6.555 Lab1: The Electrocardiogram

Tony Hyun Kim

Spring 2011

1 Data acquisition

Question 1: Draw a block diagram to illustrate how the data was acquired.

The EKG signal discussed in this report was recorded according to the signal chain as shown in Fig. 1. The initial signal from the volunteer is amplified with a gain of 1000, then filtered (analog, non-ideal) with a bandpass over 0.1 to 100Hz. The ADC operates at a sampling rate of $f_s = 250$Hz, and uses 16-bit quantization with a maximum input voltage of $V_{max} = 5$V.

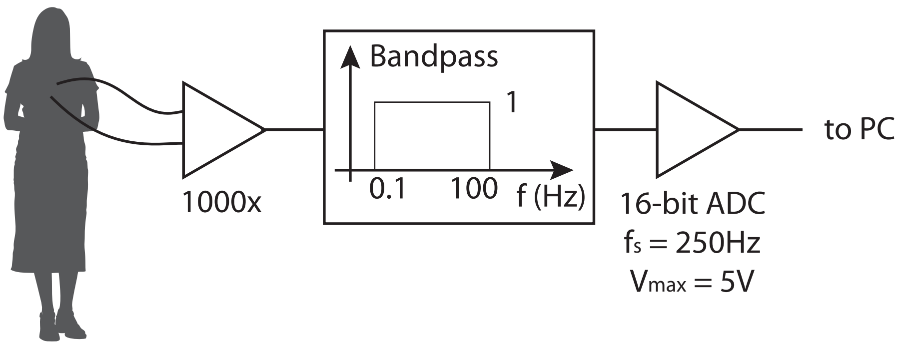

Figure 1: Block diagram ilustrating the data acquisition chain.

A complete EKG signal, as acquired by this system, is shown in Fig. 2. During the first four minutes, the signal was acquired under ideal conditions (the patient was supine and quiet). On the other hand, during the last minute, the patient added noise to the recording by periodic contraction of the chest muscles. The differences between clean and noisy segments are highlighted in two 10 second segments in Fig. 2. These clips are used as representative samples throughout the work.

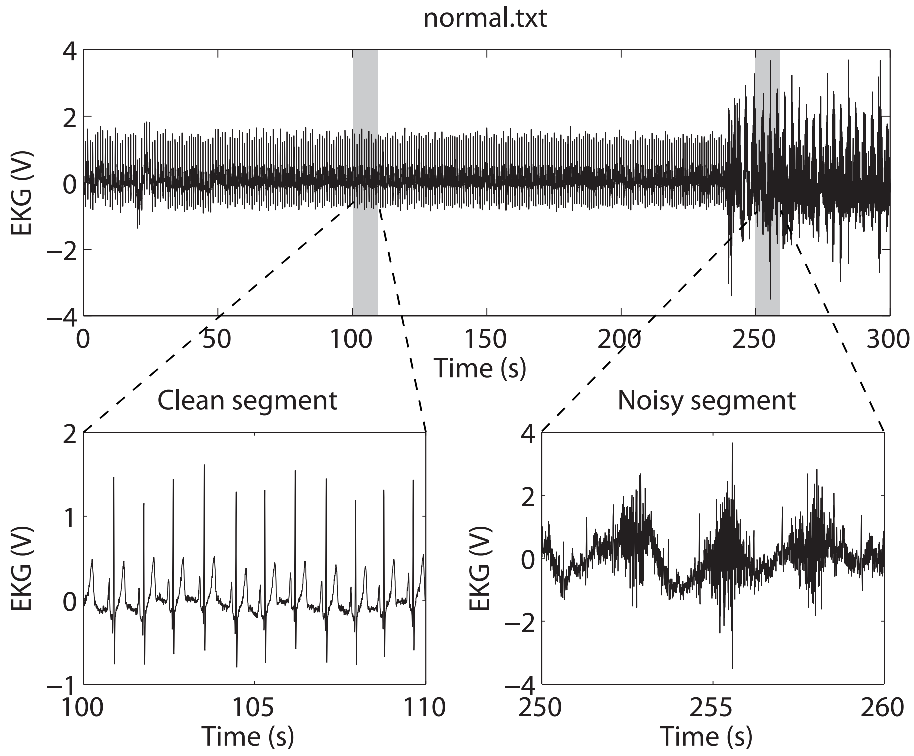

Figure 2: The complete EKG signal, "normal.txt". During the first four minutes of the recording, the patient was supine and quiet. During the last minute, the patient added noise to the recording by periodic contraction of the chest muscles.

