Digital Signal Processing 2020
Laboratory Work nr. 2

A Study of Digital Filters

Lukas Stasytis, E MEI-0 gr. Birutė Paliakaitė, PhD Student
Kaunas University of Technology, Faculty of Electrical and Electronics Engineering

Introduction

In the following laboratory work, we will be employing digital signal processing techniques, namely - filters, to process an electrocardiogram (EKG) signal[[1],[2]]. More specifically, we will be employing: a relatively low order finite impulse response (FIR), a comb infinite impulse response (IIR) and multirate FIR filters. We will conduct a careful study of said filters' characteristics and test their performance on a real EKG data set as well as two standard synthesized testbench signals. The filters are implemented in the Python programming language with a heavy reliance on the Scipy library[[3],[4],[5]].

Problem statement

Figure 1 shows the EKG signal which we will be processing with our filters. The signal is littered with various types of noise [6] varying from a 50Hz industrial voltage component to a base drift of the signal due to breathing (0.15-0.3Hz) as well as muscle movements in the 2-500Hz band among various other types of noise. Said components severely decrease the clarity of the signal when trying to look for heart-related diseases. We will specifically target the stated three types of noise with lowpass FIR, comb IIR and multirate FIR filters in an attempt to remove said noise.

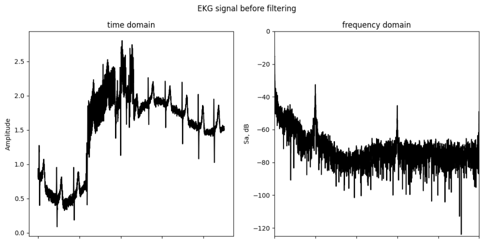

Figure 1. EKG signal for processing.

Low-pass FIR filter

We start by implementing a 13th order low-pass FIR filter to eliminate high frequency noise from our digital signal. Figure 2 shows the structural diagram of said filter [7]. Being as all the filter coefficients are varying multiplications of our input signal with no recursive components, they can be treated as taps in a FIR type filter and require no additional modifications. From the structural diagram we derive our differential equation 1 for the filter.

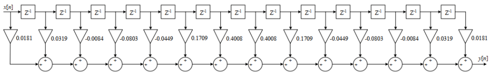

Figure 2. FIR filter structural diagram.

$$H[z] = \frac{x_0 z^0 + x_1 z^{-1} ... + x_{13} z^{-13}}{1} \tag{1}$$

Figure 3 bottom left plot shows the impulse response characteristic of our filter. We used our own implementation written in python for the impulse response analysis: a unit impulse sequence is generated and filtered using the lfilter function from the scipy python numerical library using our coefficients. The resulting filtered signal equals our impulse response. We can observe that the impulse response directly matches our filter tap coefficients.

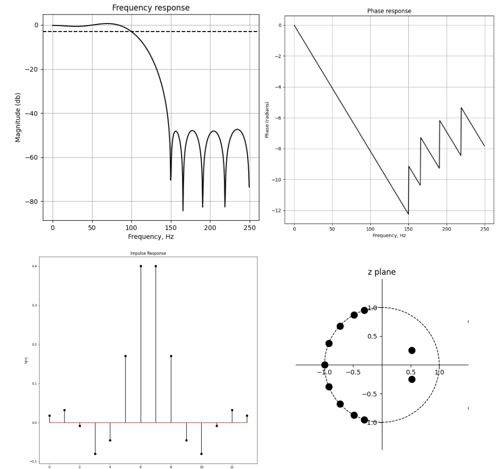

Figure 3. FIR filter impulse(bottom left), frequency (top left), phase(top right) responses and z-plane (bottom right).

