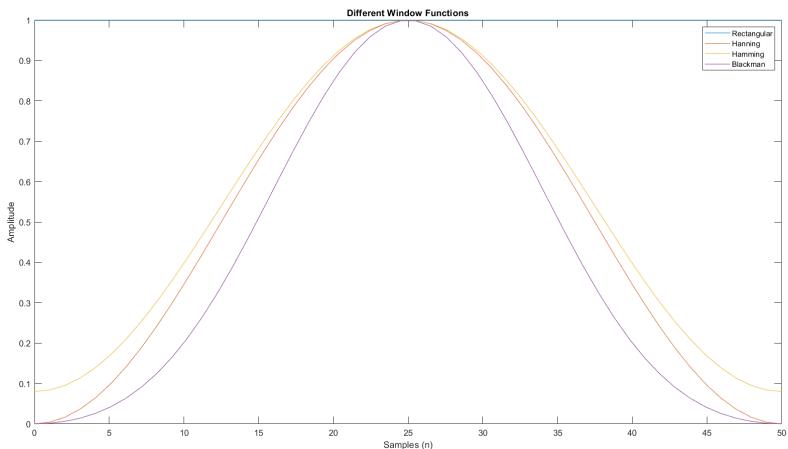


Figure 34: Amplitude vs samples: different window FIR filters

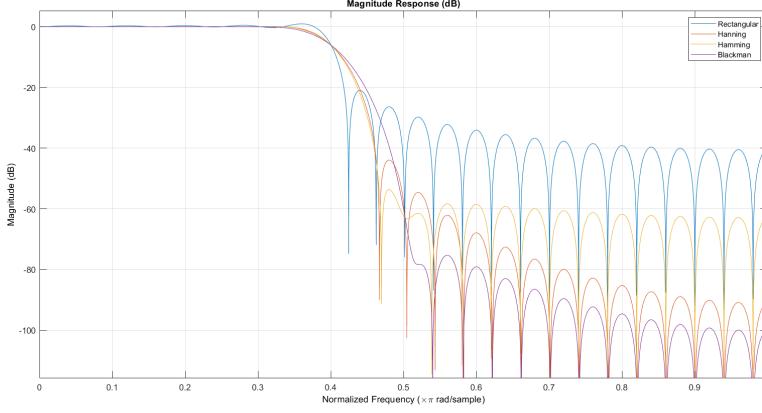


Figure 35: Magnitude response of different window FIR filters

Even though the rectangular window has a sharp transition at $n = 0$ and $n = 50$ as shown in the window functions plot, most of the other window functions are closely and symmetrically bell-shaped and smooth the sharp 0 to 1 (or vice versa) transition over a number of samples. The result of these smoothings could be observed in the magnitude plots, especially in the passband frequency range.

4.2 FIR filter design and application using Kaiser window

The Kaiser window is a generic window form to approximate an array of windows by varying two parameters: shaping parameter (β) and window length (M). This method provides a solution to the trade-off between ripple amplitude and transition width by its shaping parameter which is calculated based upon the passband ripple.

Before applying the Kaiser windowed filter to remove the noise in the ECG signal, it is needed to acquire sufficient insights from the noise distribution present in the ECG signal.

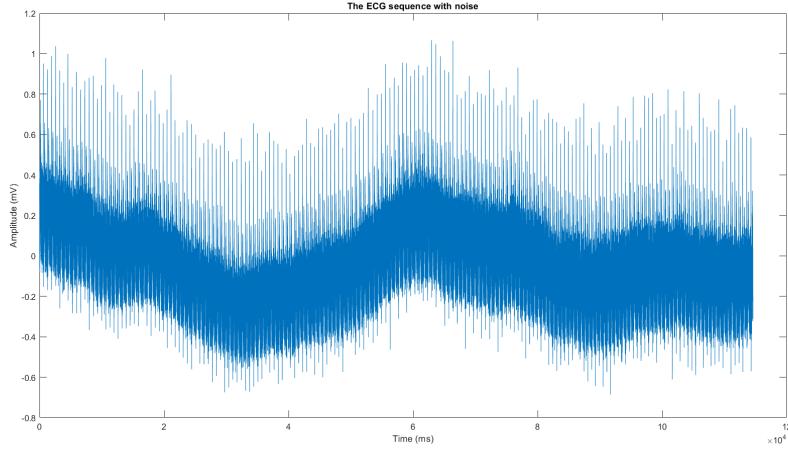


Figure 36: ECG sequence with noise

As it is observed, there exists both low-frequency noise (changes the average value of the expected signal while having a longer time period) and high-frequency noise (manip-

ulates sudden small variations in the signal). This could be further verified by visualizing the frequency spectrum of the noisy signal compared to the expected ECG frequency spectrum.