Question 2: If you examine the data, you will notice that the ECG data values have been rounded to the nearest millivolt. Is this a result of the 16-bit quantization?

With a $B = 16$-bit ADC, and a symmetric input range of $V_{max} = 5$ V, we expect a quantization step of $Q = V_{max}/2^{B-1} = 0.15$ mV. Hence, the millivolt rounding is additional loss of resolution beyond the 16-bit quantization.

Question 3: Why was the signal filtered prior to sampling? Why was a cutoff frequency of 100 Hz used (instead of 125 Hz)?

An analog filter with a cutoff at $f_s/2$ is required in order to prevent aliasing in analog to digital conversion, where f_s is the sampling frequency used. The filter must be applied in the analog domain, where it is impossible to achieve a perfect (rectangular) LPF. Hence, we use a cutoff frequency at 100 Hz, in order to account for the finite width of the transition band.

2 Signal conditioning and Noise reduction

Question 4: I began my design of the bandpass filter by contrasting the power spectral densities (PSD) of the clean and noisy segments, as shown in Fig. 3.

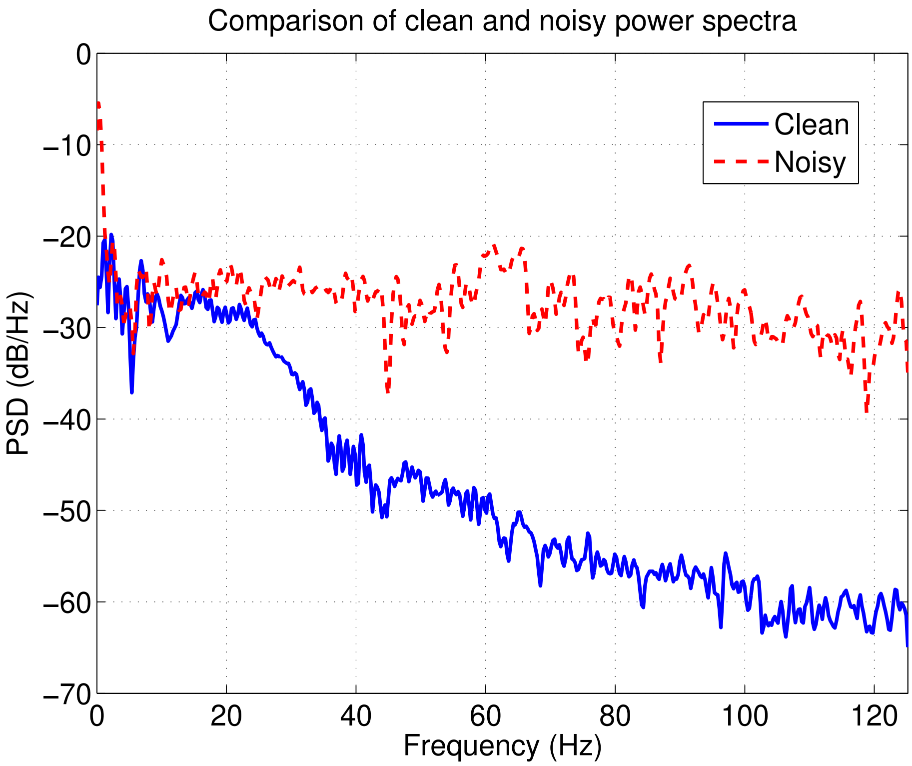

Figure 3: Comparison of the spectral densities of the clean (solid blue curve) and noisy (dashed red curve) segments.

Two features are immediately visible:

- The noisy segment has significant signal content below 5 Hz over that of the clean segment. This owes to the significant low-frequency baseline-wander that is visible in the time domain.
- The clean and noisy PSDs are easily distinguished in the band $30 - 125$ Hz. The clean signal has effectively no content in this band, whereas the noisy signal is marred by white noise (a roughly constant noise spectrum).