Following that, we generate the frequency and phase response graphs of our filter (Figure 3 top left and right graphs). The scipy function freqz is used for this purpose. We keep the frequency x axis non-normalized to have a clear representation of which frequencies our filter is targeting, but limit the frequency axis to the Nyquist rate relative to our sampling rate, the later being 500Hz.

$$h_{dB} = 20 \log_{10} |h| \tag{2}$$

Focusing on the frequency response graph first, we may observe three distinct portions of the filter. The first is the pass band, from frequencies 0 to 100Hz (cutoff point). We treat the filter's response at magnitude -3dB as the cutoff point where the pass band ends. Following that we have a transition band, during which values are starting to get filtered. At around 150Hz we see the start of the cutoff band, symbolized by the start of a harmonic oscillation. The peak of these harmonics marks the amount of attenuation applied on the given frequencies. Around -50dB in this filter's case.

In summary, the graph shows that our filter will pass all frequencies from 0 to 100Hz, followed by a -50dB attenuation of the frequencies higher than 150Hz. Thus, our cutoff frequency is 100Hz, our pass band waving is from 0.6dB down to -0.6dB. This waving can cause small signal drift distortions. Our cutoff's largest attenuation is -50dB. A low-pass FIR filter.

The phase characteristic shows a partially direct phase response, which is important for judging if a given filter will not phase shift different signal components by varying time periods, as opposed to the same for each component. Given the partially direct response in our case, we conclude that the filter is not going to distort the signal.

Lastly, we generate the z plane of our filter. To do this, we take our coefficients again and calculate the roots of each array. Plotting the real and imaginary portions of each value against each other on a plane results in the zero values in the case of b coefficient roots and the poles in the case of a coefficients. The z plane can be seen in figure 3 bottom right graph. We note that a cuttoff is marked for real and imaginary absolute values larger than 1. Values falling outside of this range constitute as unstable and imply our filter becoming a generator. The main points of observation are a.) the absence of any poles, given that this is a FIR filter and b.) none of the zeros being present outside out boundary, implying a stable system and c.) Zeros closer to the edges, which imply higher dampening effects, being on the higher frequency side of the z plane, further implying that we do have a low-pass filter ready to be used on our EKG dataset.

Comb IIR filter

We now design a comb IIR filter for eliminating signal drift and the additional noise component of industrial voltage. This filter would follow the previously outlined FIR filter as a second pipeline stage for processing the EKG signal. The comb filter features a unique characteristic of filtering only specific periodic frequencies (and their adjacent frequencies) f_0 while the remaining are all passed. Figure 4 shows the theoretical frequency response characteristic of said comb filter[7]. Parameter K_0 marks our passband attenuation (in our case 1, meaning no attenuation), K_r marks the amplitude at which we judge our cutoff band width at each frequency period. In our case, this parameter has to equal -3dB. Parameter Δf marks the width of frequencies being cutoff starting at amplitude K_r. Parameter K marks the maximum attenuation at each cutoff frequency period. Lastly, parameter f_0 marks the actual cutoff frequency which we are filtering. Subsequent periods to filter are marked kf_0. f_d marks our sampling frequency. We use the niquist rate of our sampling frequency for plotting the frequency spectrum.

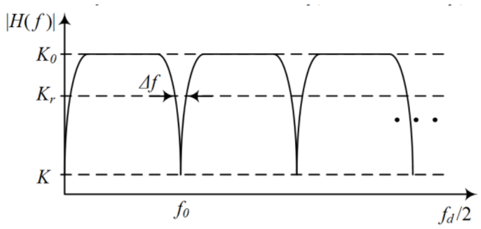

Figure 4. IIR comb filter theoretical frequency response.

Figure 5 shows the structural diagram of our comb filter. We can see that the filter is an IIR filter given the fact that the output of the filter is recursively fed back into it via the k_3 boosted value. Z^{-N}. In total, we have three coefficients which we need to calculate and divide into a and b coefficient arrays.

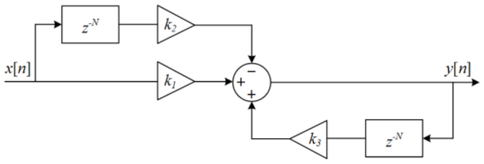

Figure 5. IIR comb filter structural diagram.

