

Elektroniczne systemy diagnostyki medycznej i terapii

Neurocybernetyka

5 lutego 2014

Spis treści

1	Lista modułów	5
1.1	GUI	6
1.1.1	Wstęp	6
1.1.2	Okno główne	6
1.2	MVC	10
1.2.1	Moduł wejść	10
1.2.2	Struktura aplikacji.	10
1.2.3	Logi	11
1.2.4	Skrypt przetwarzający dane.	13
1.3	Moduł R_PEAKS	14
1.3.1	Badania literaturowe	14
1.3.2	Koncepcja proponowanego rozwiązania	15
1.3.3	Rezultaty i wnioski	19
1.3.4	Diagramy klas	32
1.3.5	Literatura	34
1.4	Moduł VCG_T_LOOP	35
1.5	Moduł SIG_EDR	36
1.5.1	Badania literaturowe	36
1.5.2	Koncepcja proponowanego rozwiązania	38
1.5.3	Rezultaty i wnioski	39
1.5.4	Diagram klasy	41
1.6	Moduł WAVES	43
1.6.1	Badania literaturowe	43
1.6.2	Koncepcja proponowanego rozwiązania	44
1.6.3	Rezultaty i wnioski	50
1.6.4	Diagramy klas i sekwencji	54
1.7	Moduł HRV1	56
1.7.1	Badania literaturowe	56
1.7.2	Koncepcja omawianego rozwiązania	56
1.7.3	Realizacja i wnioski	58
1.7.4	Diagramy klas i sekwencji	64
1.8	Moduł QRS_CLASS	67
1.8.1	Wstęp	67
1.8.2	Koncepcja proponowanego rozwiązania	67
1.8.3	Diagramy klas	69
1.8.4	Rezultaty i wnioski	69
1.9	Moduł ST_INTERVAL	71

1.9.1	Badania literaturowe	71
1.9.2	Koncepcja proponowanego rozwiązania	72
1.9.3	Rezultaty i wnioski	75
1.9.4	Diagramy klas	83
1.10	Moduł SLEEP_APNEA	84
1.10.1	Badania literaturowe	84
1.10.2	Koncepcja proponowanego rozwiązania	84
1.10.3	Rezultaty i wyniki	86
1.10.4	Literatura	92
1.10.5	Diagram klas	93
1.11	Moduł ATRIAL_FIBR	94
1.11.1	Badania literaturowe	94
1.11.2	Koncepcja proponowanego rozwiązania	95
1.11.3	Rezultaty i wnioski	97
1.12	Moduł QT_DISP	101
1.12.1	Diagramy klas i sekwencji	101
1.12.2	Badania literaturowe	103
1.12.3	Realizacja projektu	103
1.12.4	Rezultaty i wnioski	104
1.13	Moduł HRT	107
1.13.1	Badania literaturowe	107
1.13.2	Koncepcja proponowanego rozwiązania	108
1.13.3	Rezultaty i wnioski	111

Rozdział 1

Lista modułów

1.1 GUI

1.1.1 Wstęp.

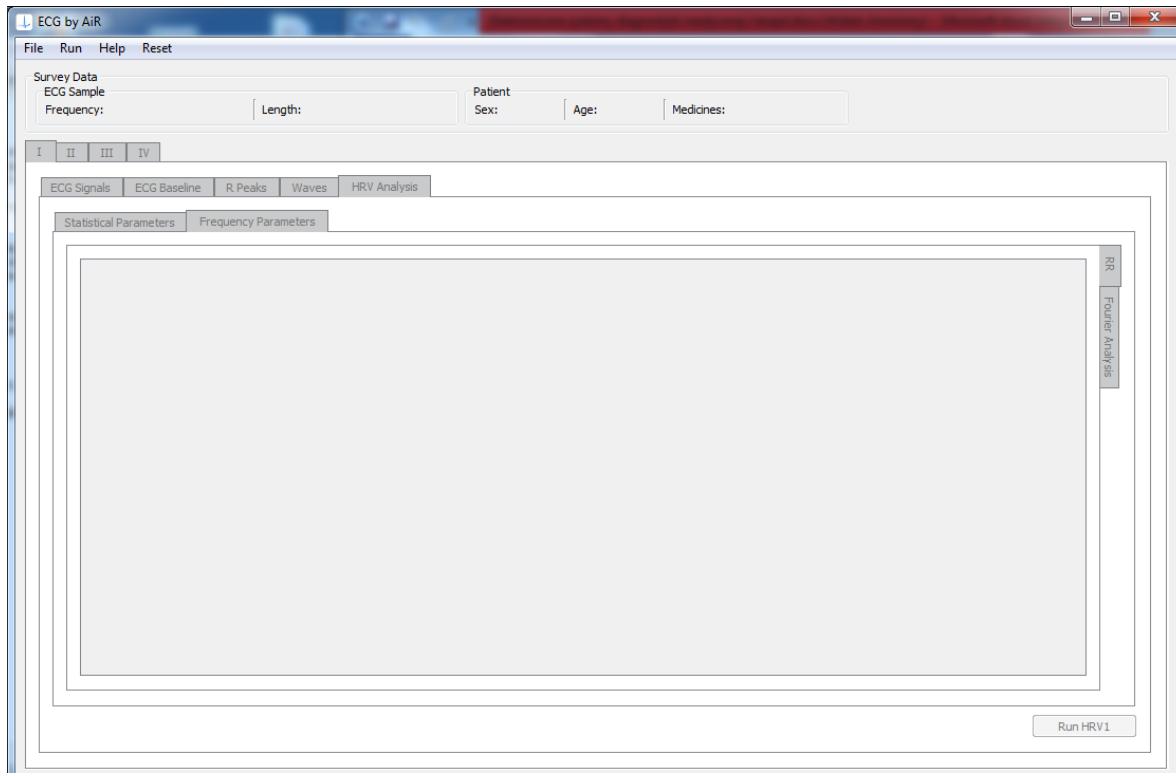
Zadanie polegało na dostosowaniu interfejsu graficznego dla aplikacji służącej do wyświetlania sygnału oraz prezentowania działania algorytmów przetwarzających sygnały EKG. Zgodnie z założeniami projektu aplikacja została napisana w języku C++ przy użyciu Qt, do edycji formularzy zostało użyte narzędzie Qt Creator.

Projekt składa się z jednego okna głównego oraz 3 okien dialogowych:

- Główne okno aplikacji – airecgmain
- Okno dialogowe wybór danych – filebrowser
- Okno dialogowe zawierające informacje o aplikacji – about

1.1.2 Okno główne

Zalecana rozdzielcość ekranu do efektywnego wyświetlania wyników to 1200x800, minimalny rozmiar okna głównego to 1024x768.