These observations motivated me to design a bandpass filter with a passband roughly over $5 - 30$ Hz.

The filter was designed using the window method, utilizing a Hamming window function. I found that 256-point filter with a bandpass over $5 - 25$ Hz worked well. The filter $h[n]$ is illustrated in Fig. 4.

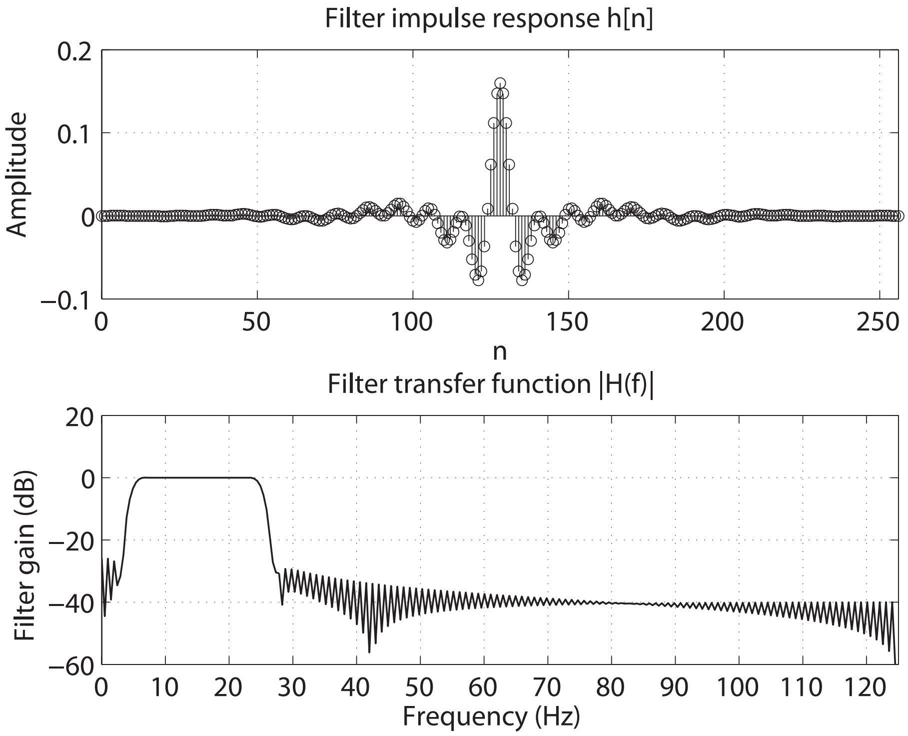

Figure 4: A 256-point bandpass ($5 - 25$ Hz) filter for the conditioning of noisy EKG segments.

One major deviation between my bandpass specification and the realized filter is in the finite transition band. The apparent width of approximately 5 Hz is consistent with a $N = 256$ point Hamming window (with $f_s = 250$ Hz).

In addition, note that the filter $h[n]$ is centered at $n_0 = 128$, rather than at $n_0 = 0$. At a sampling period of $T_s = 1/f_s = 1/250$ s, this implies that the filter will delay the input by $T_{\text{delay}} = T_s \cdot n_0 = 0.512$ seconds.

Question 5:

The effect of the bandpass filter $h[n]$ on both clean data is illustrated in Fig. 5 where I have superimposed the original and filtered signals in both time- and frequency-domains. In the time domain, I manually undelayed the filtered signal by T_{delay}.