Equations 3 to 9 were used for obtaining the k_1, k_2, k_3 coefficients. This includes all the math operations necessary to obtain the filter coefficients given a set of hyper parameters. Notable values are f_d - our sampling frequency, S - our attenuation at the cutoff bands and L - the decibel magnitude at which we measure our cutoff band's width.

$$K = K_0 \cdot 10^{-\frac{S}{20}} \tag{3}$$

$$K_r = K_0 \dot{1}0^{-\frac{L}{20}} \tag{4}$$

$$N = \frac{f_d}{f_0} \tag{5}$$

$$\beta = \sqrt{\frac{K_r^2 - K_0^2}{K^2 - K_r^2}} \cdot \tan\left(\frac{N\pi \cdot \Delta f}{2f_d}\right) \tag{6}$$

$$k_1 = \frac{K_0 + K\beta}{1+\beta} \tag{7}$$

$$k_2 = \frac{K_0 - K\beta}{1+\beta} \tag{8}$$

$$k_3 = \frac{1-\beta}{1+\beta} \tag{9}$$

We obtained the following coefficients (Table 1) with starting hyperparameters of: $S = 40dB$, $\Delta F = 0.52$, $f_0 = 50$, $k_0 = 1$ and $L = 3$.

Table 1. k parameters of the comb filter

β	0.0163
k_1	0.984
k_2	0.983
k_3	0.967

For the sake of clarity, we also append the final b and a coefficient arrays, Table 2. 9x0 symbolizes 9 padded zeroes in between the first and final element of each array.

The resulting differential equation of the filter is as follows in equation 10

Table 2. a and b arrays of the comb filter.

a	1	9x0	$-k_3$
b	k_1	9x0	$-k_2$

$$H(z) = \frac{0.984 - 0.983z^{-10}}{1 - 0.967z^{-10}} \tag{10}$$

We now look at the impulse, frequency and phase responses of our filter. Figure 6. The impulse response (bottom left graph) has two unit impulses at the first and final n values. This aligns with our calculated b coefficient array.

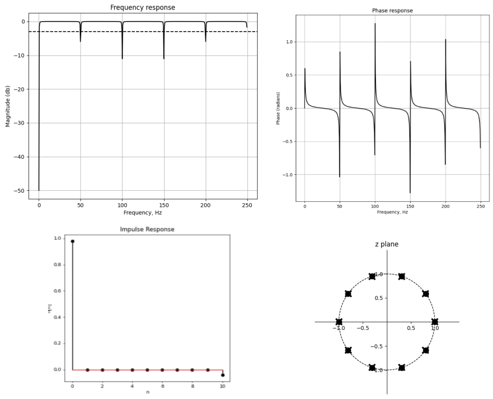

Figure 6. IIR comb filter impulse(bottom left), frequency (top left), phase(top right) responses and z-plane (bottom right).

The frequency response ended up closely matching the theoretical frequency response graph as can be seen in Figure 6 top left graph. Notable is the difference in maximum attenuations are the cutoff frequencies. In theory, we were supposed to have around -50dB attenuation at 50Hz periods. In our case, the comb filter has a -10dB attenuation at 100Hz and 150Hz frequencies and symmetrically falls off, while having a full -50dB attenuation at frequency 0. Multiple attempts were made at trying to generate a filter with even attenuations at each period to no success. The author is confident in the calculated k_1 to k_3 coefficients and can only speculate that there is a difference in matlab's filter and scipy's lfilter function implementations. However, the filter was successful in eliminating some of the signal drift and did seem to filter the industrial voltage component to a degree as will be shown in the results section. Thus, the author concludes the result as satisfactory enough for the purposes of exploring the filter's inner workings.

The frequency response at the 50Hz frequency has a narrow 0.67Hz cutoff band at the -3dB magnitude. The 0.67Hz value was chosen based on the American Health Association standard.

The phase response shows a harmonic phase following the cutoff frequency period. We expected the phases in radians to be even at the positive and negative peaks of each frequency period, but are satisfied enough with the result being harmonic.