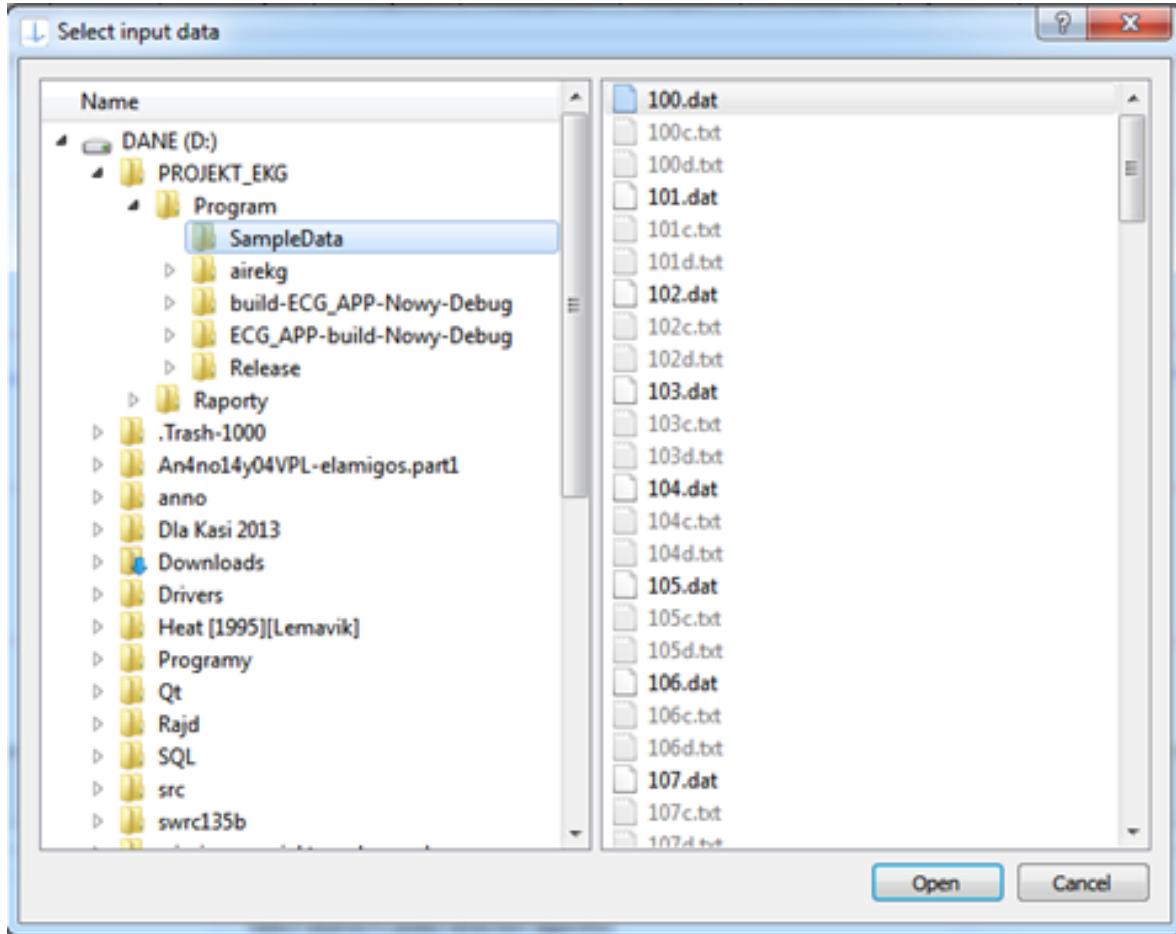
Rysunek 1.1: Okno główne programu.

Na górze okna głównego znajduje się Menu aplikacji, o następującej strukturze:

- File

- Open
 - Exit
-
- Run
 - Run ECG Baseline
 - Run R Peaks
 - Run Waves
 - HRV Analysis
 - QRS Classification
 - HRT Analysis
 - ST Segment
 - VCG Loop
 - Sig EDR
 - Atrial Fibr
 - QT Disp
 - Sleep apnea
 - RUN ALL
-
- Help
 - About

Opcja File -> Open powoduje uruchomienie okna dialogowego filebrowser, w którym możliwe jest wybranie zestawu danych do przetwarzania.



Rysunek 1.2: Wybór pliku z danymi.

Opcje z grupy Run służą do wywoływania poszczególnych modułów przetwarzających sygnał EKG. Ostatnią z opcji jest RUN ALL wybranie jej powoduje uruchomienie wszystkich modułów EKG w odpowiedniej kolejności. Podczas obliczania odpowiedzi, w dolnej części ekranu pokazywany jest pasek informujący o trwającym przetwarzaniu sygnału.

W grupie Help znajduje się opcja About powoduje ona uruchomienie okna dialogowego, zawierającego informacje o twórcach programu.

Poniżej menu znajdują się pola zawierające dane o sygnale EKG oraz badanym pacjencie:

- Częstotliwość pomiaru (Frequency)
- Długość pomiaru (Length)
- Płeć (Sex)
- Wiek (Age)
- Przyjmowane leki (Medicines)

Po wczytaniu danych, centralną część ekranu zajmuje wykres, przedstawiający sygnał EKG, za pomocą zakładek znajdujących się nad nim możemy przechodzić do poszczególnych modułów. Moduły zostały podzielone na 4 zakładki w celu ułatwienia poruszania się po programie. W

zależności od wybranego modułu, pod wykresem, znajdują się dostępne opcje konfiguracyjne. W każdej zakładce znajduje się przycisk Run pozwalający odpalić konkretny moduł, zaś na dole ekranu do dyspozycji użytkownika jest przycisk RUN ALL uruchamiający wszystkie moduły.