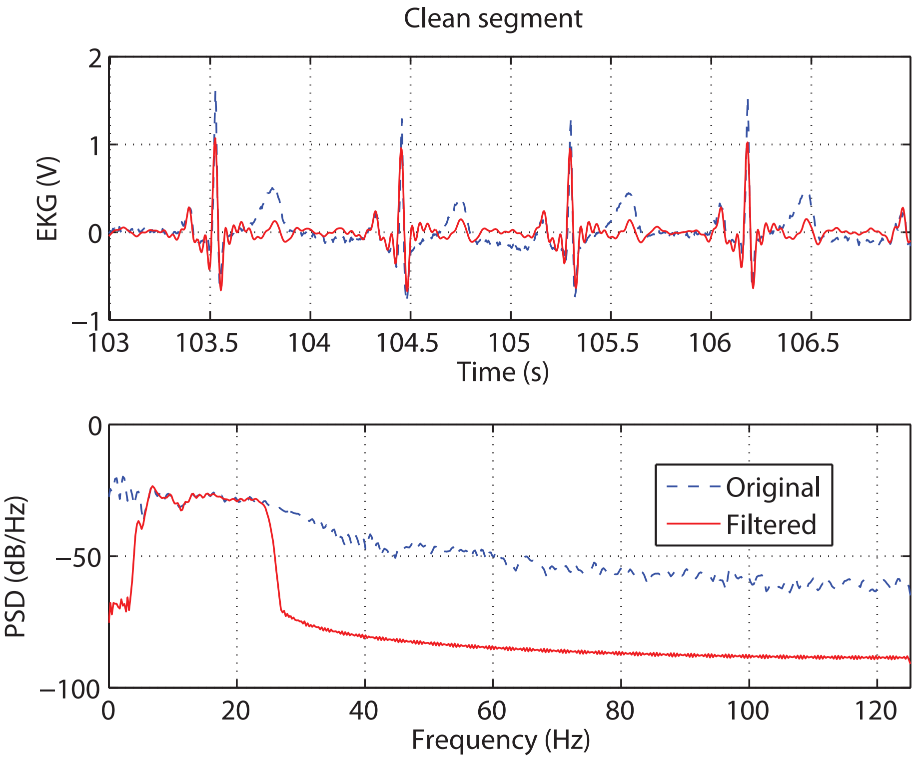

Figure 5: Effect of filtering a **clean** EKG segment by the bandpass filter $h[n]$ in the time and frequency domains. The original signal is indicated by dashed blue lines, whereas the filtered output is shown as solid red curves.

Based on Fig. 5, I find that the basic structure of the clean EKG signal is preserved by the bandpass filter $h[n]$. However, there is a notable reduction of amplitude on the T-wave. The T-wave has a time-domain duration of approximately 0.2 seconds, which corresponds to 5 Hz, which is the lower-edge of the filter passband. While one could extend the passband to include 5 Hz, the system would then more susceptible to the low-frequency noise present in the noisy segment. In any case, the current system clearly preserves the QRS-complex, which is the important feature often used in EKG analysis algorithms.

Fig. 6 shows the bandpass filter $h[n]$ applied to the noisy segment of the signal.

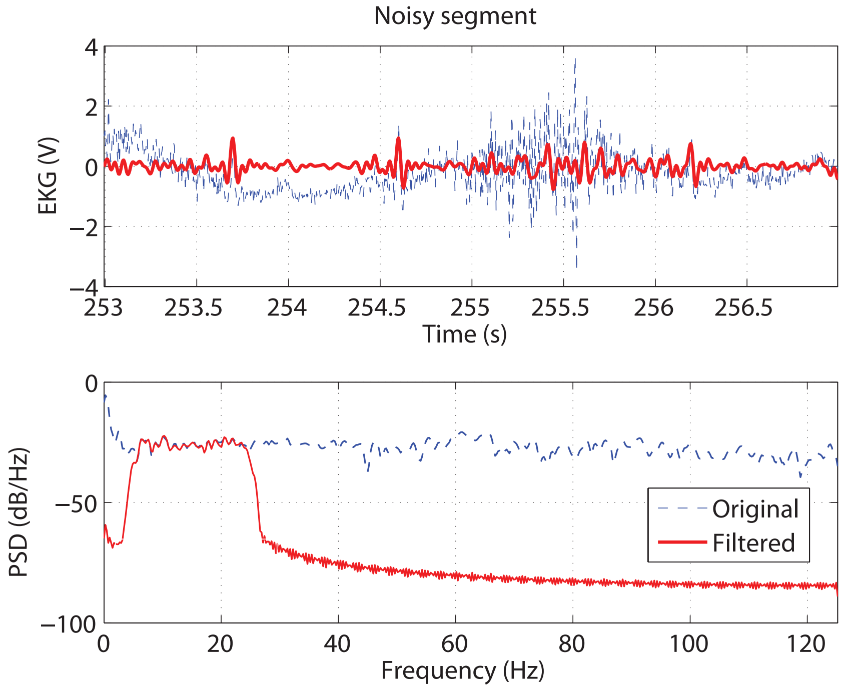

Figure 6: Effect of filtering a **noisy** EKG segment by the bandpass filter $h[n]$ in the time and frequency domains. The original signal is indicated by dashed blue lines, whereas the filtered output is shown as solid red curves.

