ECG signal processing

Abstract

Digital signal processing and data analysis are all methods that are very common in a biomedical engineering. For example, electrocardiogram (ECG) signals are obtained by the data acquisition method and after that the filtering is performed to remove any interference. The filter is designed to remove wideband additive Gaussian noise, breathing muscle activity, and the power supply network artifacts at 50Hz. In this example, we start from with the description of the ECG signal, signal acquisition, and we then proceed further to digital filter designing. At the end we will illustrate the heart rate frequency detection algorithms. Data acquisition and processing are done by using MatDeck. The example in this document displays results correctly if the document "ecgsenderg.mdd" is open and evaluated first. The document "ecgsenderg.mdd" simulates the ECG device which sends data using TCP/IP channels for all nine leads.

1. Introduction

Electrocardiogram(ECG) represents the electrical behavior of a human heart. This signal could be measured by electrodes from the human body in typical engagements. Signals from these electrodes are then brought to simple electrical circuits with amplifiers and analogue – digital converters. A typical position of the electrodes are shown in Fig. 1. The artifacts of digitalized signal is the interference with other signals like the power supply voltage at 50 Hz frequency and the breathing muscle artifacts. These elements are considered to be noise and have to be removed before the signal is used for the next stage of data processing. Digital filters are used to process the ECG signal. DSP filters should be designed for effective real-time application. The final stage of ECG processing is commonly heart rate detection which is based in the QRS complex detection and heart rate is computed like the distances between QRS complexes.

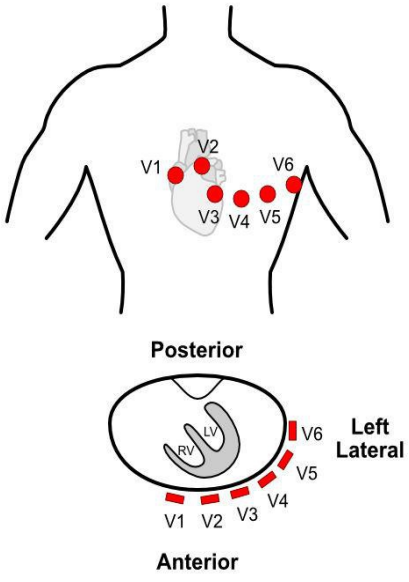


Fig. 1. Six chest (precordial) electrodes for ECG acquisition

2. Signal acquisition

ECG signals are acquired by the lead/electrode method. The most common system is the so-called standard 12-lead system, which consists of 10 electrodes. In order to measure the heart's electrical

activity accurately, proper electrode placement is crucial. Six Chest (Precordial) Electrodes are placed according to Fig. 1 and there are four Limb (Extremity) Electrodes which are placed on the left arm, left leg, right arm and right leg. The signal from the electrodes is combined to form 12 leads. After that, the signal is amplified and digitized. The sampling frequency is usually up to 1000Hz, as ECG is a low frequency signal. Twelve lead signals are illustrated in Fig. 2 for a normal heart rate.