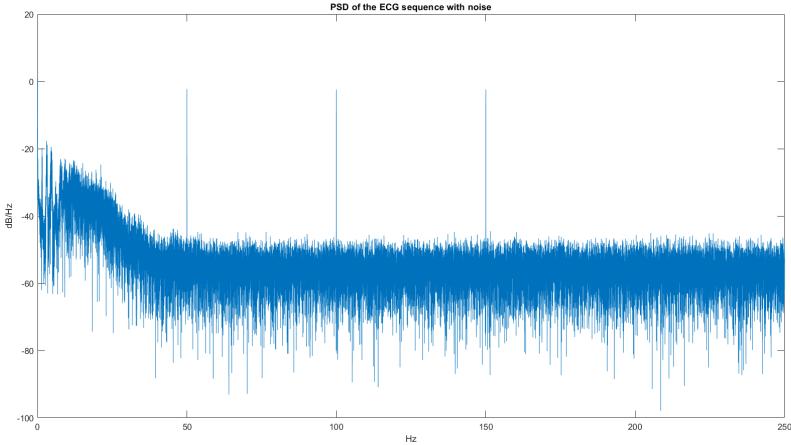


Figure 37: PSD of ECG sequence with noise

As observed in the frequency plots, exceptionally high power is conveyed at the frequencies of 50Hz , 100Hz and 150Hz which could be regarded as powerline noises. In addition, it could be perceived that the noisy signal has more power in the higher frequencies while having random white noise in throughout the frequency spectrum.

4.3 Calculation of filter parameters in Kaiser method

It is needed to have both low-pass and high-pass filters to remove high-frequency and low-frequency noise respectively. In addition, notch or comb filters could be utilized to remove the powerline periodic noise.

If the ECG bandwidth is from 5Hz to 125Hz ,

	High pass	Low pass
f_{pass}	8	122
f_c	5	125
f_{stop}	2	128
δ	0.001	

Table 1: Frequency ranges

For comb filter for powerline noise removal:

- $f_{stop_1} = 50\text{Hz}$
- $f_{stop_2} = 100\text{Hz}$
- $f_{stop_3} = 150\text{Hz}$

In practical implementations, it is impossible to single out an isolated particular frequency. Therefore, a finite transition width is set to be cut off in which the limits are

defined by f_{pass} and f_{stop} . In addition, practical windowed filters do contain ripples in both passband and stop-band and thus, a ripple amplitude parameter, δ , is defined when designing the filter. For practical simplicity, the same transition width and δ values are utilized for both highpass and lowpass filters.

With respect to Kaiser window implementation, the required parameters could be calculated using the following equations.

$$M = \frac{A - 8}{2.285\Delta\omega} \quad (25)$$

$$A = -20\log_{10}\delta \quad (26)$$

$$\Delta\omega = |\omega_{stop} - \omega_{pass}| = |f_{stop} - f_{pass}| \times T \quad (27)$$

where T is the sampling period. Further, the shaping parameter,

$$\beta = \begin{cases} 0.1102(A - 8.7) & A > 50 \\ 0.5842(A - 21)^{0.4} + 0.07886(A - 21) & 21 \leq A \leq 50 \\ 0 & A < 21 \end{cases} \quad (28)$$

Thus, the calculated filter parameters would be:

	High pass	Low pass
$ f_{stop} - f_{pass} $	$ 2 - 8 = 6$	$ 128 - 122 = 6$
$(\omega_{stop} - \omega_{pass})/\Delta\omega$		0.0754
A		60dB
M	302	84
β	5.65326	0.79290

Table 2: Filter parameters

In contrast, since the sampling frequency is $500Hz$ and thus, the highest frequency which the signal could contain is $250Hz$ (in accordance with Nyquist theorem), the ideal filters will be windowed at a lowpass frequency of 0.5π and highpass frequency at $\frac{\pi}{50}$ in normalized terms.

In application, the filters are applied in the following order: Lowpass, highpass and comb filter. This cascade of filters is possible to be convolved to a single filter which has the characteristics of all component filters as observed in the filter frequency response.