Lastly, the z plane (Figure 6 bottom right graph) shows the poles and zeroes all aligning very closely and staying within the absolute 1 value limits, signaling that the filter is stable and dampening specific frequencies at a narrow band (as a result of the poles being very close to zeroes.

We also note that an additional normalization step was done before comb filtering and de-normalization step after. Without this step, the filter caused massive drift in the synthesized tests. We see no negative impact of this scheme on our EKG signal.

Multirate FIR filter

Lastly, we design a multirate lowpass FIR filter [8] for eliminating system drift as opposed to using the comb IIR filter. The idea of our multirate filter is to decimate our input signal to a lower sampling frequency, then apply a higher order lowpass FIR filter and finally interpolate the filtered signal back into the original frequency. By doing the additional decimation and interpolation steps, we provide an essentially quantisized signal for the FIR filter to compute with much smaller computation requirements. Essentially, we trade precision for getting a relatively higher order filter for a given sampling frequency.

Figure 7 shows a structural diagram of our filtering scheme. We divide our decimation and interpolation operations into two stages each to decrease the amount of relative re-sampling happening between stages. The reason for this is because we are required to apply an additional FIR filtering operation before each decimation and after each interpolation operation to eliminate additional aliasing generated as a result of the re-sampling operations. These FIR filters would require larger orders, and thus more computations, if we did large sampling rate jumps at a time.

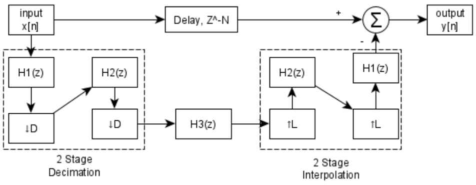

Figure 7. Multirate FIR filter structural diagram.

A non-trivial decision is the selection of decimation and interpolation degrees. We solve equation 11+12 to obtain parameter M - our re-sampling degree. In this equation, f_{sl} equals our cutoff minimum frequency, f_{pr} equals the pass band maximum frequency. We also know that our cutoff frequency proper should not be larger than 0.67Hz as discussed in the comb filter section. Using equations 13 to 16 we concluded that our M should be around 50, D_1 - 10, D_2 - 5. While the actual M value we obtained was 58, we round it to be dividable from 500 (our sampling rate). In the end, we had five separate lowpass FIR filters of the orders: $[37, 15, 6, 15, 37]$.

$$(f_{sl}^2 - f_{pr}^2)M^3 - (f_{sl} + f_{pr})^2M^2 + \tag{11}$$

$$+2f_d(f_{sl} + f_{pr})M - f_d^2 = 0 \tag{12}$$

$$D_{1,opt} = \frac{2M(1 - \sqrt{MF/(2-F)})}{2 - F(M+1)} \tag{13}$$

$$F = \frac{f_{sl} - f_{pr}}{f_{sl}} \tag{14}$$

$$M = D_1 \cdot D_2 \tag{15}$$

$$D_{2,opt} = \frac{M}{D_{1opt}} \tag{16}$$

Results

We will now be presenting the results of applying all three of our filters on one of the input EKG signals in the dataset. Additionally, we ran triangle and rectangle synthesized signal tests (EN 60601-2-51 standard [9]) on the IIR comb and multirate FIR filters.

Firstly, in Figure 9, we can see our initial EKG signal passing the lowpass FIR filter and IIR comb filter. The upper two plots show a signal with, both, high frequency and a low frequency drift components. We may also observe a frequency spike at 50Hz and 150Hz. These are the harmonic voltage components we want to remove. The middle plots feature our FIR filtered signal. Looking at the time domain, we observe a decrease in overall high frequency components through the signal. This is more evident when looking at the frequency domain, where effectively all frequencies beyond 50Hz were increasingly dampened. The final set of plots features the comb filter result. There is a noticeable change in the overall curvature of the signal as well as a massive decrease in high frequency components. The frequency domain has large dampening spikes at frequencies 50Hz,100Hz etc. These were the industrial voltage components we were trying to remove.