1.2 MVC

1.2.1 Moduł wejść

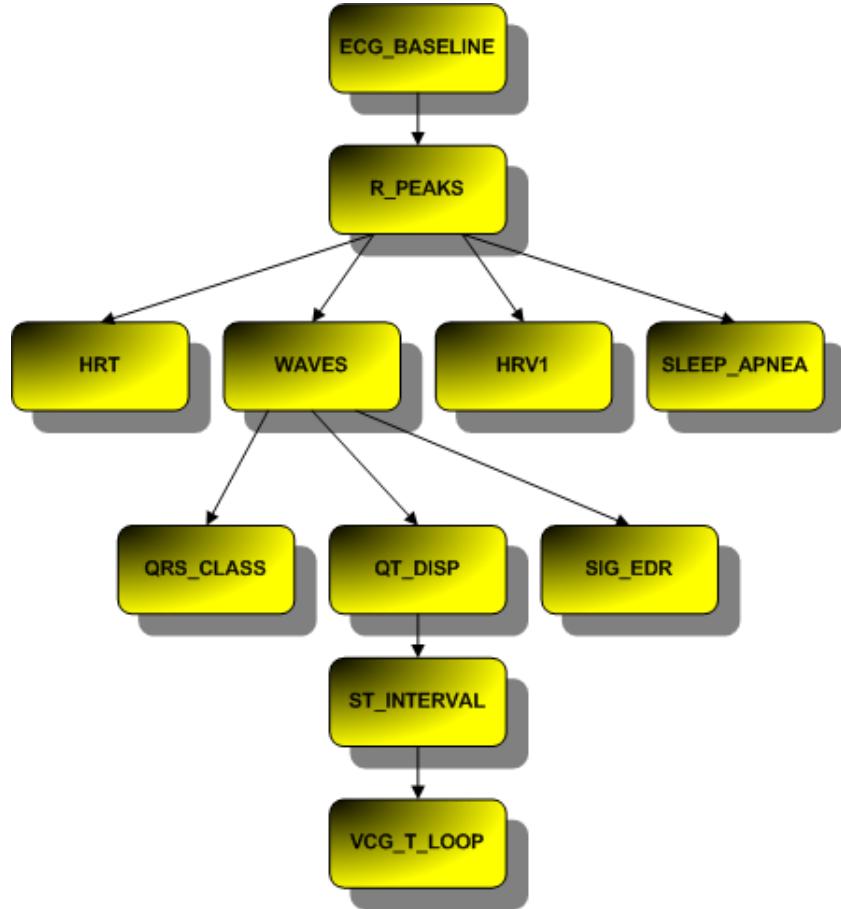
W projekcie wykorzystane zostały zserializowane dane w formacie binarnym „*.dat”. To oznacza, że program obsługuje tylko pliki z dołączonego do projektu folderu SampleData. W wypadku, kiedy użytkownik chciałby skorzystać z innych danych istnieją dwa rozwiązania:

- Można wykorzystać skrypt dołączony do projektu (1.2.4). Aby go uruchomić konieczne jest wcześniejsze pobranie z bazy Physionet oprogramowania ATM. Ta opcja jest dostępna dla użytkowników, którzy chcą pobrać dane zawierające dwa sygnały.
- Można pobrać oprogramowanie ATM , pobrać odpowiednie pliki i skonwertować je do formatu *.txt . Ta opcja odpowiednia jest dla plików zawierających sygnały z 12 elektrod i wykorzystywana jest tylko w module VCG_T_LOOP .

Taki sposób wczytywania danych jest optymalny pod względem szybkości działania programu, zdecydowanie utrudnia jednak korzystanie z innego rodzaju danych niż dostarczone.

1.2.2 Struktura aplikacji.

Istotnym z punktu widzenia rozwoju aplikacji zagadnieniem są współzależności modułów, czyli określenie kolejności wywoływania modułów. Ogólny schemat przedstawiony jest poniżej (??). Moduły wywoływane są rekurencyjnie w takim sensie, że w przypadku braku potrzebnych danych wywołany zostanie moduł wyżej. W efekcie każdy moduł można uruchomić jednym kliknięciem. W przypadku, w którym odpowiednie dane zostały już przetworzone, wywoływany moduł nie zażąda wywołania modułu nadrzednego. To oznacza, że po każdej zmianie parametrów wszystkie kolejne moduły należy wywoływać ręcznie, w odpowiedniej kolejności.



Rysunek 1.3: Schemat zależności modułów aplikacji

Poza modułami w projekcie występują następujące klasy:

- „appcontroller” czyli klasa zarządzająca całym programem.
- „airecgmain”, wraz z klasami pomocniczymi, czyli klasa obsługująca GUI.
- „ecgdata”, wraz z klasami pomocniczymi, czyli klasa przechowująca dane.
- „ecgentry”, wraz z klasami pomocniczymi, czyli klasa wczytująca dane binarne z pliku.

Całość projektu jest tworzona według wzorca Model-View-Controller. Oznacza to, że te trzy części są od siebie wyraźnie rozdzielone. Zarówno widoki (prezentacja wyników), jak i kontroler (interakcja z użytkownikiem) realizowana jest w całości przez GUI. Model (wszystkie obliczenia) wykonywane są w odpowiednich modułach, które nie muszą być dołączane do projektu. Zarządzanie tymi elementami realizowane (czyli silnik aplikacji) realizowane jest przez klasę „appcontroller”. Spełnienie ogólnego wzorca MVC w praktyce oznacza, że klasa appcontroller jest jedyną klasą, która może wymagać modyfikacji po zmodyfikowaniu któregokolwiek innego elementu programu.

1.2.3 Logi

Poza normalnym działaniem aplikacja oferuje również logowanie przebiegu ostatniego wywołania aplikacji z użyciem pakietu QsLog. Do logu zapisywane są najważniejsze zdarzenia i

błędy. Raportowany jest także czas wykonania wpisu, co pozwala na oszacowanie szybkości działania poszczególnych modułów, w zależności od wybranych w GUI parametrów. Przykładowy log przedstawiony jest poniżej.

```
INFO 2014-01-29T22:23:34.965 Program started
INFO 2014-01-29T22:24:01.591 Stworzono obiekt typu ecgdata dla pacjenta "107" .
INFO 2014-01-29T22:24:01.756 "649999" Samples loaded
INFO 2014-01-29T22:24:01.841 Reset procedure started:
INFO 2014-01-29T22:24:01.841 MVC/ Sleep Apnea deleted
INFO 2014-01-29T22:24:01.841 All removed.
INFO 2014-01-29T22:24:07.936 Start AtrialFibr
INFO 2014-01-29T22:24:07.936 Waves started.
INFO 2014-01-29T22:24:07.936 RPeaks stared.
INFO 2014-01-29T22:24:07.937 Ecg baseline started.
INFO 2014-01-29T22:24:07.937 BASELINE/ Using butterworth filter with coefficiets "Baseline wander"
INFO 2014-01-29T22:24:07.937 MVC/ "V1" signal will be processed.
INFO 2014-01-29T22:24:08.477 Ecg baseline done.
INFO 2014-01-29T22:24:08.542 RPeaks/ using Hilbert
INFO 2014-01-29T22:24:23.031 RPeaks done.
INFO 2014-01-29T22:24:37.838 Waves/ calculated "2114" QRS_onset points.
INFO 2014-01-29T22:24:37.838 Waves/ calculated "2114" QRS_end points.
INFO 2014-01-29T22:24:37.838 Waves/ calculated "1029" PWaveStart points.
INFO 2014-01-29T22:24:37.839 Waves/ calculated "908" PWaveEnd points.
INFO 2014-01-29T22:24:37.902 GUI/ ecgFrames.Count... "908"
INFO 2014-01-29T22:24:37.927 Waves done.
TRACE 2014-01-29T22:24:37.931 Atrial_FIBR/ calculated parameters:
Atrial_FIBR/ PWaveOccurrenceRatio: "0.208941"
Atrial_FIBR/ RRIntDivergence: "0.0319016"
Atrial_FIBR/ RRIntEntropy: "0.219118"
INFO 2014-01-29T22:24:37.931 AtrialFibr done
```