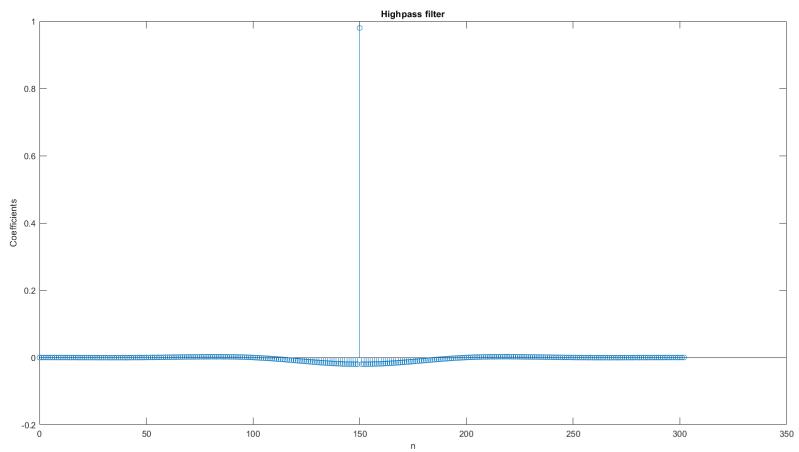


Figure 38: Highpass filter using Kaiser method

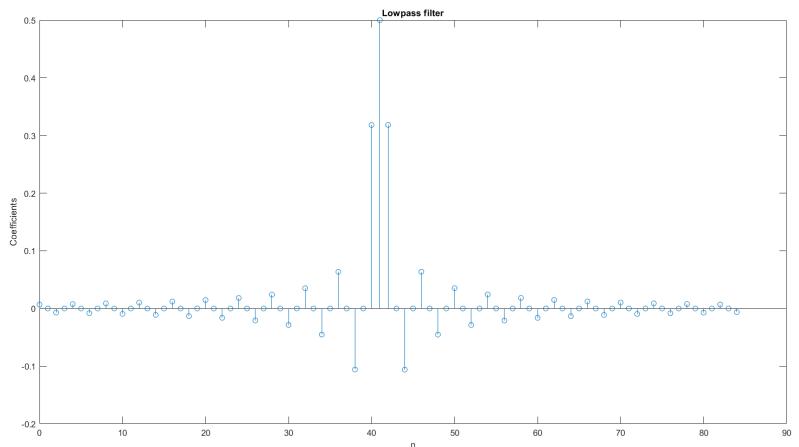


Figure 39: Lowpass filter using Kaiser method

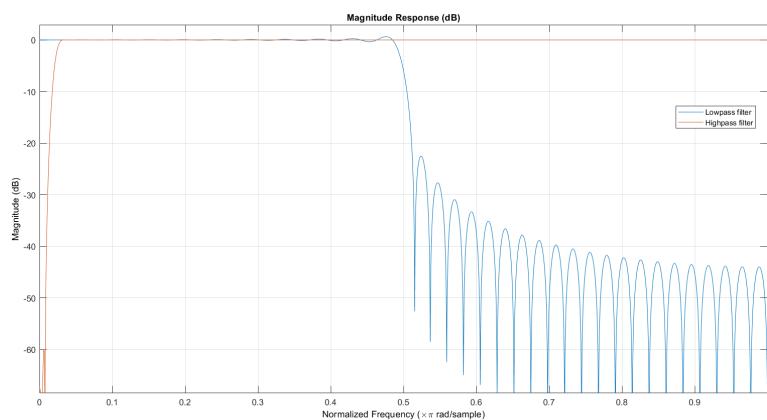


Figure 40: Magnitude response of lowpass and highpass filters from Kaiser method

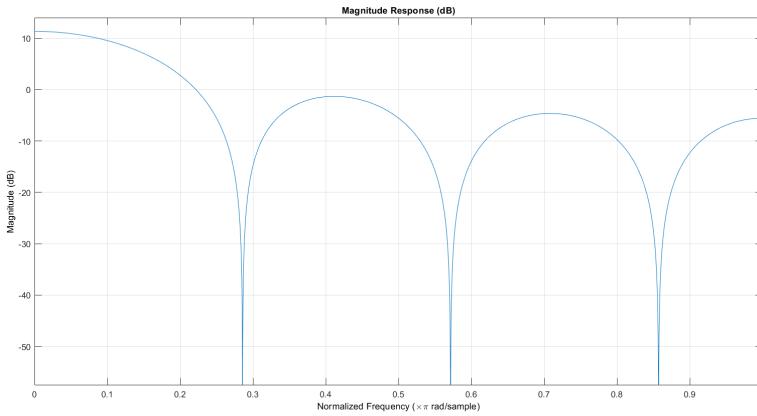


Figure 41: Magnitude response of comb filter

The PSD of the noisy ECG signal, filtered ECG signal and the expected ideal one shows that the lowpass filter has reduced the power in higher frequencies while the high-pass filter has reduced the power in frequencies below 5Hz. Further, the comb filter is effective in removing the peaks at the powerline noises of 50Hz, 100Hz and 150Hz.