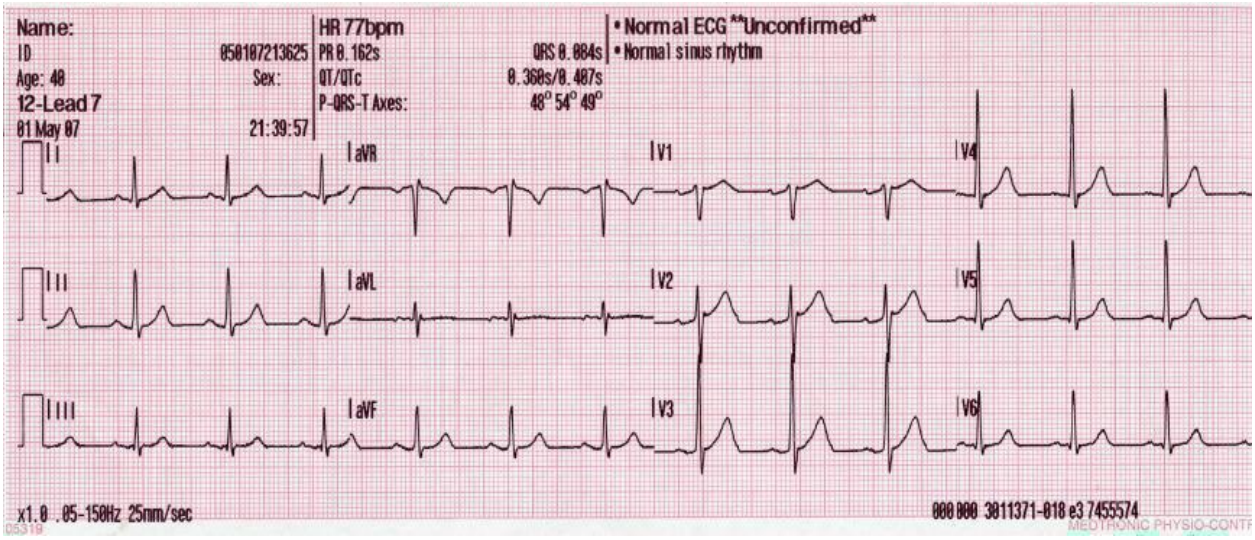


Fig. 2. 12 lead signals from common diagnostic equipment

The process explained above is done in real-time but only with specific hardware, which can be connected to MatDeck. However, the purpose of this example is to illustrate data processing and thus how we generate the ECG signal in real-time. The signal generator is used to send signal from the nine leads V1-V6, and I, II and III. The signal is read into specific variables and then plotted in Graphs. The lead ,V4, is treated as an ECG signal for further processing

```
V1 := 0   V2 := 0   V3 := 0   V4 := 0   V5 := 0   V6 := 0   I := 0   II := 0   III := 0
```

3. Digital ECG signal processing with digital filters

In this part we show how noise elements, and baseline wonder are removed from the ECG signal by means of a digital filter. The main noise in ECG apparatus is the power line network at 50Hz and the breathe muscle movements. These artifacts must be removed before the ECG signal is fed to the next stage where the heart rate frequency is determined. Beside the basic filtering, the R peaks are detected from the ECG signal which are needed for heart rate detection. The filtering is performed by the 4th order bandpass Butterworth filter with cut-off frequencies of 0.5 and 30 Hz. MatDeck provides all necessary functions to design aforementioned Butterworth filter and to filter the ECG signal in real time.

We start by designing the 4th order bandpass Butterworth filter with cut-off frequencies of 0.5 and 30 Hz, and the sampling rate is 1000Hz.

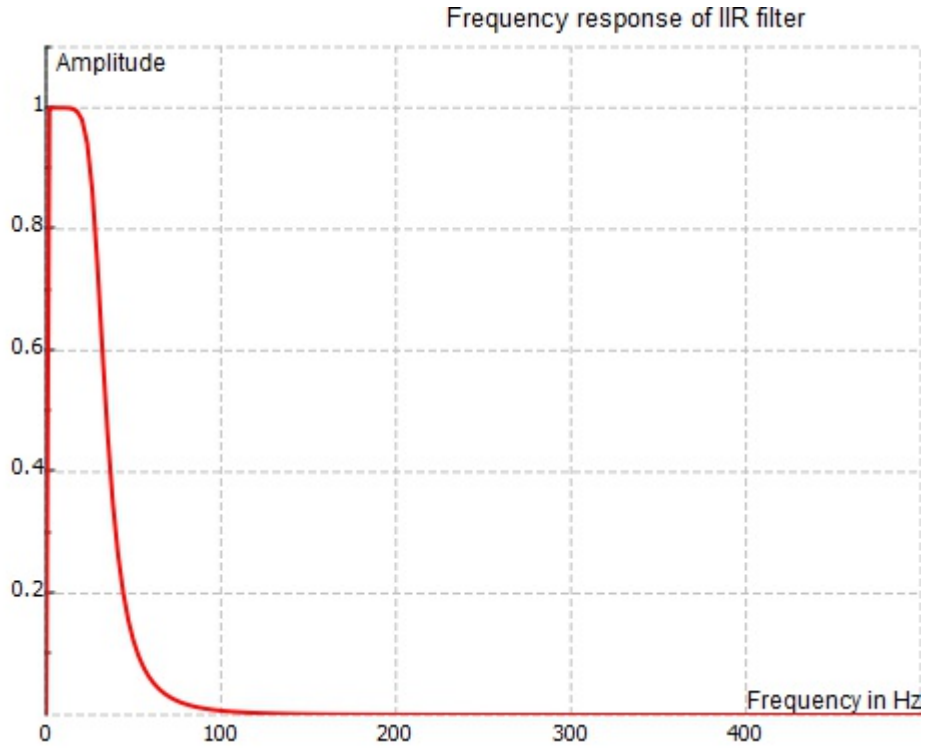
```
fc1 := 0.5        fc2 := 30        Fs := 1000

Coeff := buttband(4 , "pass" , fc1 , fc2 , Fs)
```