1.2.4 Skrypt przetwarzający dane.

```
REM written by mickl , AGH UST
CLS
@ECHO OFF
ECHO Hello ,
REM ECHO PhysioNet availabl indexes : 100 -124 , 200-234
REM SET /p index=Please insert record index
SET /a index=100
: loop
IF %index%==125 GOTO assign
IF %index%==235 GOTO quit

ECHO Downloading samples , record #%" + index%
rdsamp -r mitdb/%index% > %index%.a.txt
ECHO Succesfully saved %index%.a.txt
ECHO Serializing data
serializer.exe %index%.a.txt %index%.dat
DEL %index%.a.txt
ECHO Removing 101a.txt
ECHO Downloading annotations, record #%" + index%
rdann -r mitdb/%index% -a atr > %index%.c.txt
ECHO Succesfully saved %index%.c.txt
ECHO Downloading notes , record #%" + index%
wfdbdesc mitdb/%index% >%index%.d.txt
ECHO Succesfully saved %index%.d.txt

SET /a index=%index%+1
GOTO loop

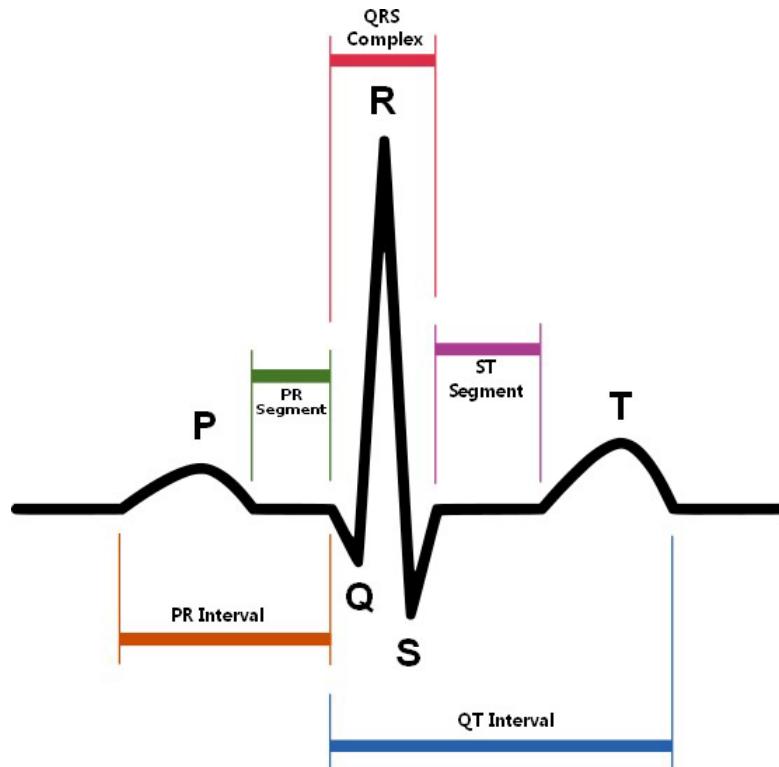
: assign
SET /a index=200
GOTO loop

: quit
ECHO All records converted
```

1.3 Moduł R_PEAKS

1.3.1 Badania literaturowe

Zadaniem modułu R_PEAKS jest detekcja załamków R w elektrokardiogramie. Jest to jeden z kluczowych etapów analizy sygnału EKG. Poprawne wykrycie i oznaczenie załamków R umożliwia bowiem wykrycie pozostałych punktów charakterystycznych cyklu pracy serca. Pozwala to dalej np. na przeprowadzenie dokładnej analizy zmienności rytmu serca. Załamki R są częścią tzw. zespołu QRS, który został przedstawiony na rysunku 1.4.



Rysunek 1.4: Fragment sygnału EKG z zaznaczonym zespołem QRS (QRS Complex)

Zespół QRS tworzą trzy sąsiadujące ze sobą punkty elektrokardiogramu, które składają się na wychylenie w kształcie wąskiego impulsu. Załamek Q to pierwsze ujemne wychylenie elektrokardiogramu, załamek R to pierwsze dodatnie wychylenie elektrokardiogramu, a załamek S to ostatnie ujemne wychylenie elektrokardiogramu w zespole QRS.

Zespół QRS jest odzwierciedleniem elektrycznej aktywności serca podczas skurczu komorowego. Reprezentuje on pobudzenie, czyli depolaryzację komór serca. Zespół QRS charakteryzuje się najwyższym wychyleniem i najkrótszym czasem trwania, co sprawia, że posiada on szerokie widmo częstotliwościowe. Charakterystyczny kształt tego zespołu oraz czas jego pojawiania się dostarcza najwięcej informacji diagnostycznych o bieżącym stanie serca i dlatego też jest punktem wyjściowym do klasyfikacji schematu cyklu sercowego. Służy też jako podstawa do automatycznego określania rytmu. Wykrycie zespołu QRS dostarcza podstawowych zasad i informacji dla prawie wszystkich automatycznych algorytmów analizy EKG. Czas trwania oraz wysokość zespołu QRS jest zależny od wielu czynników. Przyjmuje się jednak, że dla do-

rosłego i zdrowego człowieka średni czas trwania zespołu powinien wynosić od 0,06 s do 0,10 s. Wysokość sygnału zaś powinna mieścić się między 0,7 mV a 1,8 mV.