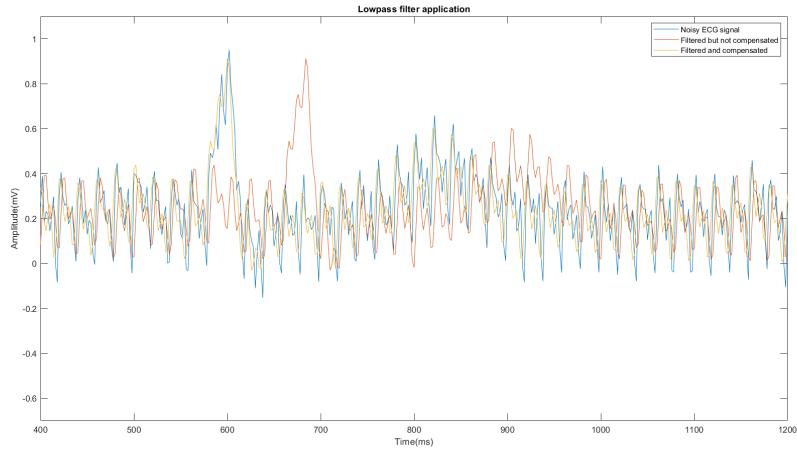


Figure 42: Lowpass filter application on noisy ECG signal

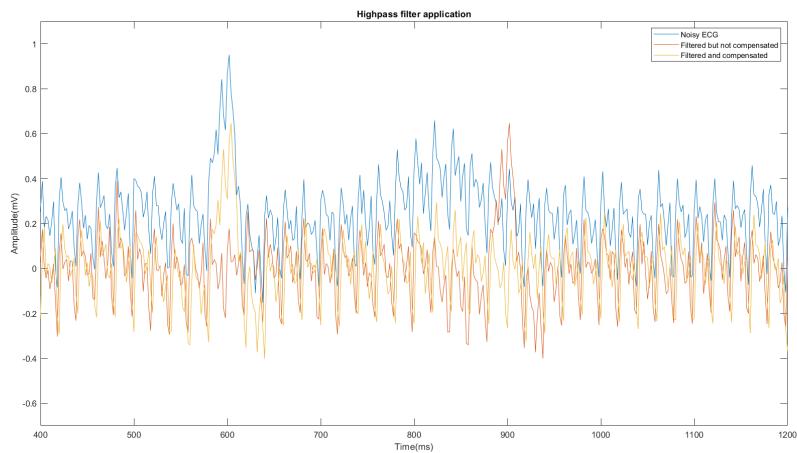


Figure 43: Highpass filter application on noisy ECG signal

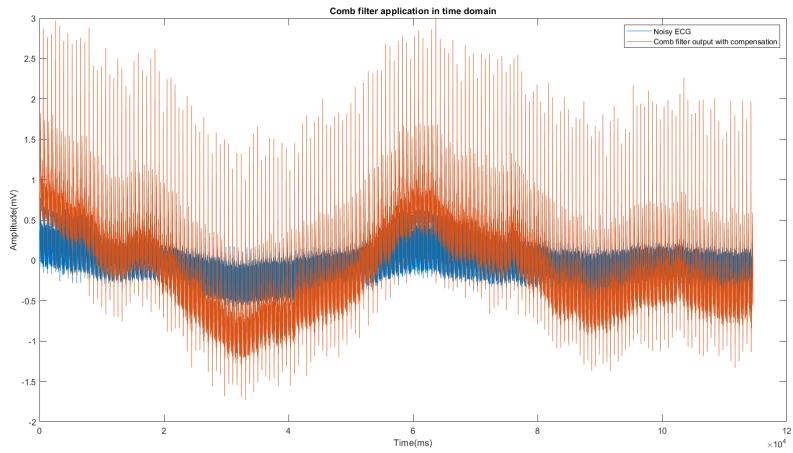


Figure 44: Comb filter application on noisy ECG signal

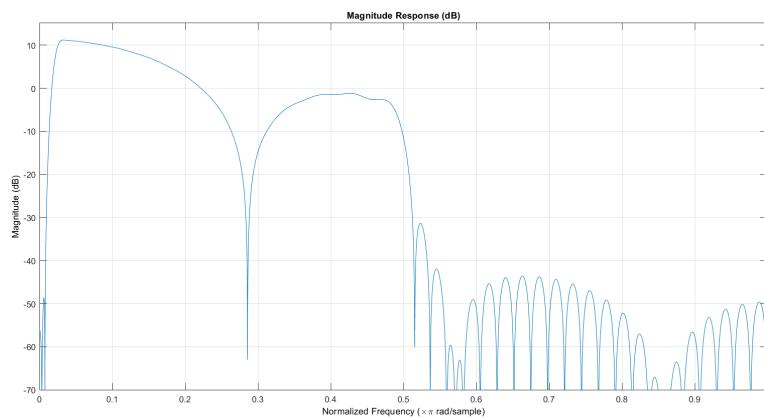


Figure 45: Magnitude response of the cascaded filter

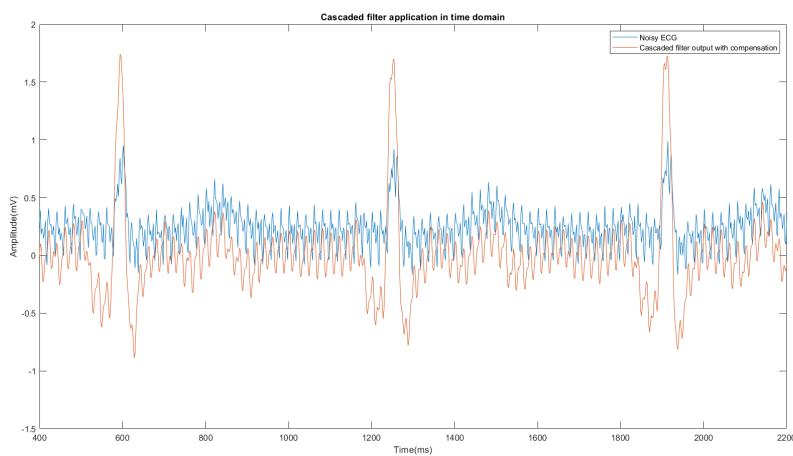


Figure 46: Cascaded filter application in time domain on the noisy ECG signal

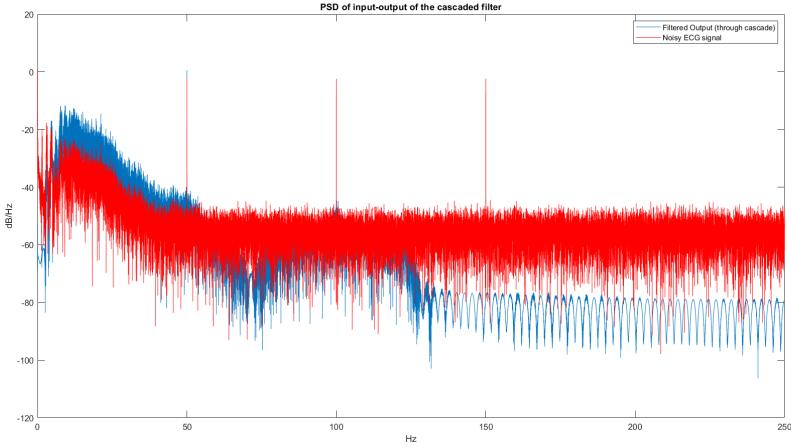


Figure 47: PSD of the input and output of the cascaded filter

5 IIR filters

IIR filters are a set of filters which exhibit non-linear phase within the passband with the realization using a recursive feedback mechanism. This set of filters typically has an infinite impulse response but is not necessarily stable. In general IIR filters have better frequency response compared to FIR filters of the same order.