fc1, fc2 cutoff frequencies in Hz
Fs sampling frequency in Hz
Coeff Matrix of filter coefficients

Next, we plot the amplitude response of the obtained filter, in order to verify design.

```
Ac := col2vec(Coeff , 0)        Bc := col2vec(Coeff , 1)
Hf := iirfreqres(Bc , Ac , 512 , 1)
Hg := join mat cols(col2vec(curve2d(x , 0 , Fs/2 , 512) , 0) , |Hf|)
```

Ac	numerator coefficients of IIR filter
Bc	denominator coefficients of IIR filter
Hf	frequency response of IIR filter
Hg	matrix, magnitude response vs frequency



The filter is applied to lead V4 in order to detect R peaks. The designed Butterworth filter is applied in real time on data as they arrive from device and the filter is initialized before in order to keep states between chunks of data. The filtered signal is plotted in real time.

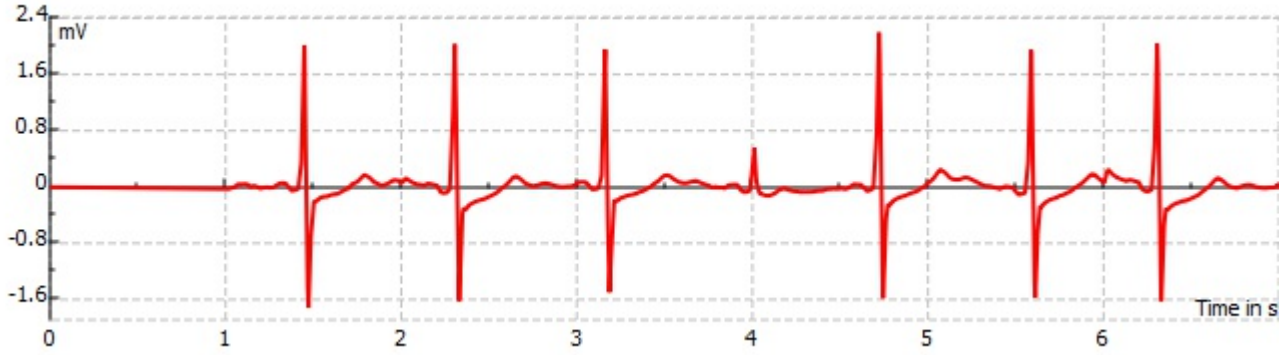
4. Heart rate detection

In order to detect heart rate, the peaks are determined first. The function peaks() finds the local maxima of the filtered ECG signal. Therefore, the thresholding is performed and single peaks corresponding to heart beats are detected. The ECG peaks are stored in ECGp and compared to 1mV thresholding. The heart rate is derived from time-stamps of the detected peaks in the real time. The thresholding operation is defined in the code above, together with the filtering.

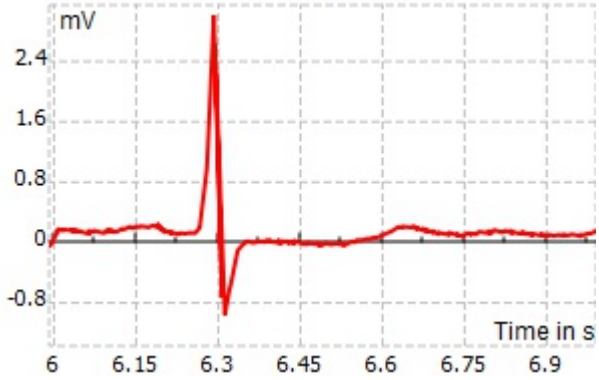
5. Conclusions

This example has shown capabilities of MatDeck software in a very common real-time scenario of ECG signal acquisition and processing. The 9-lead ECG signal is read through nine TCP/IP channels for acquisition. After that, the fourth order Butterworth filter is designed and lead V4 is selected for filtering. Further, the peak detection and heart beat rate determination are performed. As a conclusion, MatDeck shows satisfactory performance for real-time digital signal acquisition and processing,