The bandpass filter effectively removes the baseline wander of the noisy segment, and greatly reduces the amplitude of the noise pulse around $t = 255 - 256$ s. Furthermore, the QRS complexes are more readily identified in the filtered output, and shows a heart rate of approximately 1 Hz, which is in agreement with the periodicity observed in the clean segment.

Question 6:

On the other hand, the bandpass approach turns out to be inadequate to unambiguously reveal the EKG waveform underneath the noisy pulse during $t = 255 - 256$ s (Fig. 6). Even with an ideal, rectangular filter, we cannot expect perfect removal of noise. For instance, the QRS-complex has a time-width of roughly 0.03 s, or 30 Hz in frequency. Thus, at 30 Hz, it will not be possible to distinguish between QRS-complex and noise. This is evident by examining the filtered output around $t = 255 - 256$ s, which shows oscillations around the "QRS frequency." We cannot remove this oscillation without throwing out the QRS feature along with it!

3 Ventricular Arrhythmia Detection

In analyzing the ventricular arrhythmia dataset, I found it useful to perform some preprocessing. Namely, I:

- Removed the baseline wander by pre-filtering the EKG waveform with a HPF with a cutoff at 3 Hz. I chose the lowest cutoff value that seemed to reliably remove wander for all files in the provided dataset.
- Normalized the EKG amplitude by normalizing to the total energy in the signal. Note that in a real-time application, one would need to compute a running estimate of the signal energy, rather than computing the total energy of a recorded signal as I have done here. The normalization was performed, because I found that each of the files had wildly varying amplitudes. Obviously, the normalization was performed after removing the baseline!

All signals presented in this section have been pre-processed as described above.

Question 7: Choice of parameters for spectral analysis

I performed arrhythmia detection on segments of 10 seconds. At a sampling rate of $f_s = 250$ Hz, this divides the total data vector into chunks of $N_{\text{data}} = 2500$.

The spectrum was computed with a Hamming window of length of $N_w = 1024$. I used $N_{\text{fft}} = 1024$, after observing that a FFT performed with $N_{\text{fft}} = 2048$ did not show any appreciable changes in the DFT – implying that I was not undersampling the DTFT. The Matlab command is:

```
>> pwelch(data,Nwindow=1024,Noverlap=[],Nfft=1024,fs=250);
```

where an unspecified N_{overlap} specifies 50% window overlap in the Welch method.

The spectral resolution for a Hamming window (in DT frequency) is $\Delta F = 4/N = 4/1024$, which can be converted to resolution in real frequency as: $\Delta f = f_s \cdot \Delta F \approx 1$ Hz. This is corroborated in the spectra that I obtained.

Question 8: How do the ventricular arrhythmia segments differ from the normal segments in both the time and frequency domains?

The most striking feature of ventricular arrhythmia is that the EKG segment appears to be nearly a pure sinusoid, which is evident in both time and frequency domains. This is demonstrated in Fig. 7, which shows two episodes of ventricular fibrillation and a normal sinus rhythm for comparison. Note that the amplitudes have been normalized, and are thus marked as arbitrary units "(a.u.)".