Detekcja załamków R wciąż jest przedmiotem wielu badań oraz licznych prac naukowych. Nie opracowano do tej pory metody, która gwarantowałaby stuprocentową skuteczność działania. Istnieją jednak algorytmy, które zapewniają ponad dziewięćdziesięciodziewięcio procentową skuteczność. Najpowszechniej stosowanym algorytmem jest algorytm Pan-Tompkins. Bazuje on na analizie odpowiednio przefiltrowanego oraz uśrednionego sygnału EKG. Algorytm ten łatwo adaptuje się do dynamiki sygnału. Dzięki temu jest bardzo skuteczny nawet w przypadku nieregularnej pracy serca. Ponadto metoda ta bardzo dobrze spisuje się w systemach czasu rzeczywistego. Wśród pozostałych algorytmów dużą popularnością cieszą się algorytmy oparte na przekształceniach Hilberta oraz transformacji falkowej. Przedmiotem obecnych badań są też metody, które łączyłyby te dwa algorytmy. Pozostałymi algorytmami detekcji załamków R są między innymi metody oparte na sieciach neuronowych, algorytmach genetycznych czy procesach stochastycznych.

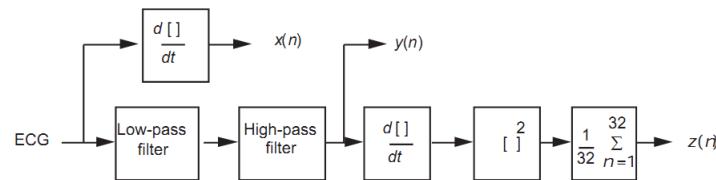
1.3.2 Koncepcja proponowanego rozwiązania

W trakcie realizacji modułu R_PEAKS zdecydowano się na implementację trzech algorytmów detekcji załamków R:

- algorytm Pan - Tompkins,
- algorytm bazujący na przekształceniach Hilberta,
- algorytm bazujący na transformacji falkowej.

1.3.2.1 Algorytm Pan - Tompkins

Algorytm ten można podzielić na dwa etapy. Pierwszym jest odpowiednie przygotowanie sygnału poprzez przefiltrowanie go oraz późniejsze jego uśrednienie. Drugim etapem jest znajdowanie załamków R we wcześniej przygotowanym sygnale. Na rysunku 1.5 przedstawiono schemat blokowy pierwszego etapu algorytmu.



Rysunek 1.5: Kolejne etapy algorytmu Pan - Tompkins przygotowujące sygnał do detekcji załamków R

I. Przepuszczenie sygnału przez filtry dolno i górnoprzepustowy.

II. Różniczkowanie

Celem różniczkowania sygnału jest wyłumienie komponentów niskoczęstotliwościowych oraz wzmacnianie komponentów wysokoczęstotliwościowych charakterystycznych dla

zespołów QRS. W tej fazie skorzystano z różniczkowania pięciopunktowego opisanego wzorem:

$$y(nT) = \frac{1}{8}[-x(nT - 2T) - 2x(nT - T) + 2x(nT + T) + x(nT + 2T)] \quad (1.1)$$

III. Potęgowanie

Potęgowanie sygnału ma na celu jeszcze większe wzmacnienie próbek sygnału odpowiadających zespołom QRS oraz wyłumienie pozostałych próbek sygnału. Ponadto potęgowania zapewnia nieujemność sygnału.

$$y(nT) = [x(nT)]^2 \quad (1.2)$$

IV. Całkowanie

Całkowanie sygnału w ruchomym oknie ma na celu jego wygładzenie oraz uzyskanie pojedynczej "fali" w obrębie zespołu QRS. Całkowanie sygnału wyraża się następującym wzorem:

$$y(nT) = \frac{1}{N} \sum_{i=nT-NT}^{nT} x(i) \quad (1.3)$$

Gdzie N to szerokość okna, którą należy odpowiednio dobrać w zależności od częstotliwości próbkowania. Należy mieć tu na uwadze fakt, że zbyt duża wielkość okna spowoduje zlanie się dwóch fal, a zbyt mała z kolei skutkować będzie utworzeniem zbyt dużej liczby fal.

V. Progowanie i detekcja załamków R

W trakcie pracy algorytmu odnajdywane są kolejne maksima lokalne, które mogą być oznaczone jako załamek R lub jako szum. Wykorzystywane są przy tym dwa proggi: THRESHOLD1 i THRESHOLD2. Są one na bieżąco aktualizowane na podstawie estymaty amplitudy załamków R (SPK) i szumu (NPK). Dodatkowo wykorzystywana jest technika searchback z parametrem RRMISSLEDLIMIT, który wyznaczany jest na podstawie różnic między kolejnymi załamkami R. Poniżej przedstawiono reguły detekcji załamków R:

if załamek R wykryto w czasie maksymalnie 200 ms od ostatniego załamka R **AND**
amplituda nowego załamka jest mniejsza od poprzedniego wykrytego **then**
 odrzuć załamek

else if nowy załamek R wykryto w czasie ponad 200 ms, a mniej niż 360 ms od ostatniego załamka **AND** nachylenie sygnału przy nowym maksymum jest przynajmniej równe połowie nachylenia dla poprzedniego maksymum **then**

 dodaj załamek

else if wartość wykrytego maksymum jest większa niż THRESHOLD1 **then**
 dodaj załamek

else if załamek R nie został wykryty przez okres RRMISSLEDLIMIT **AND** w ciągu tego czasu zostały znalezione maksima większe niż THRESHOLD2 **then**
 dodaj jako załamek największe znalezione maksimum

end if

W takcie działania algorytmu konieczna jest aktualizacja progów. Są one aktualizowane na podstawie wartości parametrów SPK i NPK, których początkowe wartości są następujące:

- SPK = średnia z ośmiu maksimów z pierwszych ośmiu sekund badania,
- NPK = 0.

Wartości tych parametrów są aktualizowane w następujący sposób:

```

if załamek R został wykryty za pomocą porównania z THRESHOLD1 then
     $SPK \leftarrow 0.125PEAK + 0.875SPK$ 
else if załamek R został wykryty za pomocą porównania z THRESHOLD2 then
     $SPK \leftarrow 0.250PEAK + 0.750SPK$ 
else
     $NPK \leftarrow 0.125PEAK + 0.875NPK$ 
end if
```

Progi natomiast są aktualizowane w następujący sposób:

$THRESHOLD1 = NPK + 0.25 (SPK - NPK)$

$THRESHOLD2 = 0.5 THRESHOLD1$

Interwały pomiędzy kolejnymi załamkami R obliczane są jako różnica wystąpienia ich czasów. Jeżeli interwał jest większy niż 0.92 RRAVERAGE2 i mniejszy niż 1.12 RRAVE-RAGE2, bufor interwałów jest aktualizowany o ostatnią wartość interwału i obliczana jest nowa wartość RRAVERAGE2. Wartość RRMISSDLIMIT używana w technice searchback jest równa 1.66 RRAVERAGE2 i stanowi czas, w którym poszukiwane są zaginione maksima.