After additionally running triangle and rectangle synthesized tests, our comb filter had a 250 μV slope error in the rectangle test and a 3.3 percent dampening error in the triangle test. The rectangle test failed to meet the 150 μV maximum slope error requirement.

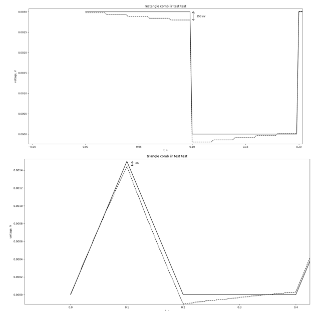

Figure 8. Comb IIR filter test on rectangle and triangle synthesized signals.

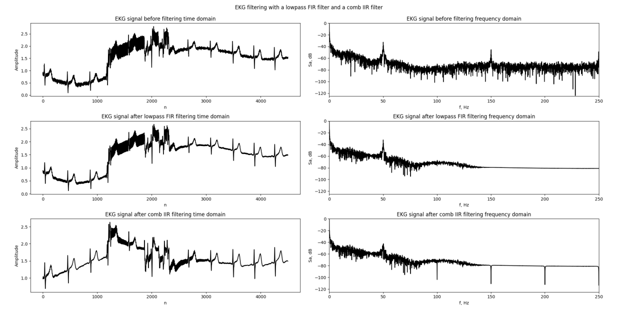

Figure 9. EKG signal filtering with a lowpass FIR filter and a comb IIR filter.

Next, we look at the multirate FIR filter results. Figure 10. The middle plot features the extracted drift component of the initial signal. We can observe a very clear correspondence between this component and the initial signal. The third plot is the result of subtracting the drift component, we can see a very clear elimination of all drift in the signal.

The rectangle test on our multirate filter has a 150 μV slope error at around 5 percent of the rectangles and under 50 μV in the rest, while the triangle test has a dampening error of $6 \cdot 10^{-5}$. The filter had a substantially higher test performance than the comb filter.

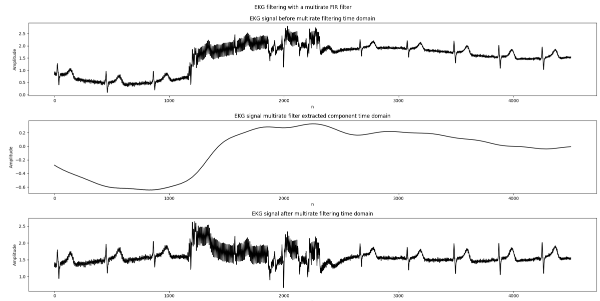

Figure 10. EKG signal filtering with a multirate FIR filter.

Conclusions

In this laboratory work, we designed, implemented and tested three separate digital signal processing filters: a.) a lowpass FIR filter, b.) a comb IIR filter, c.) a multirate FIR filter. The filters proved effective at eliminating signal noise from our EKG dataset. The comb filter did not pass the rectangle test, which we consider to be a result of an overly large attenuation rate. The multirate FIR filter, on the other hand, flawlessly passed the tests and eliminated signal drift with extreme precision.

Bibliography

[1] Malmivuo j., plonsey r. biolelectromagnetism. Link: https://bem.fi/book.

[2] Kopustinskas, a.; kopustinskienė, g. biofizika. technologija. 2009, p. 79-84.

[3] numpy python computing package. Link: https://numpy.org.

[4] scipy python scientific computing package. Link: https://www.scipy.org.

[5] matplotlib python package. Link: $https://matplotlib.org/api/pyplot_api.html.$

[6] Sörnmo l. , laguna p. bioelectrical signal processing in cardiac and neurological applications. academic press. 2005, p. 688.

[7] Laboratory work nr2 outline. ktu 2020 dsp module t121m501. accessed 05.11.2020.

[8] Milie l. multirate filtering for digital signal processing: Matlab applications. igi global, 2009, p. 414. Link: http://firasaboulatif.free.fr.

[9] En 60601-2-51 international standard. 2003.

Appendix

Source code