When directly implementing higher-order IIR filters, stability and frequency distortion issues typically arise leading to not having the required frequency response due to errors in coefficient optimization, rounding off approximations and overflow errors due to the finite number of bit calculations. These issues could be rectified by cascading several smaller order filters such as second-order sections.

If it is required to utilize a direct implementation in MATLAB, a lower-order filter such as 10th order Butterworth filter could be utilized since MATLAB's direct implementation of Butterworth appears to be unstable at orders above 10. This low order will result in a lesser sharp cut-off in the filter compared to the higher order filter.

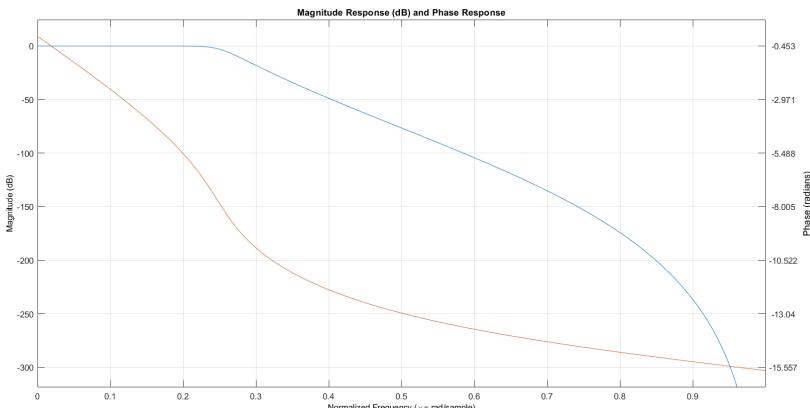


Figure 48: Magnitude and phase responses of the 10th order Butterworth lowpass filter

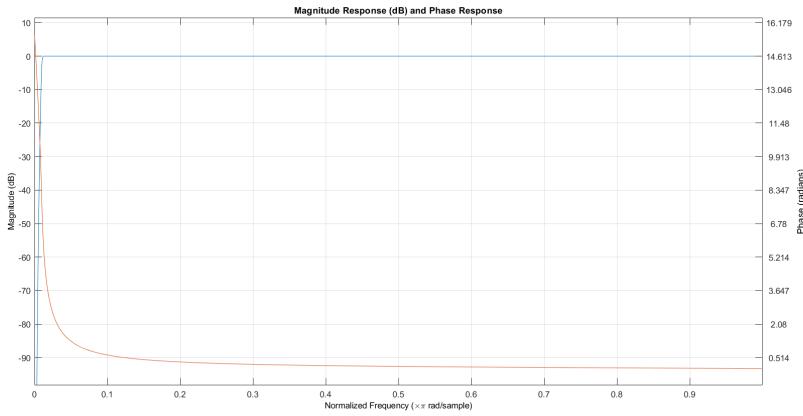


Figure 49: Magnitude and phase responses of the 10th order Butterworth highpass filter