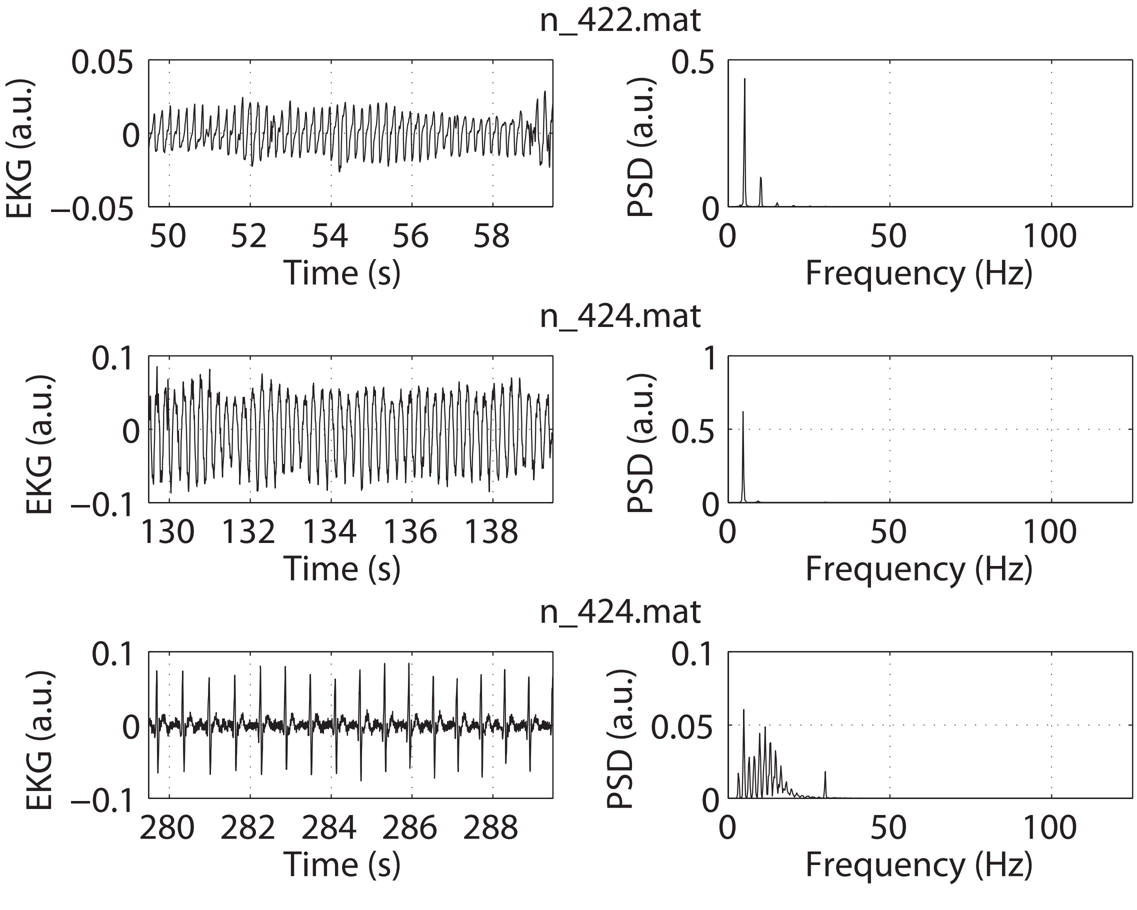

Figure 7: Two examples showing the sinusoidal characteristics of ventricular fibrilation, and one normal sinus rhythm for comparison.

Hence, my strategy for arrhythmia detection was to detect, in the frequency domain, the presence of a dominant single sinusoid.

Question 9: Describe your approach to distinguishing ventricular arrhythmias from normal rhythms.

My general approach was to detect when the EKG signal devolves into a pure sinusoid. A flowchart that describes the process is shown in Fig. 8.