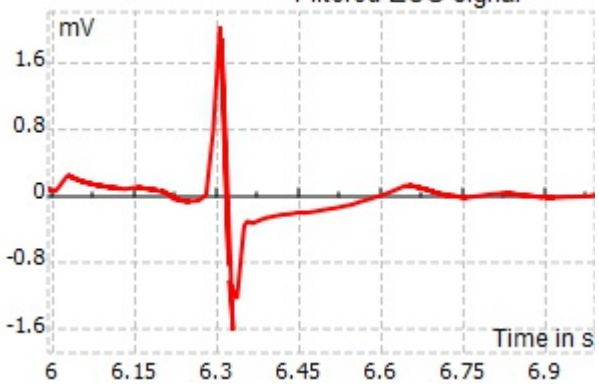
Final ECG signal



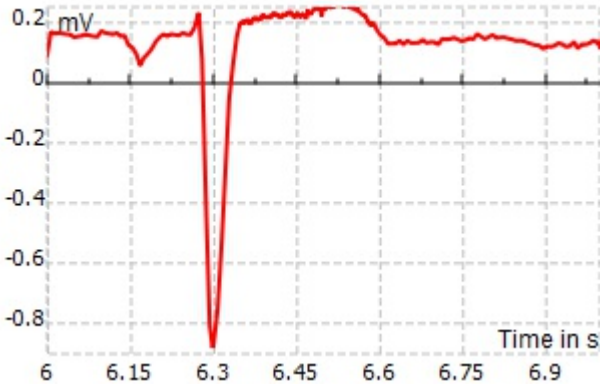
V4 lead

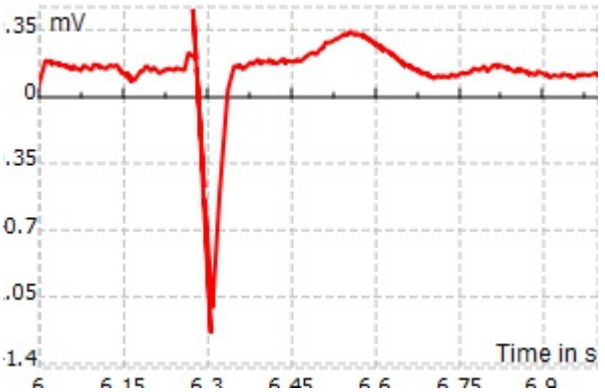


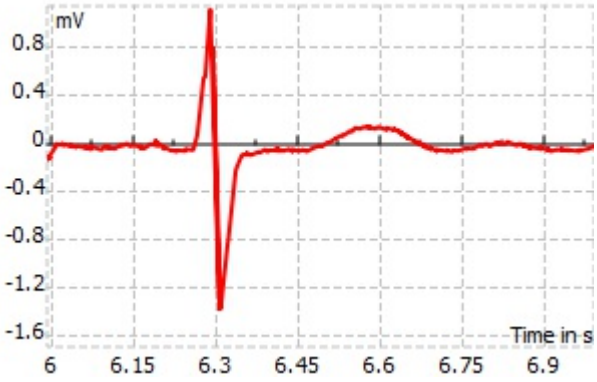
Filtered ECG signal

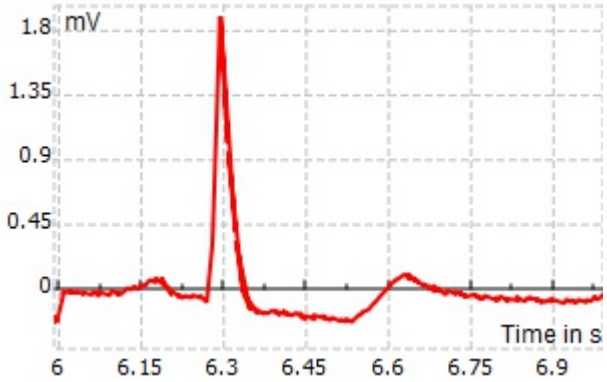


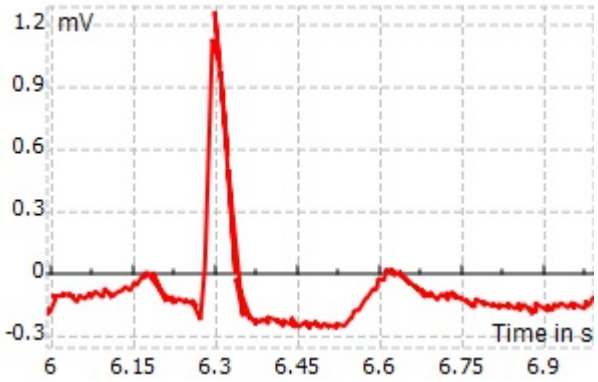
Heart beats per minutes

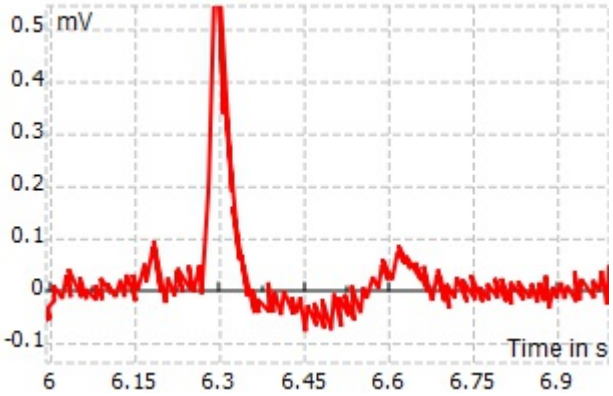

mV
0.2
0
-0.2
-0.4
-0.6
-0.8
Time in s
6
6.15
6.3
6.45
6.6
6.75
6.9

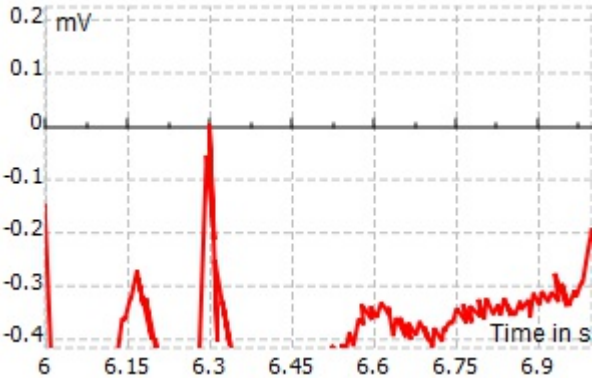

mV
.35
0
.35
0.7
.05
1.4
Time in s

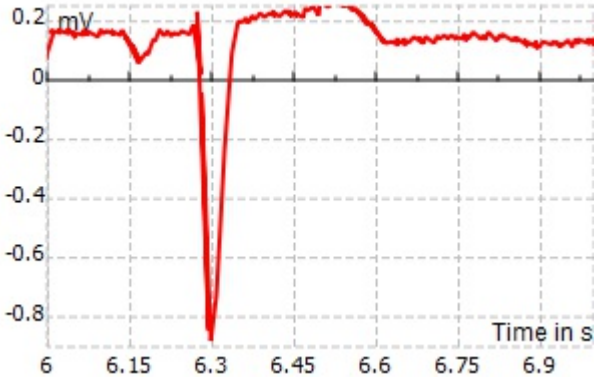

mV
0.8
0.4
0
-0.4
-0.8
-1.2
-1.6
Time in s
6
6.15
6.3
6.45
6.6
6.75
6.9


mV
1.8
1.35
0.9
0.45
0
Time in s
6
6.15
6.3
6.45
6.6
6.75
6.9


mV
1.2
0.9
0.6
0.3
0
-0.3
Time in s
6
6.15
6.3
6.45
6.6
6.75
6.9


mV
0.5
0.4
0.3
0.2
0.1
0
-0.1
Time in s
6
6.15
6.3
6.45
6.6
6.75
6.9


mV
0.2
0.1
0
-0.1
-0.2
-0.3
-0.4
Time in s
6
6.15
6.3
6.45
6.6
6.75
6.9


mV
0.2
0
-0.2
-0.4
-0.6
-0.8
Time in s
6
6.15
6.3
6.45
6.6
6.75
6.9

```
ECG := 0        j := 0        HeartBeats := [0  60]        ECGfinal := 0        ECGfinal1 := [0  0]
initiirfilter(0 , size(Bc))

filtering( )
{
  ECGg := iirfilter(0 , col2vec(V4 , 1) , Bc , Ac)
  ECG = join mat cols(col2vec(V4 , 0) , ECGg)
  ECGp := peaks(ECG)
  THold := 1
  for(i := 0 , i < size(ECGp) / 2 , i += 1)
  {
    ind0 := 2 i