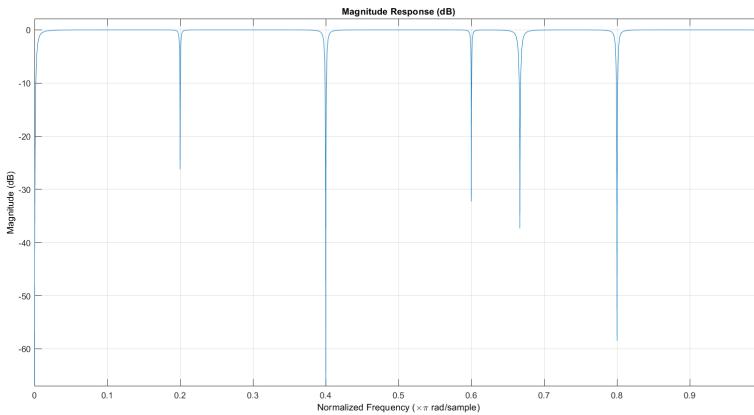


Figure 50: Magnitude response of the notch filter

5.1 IIR implementation of ECG filters

- Lowpass: $f_c = 125\text{Hz}$, $f_s = 500\text{Hz}$. $M = 10$
- Highpass: $f_c = 5\text{Hz}$, $f_s = 500\text{Hz}$. $M = 10$
- Comb: $f_{stop} = [5\text{Hz}, 100\text{Hz}, 150\text{Hz}]$, $F_s = 500\text{Hz}$

These three filters could be cascaded to create the combined IIR filter which is needed.