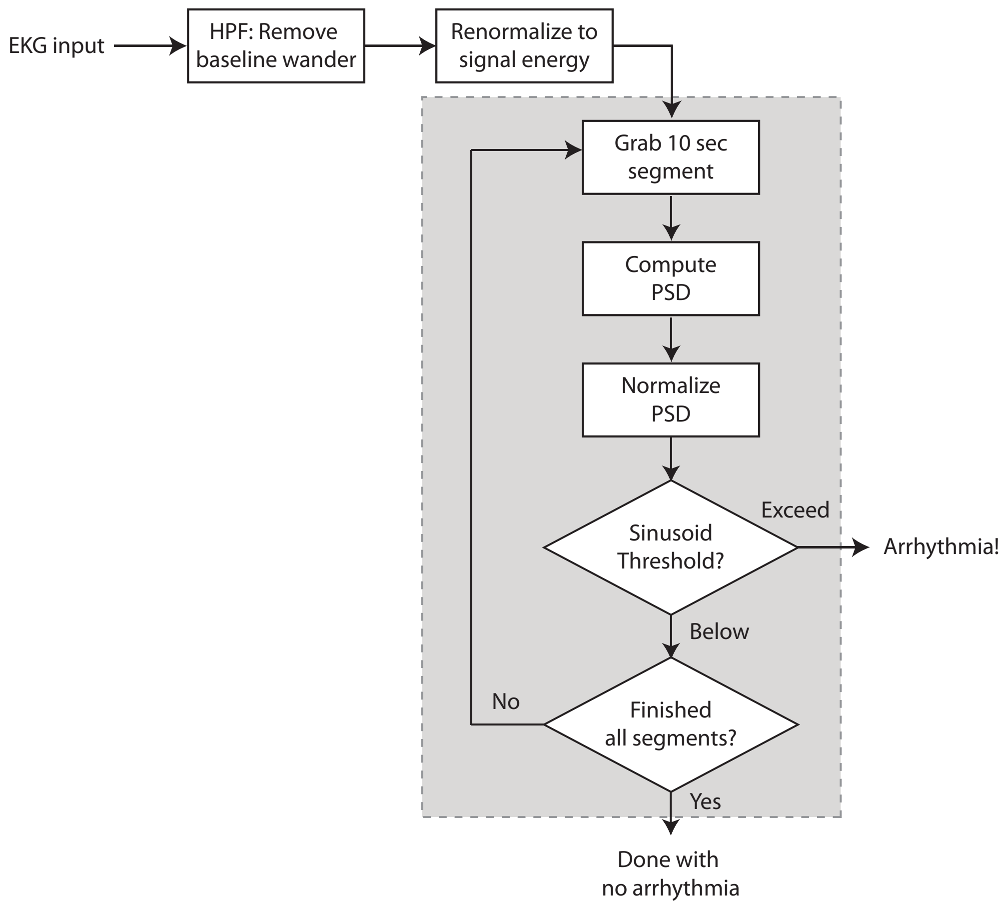

Figure 8: A flowchart for a rudimentary ventricular arrhythmia detection algorithm.

By "normalization of the PSD", I sum the PSD over all sampled frequency points, and divide the PSD vector by this sum. Note that a perfect sinusoid would result in a unity amplitude by this process (up to a multiplicative factor arising from the mainlobe width associated with the window used). On the other hand, if there is a broadband spectral content, as in the case of a clean EKG signal, the individual amplitudes will be bounded by roughly the inverse of the signal bandwidth. The PSDs of Fig. 7 demonstrate this behavior. I experimentally determined a good "sinusoid threshold" that based on the arrhythmia dataset provided.

Source	Expert	My algorithm
n_422.mat	Ventricular tachycardia (VT) at $t \approx 43$ s	Detected in $[44, 54]$ s
n_424.mat	Ventricular fibrillation (VF) at $t \approx 110$ s	Detected in $[107, 117]$ s
n_426.mat	VF at $t \approx 105$ s	Detected in $[106, 116]$ s
n_429.mat	Ventricular flutter (VFL) at $t \approx 115$ s	Detected in $[111, 121]$ s
n_430.mat	VT at $t \approx 130$ s	Detected in $[132, 143]$ s
n_421.mat	Normal rhythm with noise	No detection
n_423.mat	Atrial fibrillation at $t \approx 16, 60$ s	No detection

Table 1: Comparison of algorithm performance against human expert classification.

Question 10: How well does your detector perform? Compare the output of the system against the expert classification.

I find that this rudimentary algorithm works quite well! See the comparison against a human expert classification in Table 1.

The algorithm produced a **missed detection** for `n_423.mat`. Presence of noise will interfere with the pure sinusoid detection, since it will reduce the normalized amplitude of the fibrillation sinusoid.

Question 11: How would you attempt to improve your detector?

I find the strategy of pure sinusoid detection promising. As noted above, however, this detection is susceptible to noise in the input signal, and will tend to miss detections in such cases. Thus, I would attempt to more precisely detect and remove noise in the pre-processing steps.

Question 12: Most important thing learned in this exercise

I gained an ability to quickly deal with all types of normalized frequencies in the sampled DT domain. I also can quickly relate these normalized frequencies to real frequency by use of the sampling frequency.

Question 13: What did you like/dislike about this lab exercise?

We should have recorded our own EKG data!