    ind1 := 2 i + 1
    if(ECGp[ind1] > THold)
    {
      tindp := 2 j
      HeartBeatsp := [0  0]
      HeartBeatsp[0] = ECGp[ind0]
      HeartBeatsp[1] = ECGp[ind0] - HeartBeats[tindp]
      HeartBeatsp[1] = 60 / HeartBeatsp[1]
      if(j == 0)
      {
        HeartBeatsp[1] = 60
      }
      HeartBeats = join mat rows(HeartBeats , HeartBeatsp )
      j = j += 1
    }
  }
  ECGfinal1 = join mat rows(ECGfinal1 , ECG)
  if(size(ECGfinal1) / 2 > 10000 )
  {
    r1 := size(ECGfinal1) / 2 - 10000
    r2 := r1 + 9999
    ECGfinal = subset(ECGfinal1 , r1 , 0 , r2 , 1)
  }
  if(size(ECGfinal1) / 2 <= 10000 )
  {
    ECGfinal = ECGfinal1
  }
}

fn( )
```

ECG	filtered signal
j	number of beats
filtering()	functions performs filtering and peak detection
HeartBeats	number of heart beats per second
V4	lead signal used for processing
THold	threshold value
ECGp	peaks in the filtered ECG signal
ind0, ind1	indices to access ECGp matrix cells
r1, r2	indices of rows to extract part of signal
HeartBeatsp	single peak detected and processed
ECGfinal	keeps last 10 seconds of signal
ECGfinal1	keeps whole signal during recording

```
fn()
{
    g1 := channel connect("127.0.0.1", 1805)
    g2 := channel connect("127.0.0.1", 1806)
    g3 := channel connect("127.0.0.1", 1807)
    g4 := channel connect("127.0.0.1", 1808)