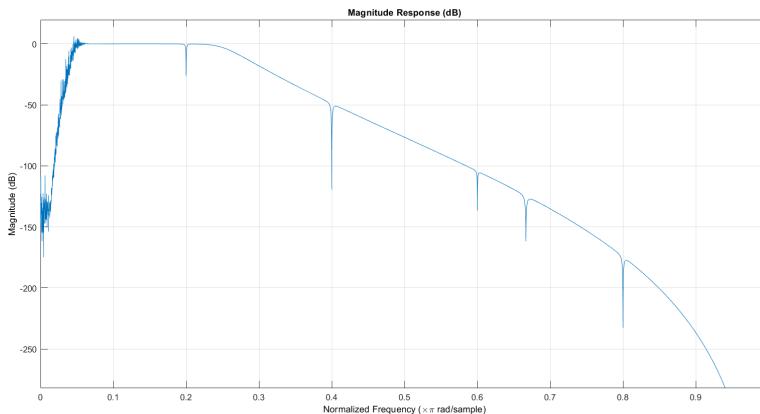


Figure 51: Magnitude response of the cascaded IIR filter

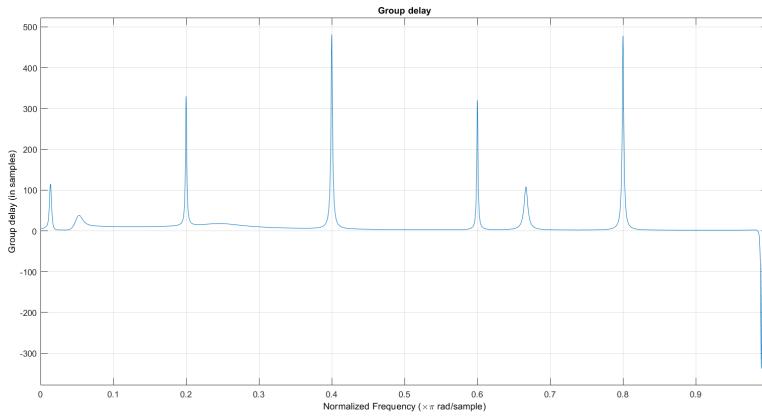


Figure 52: Group delay response of the cascaded IIR filter

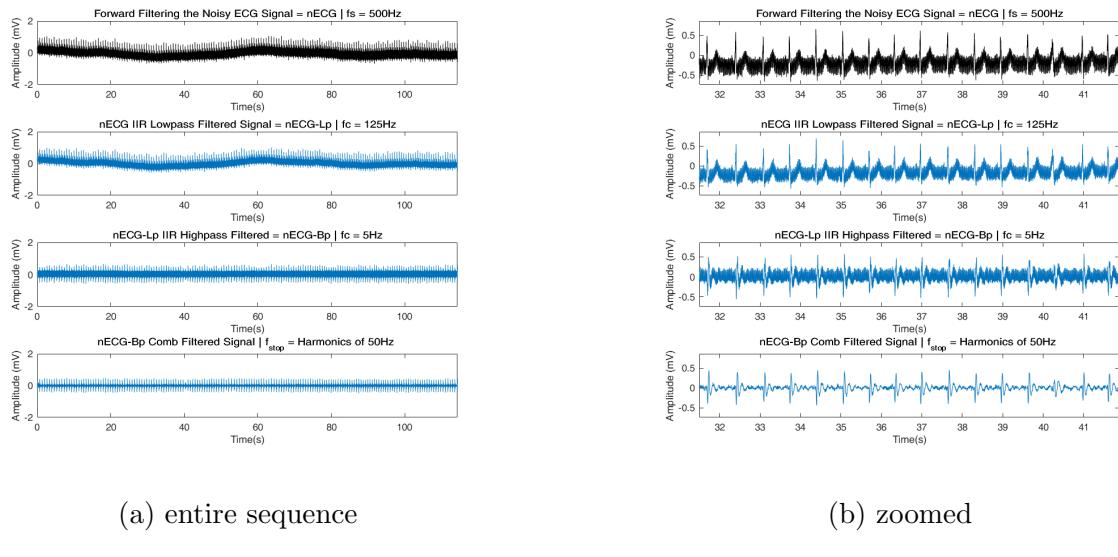
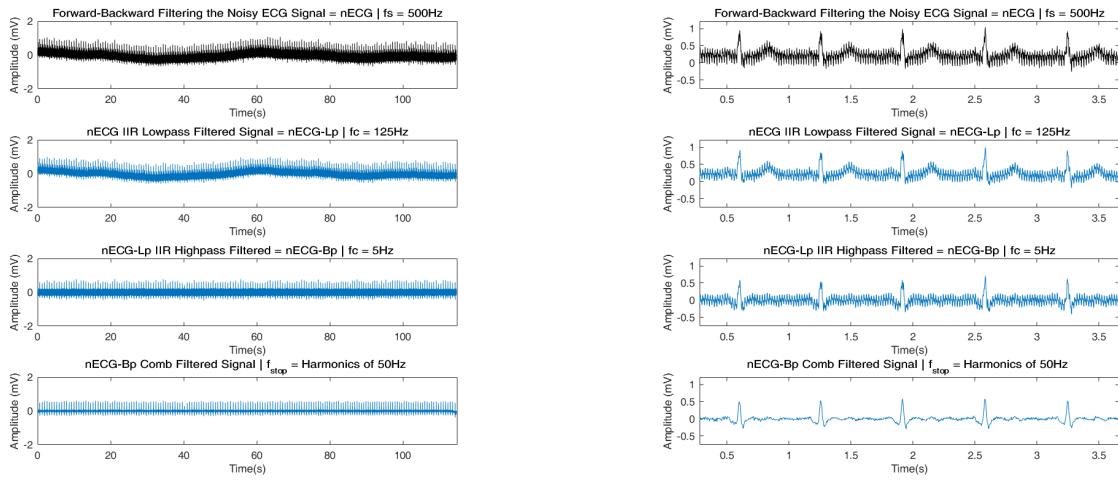