1.3.2.2 Algorytm bazujący na przekształceniach Hilberta

Przekształcenie Hilberta to przekształcenie liniowe. Funkcji $x(t)$ przypisywana jest funkcja $x_h(t)$ postaci:

$$x_H = \mathcal{H}\{x(t)\} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\lambda)d\lambda}{t - \lambda} \quad (1.4)$$

W powyższym wzorze występuje dość trudna do wyliczenia niewłaściwa całka Riemann'a. W takiej sytuacji należy skorzystać z transformaty Fouriera. Wzór na przekształcenie Hilberta przyjmuje wtedy postać:

$$x_H(t) = \text{Im}\{\mathcal{F}^{-1}\{\mathcal{F}\{x(t)\}(\omega) \cdot (-j)\text{sgn}(\omega)\}\} \quad (1.5)$$

W pierwszym etapie algorytmu dokonujemy transformacji Hilberta całego sygnału EKG. Następnie na podstawie sygnałów $x(t)$ oraz $x_h(t)$ wyznacza się tzw. obwiednię sygnału postaci:

$$\mathcal{B}(t) = \sqrt{x(t)^2 + x_H(t)^2} \quad (1.6)$$

Po wyznaczeniu obwiedni sygnału należy przystąpić do etapu progowania.

I. Wyznaczenie początkowej wartości progu

Pierwsza wartość progu powinna być nieco mniejsza niż maksymalna wartość sygnału $B(t)$. Najczęściej spotykana wartość progu to ok. 0,9 wartości maksymalnej sygnału $B(t)$.

II. Wyznaczenie wartości o jaką należy zmniejszać próg w kolejnych iteracjach

Jeżeli za M1 przyjęto maksymalną wartość sygnału $B(t)$, a za M2 wartość minimalną tego sygnału to wartość o jaką należy zmniejszać próg w każdej iteracji wyraża się następującym wzorem:

$$fi = 0,01 \cdot M1 \cdot x \cdot \text{mantysa}(M1 - M2) \quad (1.7)$$

III. Wyznaczenie załamków

W tym kroku należy wyliczyć ile próbek sygnału $B(t)$ jest większych od obecnego progu. Jeżeli liczba próbek powyżej progu jest większa od ilości próbek w poprzedniej iteracji to należy zmniejszyć próg o wartość δ i ponownie policzyć próbki powyżej progu. Jeżeli ilość próbek powyżej progu się nie zmieniła to należy przerwać etap progowania.

Ostatnim etapem jest detekcja załamków R . Wyznaczone w etapie progowania próbki powyżej progu będą występować w grupach. W każdej z tych grup znajdująć się będzie próbka o największej amplitudzie, która będzie szukanym załamkiem R .

1.3.2.3 Algorytm bazujący na transformacji falkowej

Falki (z ang. wavelet) – rodziny funkcji zbioru liczb rzeczywistych w zbiór liczb rzeczywistych, z których każda jest wyniesiona z funkcji-matki (z tzw. funkcji macierzystej) za pomocą przesunięcia i skalowania:

$$\psi_{j,k}(t) = \psi(2^j + k) \quad (1.8)$$

gdzie: j, k – liczby całkowite, ψ – funkcja - matka, $\psi_{j,k}$ – falka o skali j i przesunięciu k (zwana też funkcją falkową). Funkcje te dążą do zera lub po prostu wynoszą zero dla argumentów dążących do nieskończoności. Ponadto funkcje te mają energię skupioną w jednym miejscu tzn.

$$\int_{-\infty}^{\infty} \psi(t) dt = 0 \quad (1.9)$$

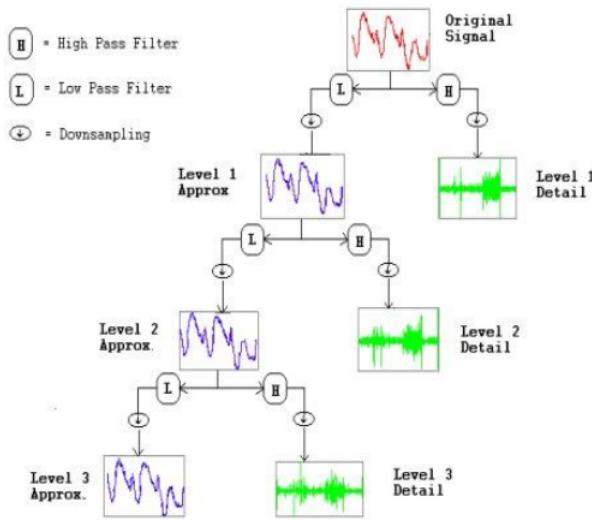
Transformacja falkowa dla sygnałów ciągłych jest przekształceniem całkowym:

$$\tilde{s}_{\Psi} = \frac{1}{\sqrt{a}} \int_{-\infty}^{\infty} s(t) \Psi\left(\frac{t-b}{a}\right) dt \quad (1.10)$$

gdzie:

- a - parametr skali
- b - parametr przesunięcia
- $s(t)$ - sygnał badany zależny od czasu t
- Ψ - funkcja falkowa
- \tilde{s}_{Ψ} - współczynnik falkowy zależny od parametrów a i b
- $\Psi\left(\frac{t-b}{a}\right)$ - jądro przekształcenia

Zastosowanie transformacji falkowej pozwala rozłożyć sygnał na kilka subpasm (rysunek 1.6). Następnie dobiera się odpowiednie subpasma i rekonstruuje sygnał. Tak uzyskany sygnał poddaje się takiej samej procedurze progowania i detekcji załamków R jak w przypadku wcześniej opisowanego algorytmu wykorzystującego przekształcenia Hilberta.