    g5 := channel connect("127.0.0.1", 1809)
    g6 := channel connect("127.0.0.1", 1810)
    g7 := channel connect("127.0.0.1", 1811)
    g8 := channel connect("127.0.0.1", 1812)
    g9 := channel connect("127.0.0.1", 1813)
    c := 0
    while(true)
    {
        v1 := channel read(g1, true)
        v2 := channel read(g2, false)
        v3 := channel read(g3, false)
        v4 := channel read(g4, false)
        v5 := channel read(g5, false)
        v6 := channel read(g6, false)
        i := channel read(g7, false)
        ii := channel read(g8, false)
        iii := channel read(g9, false)
        n := size(v1)
        t := 1/n
        c += (n t)
        d1 := col2vec(curve2d(x, c, c+(n-1)*t, n), 0)
        V1 = join mat cols(d1, v1)
        V2 = join mat cols(d1, v2)
        V3 = join mat cols(d1, v3)
        V4 = join mat cols(d1, v4)
        V5 = join mat cols(d1, v5)
        V6 = join mat cols(d1, v6)
        I = join mat cols(d1, i)
        II = join mat cols(d1, ii)
        III = join mat cols(d1, v1)
        filtering()
    }
}
```

fn ()	function which performs all operations
g1-g9	channel ids
v1-v6	temporary ECG leads data
i-iii	temporary ECG leads data
c	current time
n	buffer length, same as sampling frequency
V1-V6	ECG leads data including time stamps
I-III	ECG leads data including time stamps