Figure 53: IIR forward filtering



(a) entire sequence

(b) zoomed

Figure 54: IIR backward filtering

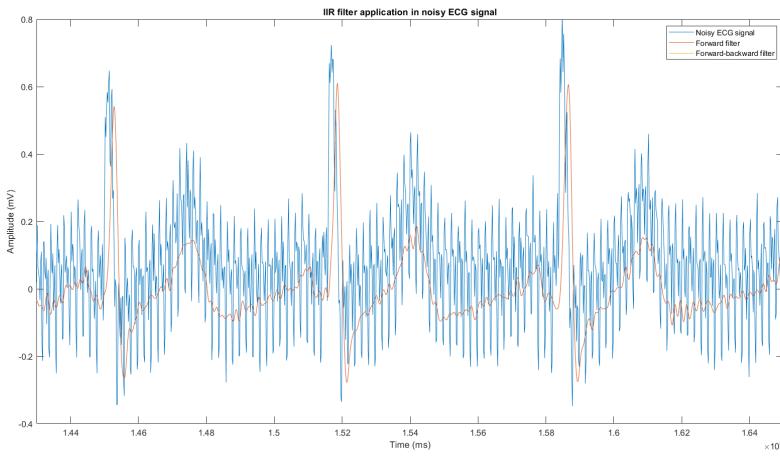


Figure 55: IIR filter application on noisy ECG signal

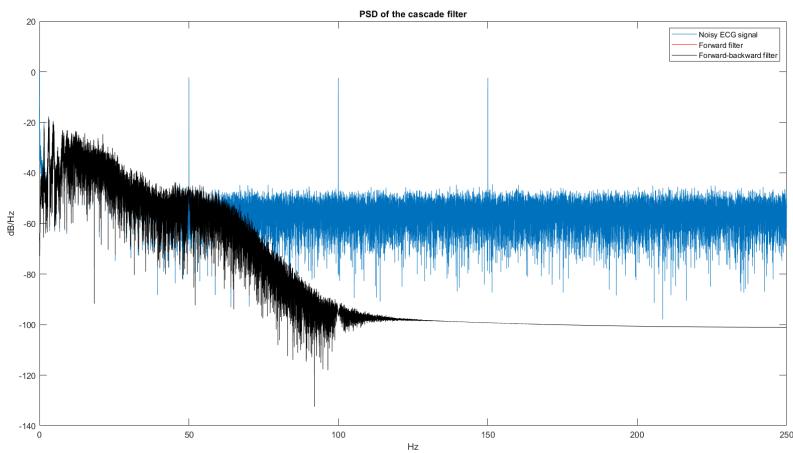


Figure 56: PSD from the cascaded IIR filter

5.2 Comparison between IIR and FIR filters

As could be observed, the IIR filter has a better magnitude response in the pass band in compared to the FIR filter. In addition, the notch filter characteristics do also have a narrower bandwidth compared to the FIR while having high stopband attenuation (in IIR). However, the transition bandwidth is larger in the IIR filter (due to lower order) than in the FIR filter. Further, the IIR filter has a non-linear response in the passband despite its better frequency response.

However, if the entire required signal is stored, forward-backwards filtering through a secondary backward filtering process could be implemented to create zero phase response as observed in the time and frequency domain plots.

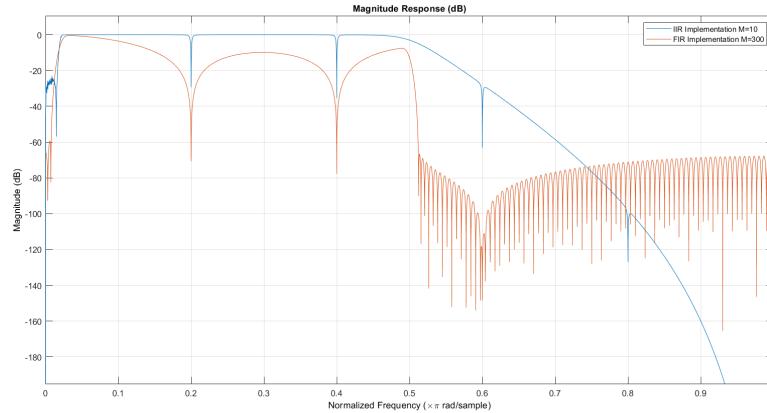


Figure 57: Magnitude response comparison between IIR ($M=10$) and FIR filters

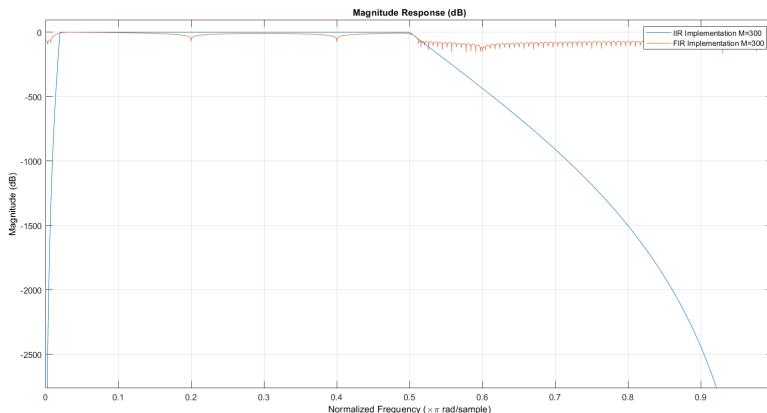


Figure 58: Magnitude response comparison between IIR and FIR filters with same orders

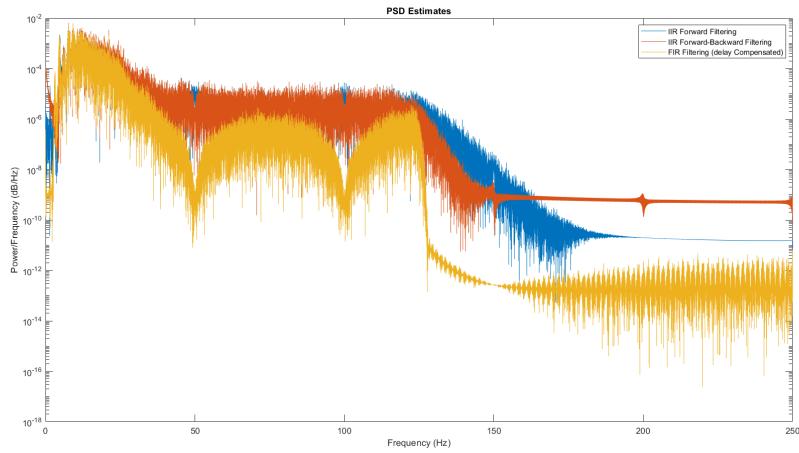


Figure 59: PSD estimates from the cascaded IIR filter and cascaded FIR filters