Rysunek 1.6: Dekompozycja sygnału na subpasma za pomocą transformacji falkowej

Działanie algorytmu można przedstawić w następujących krokach:

- I. Dwukrotna dekompozycja oryginalnego sygnału z wykorzystaniem transformacji falkowej.
- II. Wyszukanie próbki o największej wartości w zdekomponowanym sygnale.
- III. Znalezienie wszystkich próbek o wartości nie mniejszej niż 30% wartości próbki znalezionej w poprzednim kroku.
- IV. Filtracja próbek, które zostały wykryte w poprzednim kroku w celu odrzucenia tych, które należą do tego samego załamka R.
- V. Znalezienie indeksów wyszukanych próbek w sygnale po transformacji i przemnożenie ich razy 4 w celu znalezienia w przybliżeniu indeksów próbek w sygnale oryginalnym, które zostały sklasyfikowane jako załamki R.
- VI. Wyszukanie faktycznego indeksu próbki w oryginalnym sygnale poprzez przeszukanie odpowiedniego przedziału oryginalnego sygnału wokół próbki o indeksie znalezionym w poprzednim kroku.

1.3.3 Rezultaty i wnioski

Zaimplementowane metody w większości przypadków działają dobrze i poprawnie wykrywają załamki R. Ewentualny błąd wynosi co najwyżej kilka próbek. W przypadku sygnałów dosyć zniekształconych zdarzają się sytuacje, że zostały wykryte nadmiarowe załamki R, bądź niektóre z nich nie zostały w ogóle wykryte.

Należy jednak nadmienić, iż metody detekcji załamków R, Pan - Tompkins oraz bazująca na transformacie Hilberta, które zostały zaimplementowane w roku poprzednim przez kolegów Pawła Kaspereczyka i Kamila Pękałę działają poprawnie. Z tego powodu zostały również wykorzystane w tegorocznym projekcie. Zostały wprowadzone jednak drobne modyfikacje związane z dopasowaniem tych metod pod API tegorocznego projektu, jak również pewne modyfi-

kacje w sposobie zwracania rezultatów obliczeń funkcji składowych klasy reprezentującej moduł oraz pewne zmiany kosmetyczne związane z formatowaniem kodu. Konsekwencją wynikającą z wykorzystania zaimplementowanych już metod jest to, że są pewne różnice pomiędzy proponowanym w wcześniej rozwiązaniem a rozwiązaniem zastosowanym w programie.

W zaimplementowanej metodzie bazującej na transformacji Hilberta został zastosowany podział sygnału na przedziały. Transformacja dokonywana jest zatem na fragmencie sygnału, po czym otrzymane rezultaty sklejane są w całość. Zgodnie z informacjami zawartymi w raporcie z zeszłorocznego projektu, daje to lepsze rezultaty, niż obliczenia dokonywane na całym sygnale bez podziału na przedziały. Kolejną różnicą jest to, że nie jest obliczany pierwiastek kwadratowy, który występuje we wzorze na obwiednią sygnału. Upraszczają to obliczenia. Podyktowane jest to faktem, iż liczba rzeczywista o większej wartości ma również większy pierwiastek kwadratowy. Również sama metoda detekcji załamków R jest nieco inna niż proponowana w jednym z wcześniejszych rozdziałów. Dokonywana ona jest na podstawie przeglądu fragmentu tablicy *diff* zawierającej posortowane malejąco elementy, które są większe niż $\frac{1}{21}$ maksymalnej wartości z tej. Elementy te są w składzie transformaty Hilberta w obwiednię sygnału, obliczone na podstawie wspomnianego wyżej, uproszczonego wzoru niezawierającego pierwiastka kwadratowego. Próbka sygnału wejściowego, która odpowiada aktualnie przeglądaniu elementowi tablicy, jest odpowiednio daleko od poprzednio zakwalifikowanych jako załamek R próbek, jest również traktowana jako szukany załamek. W przeciwnym przypadku jest odrzucana. Minimalna odległość między załamkami jest oszacowana na podstawie założenia, iż maksymalna częstotliwość pracy serca wynosi 220 uderzeń na minutę. Wyznaczona odległość pomnożona jest jeszcze przez współczynnik wynoszący 0,8. Szerokość fragmentu sygnału poddawanego transformacji Hilberta przyjęta została jako 200 sekund.

Bardzo ważnym elementem detekcji załamków R jest uzyskanie do analizy dobrze przefiltrowanego sygnału, pozbawionego dużych zakłóceń i szumów. Podczas testów poprawności działania modułu został przeprowadzony eksperyment polegający na detekcji załamków R w sygnale nie poddanym filtracji. Metoda Pan - Tompkins mimo wszystko dała w miarę dobre rezultaty, natomiast metoda bazująca na transformacji Hilberta kompletnie nie poradziła sobie z wykryciem pików. Prawdopodobnie główną przyczyną tego faktu jest obecność prostej i odwrotnej transformaty Fouriera w zaimplementowanej metodzie. Metoda oparta na transformacji falkowej, w zaimplementowanym w module algorytmie, wyszukuje załamki, których wartość przekracza pewien ustalony próg wyznaczony w oparciu o próbę o największej wartości. Z tego powodu obecność zakłóceń o zbyt dużej wartości może powodować zakwalifikowanie tych zakłóceń jako nadmiarowe załamki.

Z racji, iż stosowane algorytmy przeważnie wykrywają trochę mniejszą liczbę załamków niż wynosi ona w rzeczywistości, ciężko było dokonać analizy różnic pomiędzy wynikami zwróconym przez algorytm a wynikami referencyjnymi w sposób automatyczny, przy wykorzystaniu programu. Dla wybranych sygnałów EKG zostały narysowane wykresy z naniesionymi, referencyjnymi i wyznaczonymi przez algorytmy, załamkami R. Sygnał oryginalny był filtrowany przy pomocy filtru Butterwortha. Nie zostały przeprowadzone testy dla innych metod filtracji w oparciu o założenie, że każda z tych metod daje poprawne rezultaty i wyniki detekcji załamków R powinny być podobne dla każdej z metod filtrowania sygnału.

1.3.3.1 Analiza rezultatów działania modułu oraz porównanie algorytmów detekcji załamków R na podstawie wybranych sygnałów

- Sygnał nr 105 Sygnał o numerze 105 został wybrany do analizy efektów działania modułu ze względu na jego ciekawy kształt w drugiej połowie.

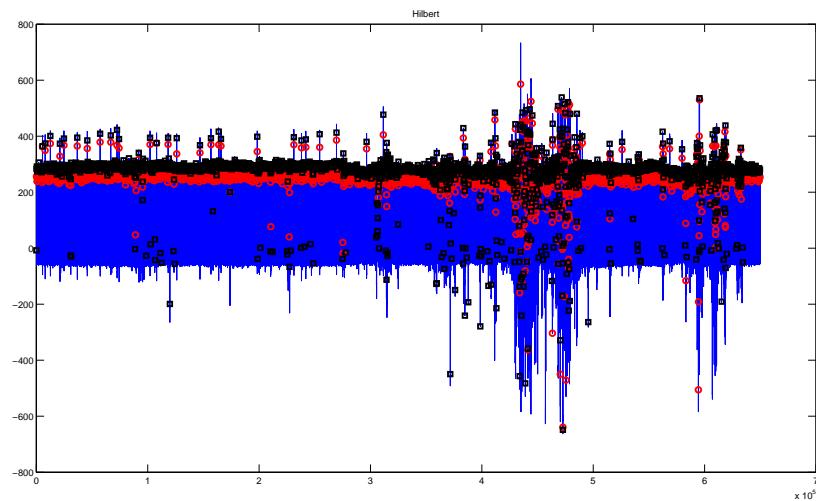
Tablica 1.1: Załamki R dla sygnału 105

Sygnał nr 105	
Referencyjne	2691
Hilbert	2607
Pan - Tompkins	2554
Transformata falkowa	2587

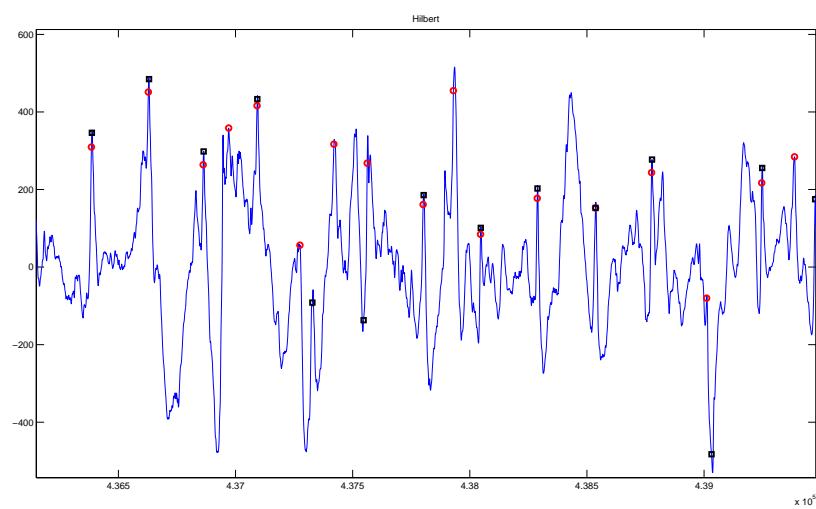
I. Algorytm bazujący na transformacji Hilberta

Na wykresie 1.7 zaznaczone zostały czerwonymi kółkami wykryte przez algorytm załamki R, natomiast kwadratami koloru czarnego naniesione są piki referencyjne. Taki sposób oznaczania załamków jest przyjęty dla pozostałych wykresów.

Można odnieść wrażenie, że różnice między załamkami referencyjnymi i tymi znalezionymi przez algorytm są dosyć duże. Jednak po dokonaniu szybkiej analizy wykresu 1.8 nasuwa się wniosek, że różnica nie jest duża, rzędu kilku próbek, a wrażenie dużych różnic spowodowane jest stromym nachyleniem załamka. Można również zauważać, że wykryte zostały pewne nadmiarowe piki. W związku z tym, należy się spodziewać, że wiele pików zostało również pominiętych. Wynika to z faktu, że sumaryczna ilość załamków wykrytych przez algorytm jest ciut mniejsza niż ilość załamków referencyjnych. Można również stwierdzić, iż załamki referencyjne nie są również idealne. Może to być podyktowane ewentualnymi opóźnieniami powstałymi np. w przypadku filtracji sygnału.



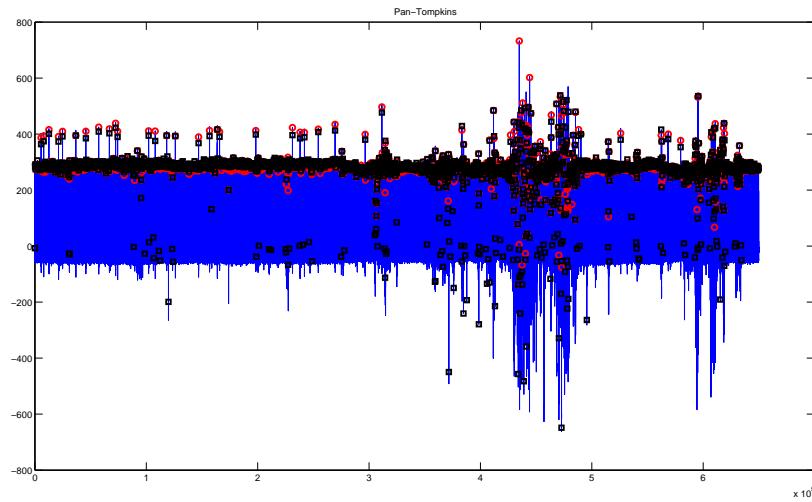
Rysunek 1.7: Sygnał nr 105 z naniesionymi pikami R - Hilbert



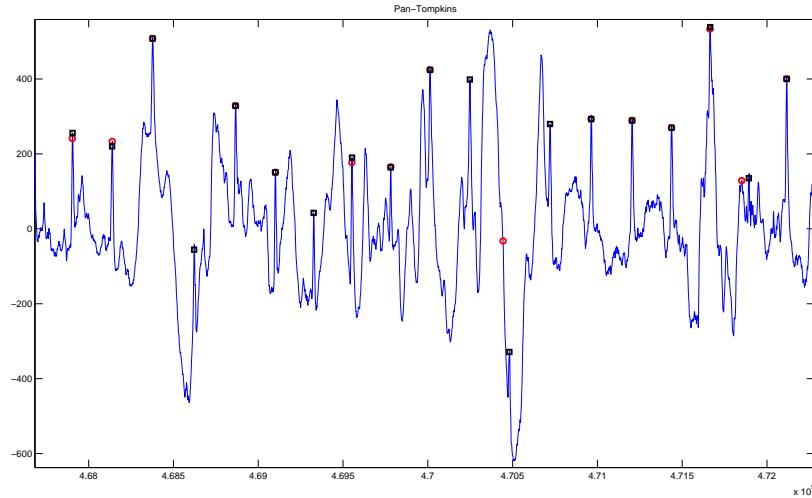
Rysunek 1.8: Fragment sygnału nr 105 - Hilbert

II. Algorytm Pan - Tompkins

Dla sygnału o nr. 105, wynik działania algorytmu Pan - Tompkins wydają się być lepsze niż dla algorytmu, który bazuje na przekształceniu Hilberta. Wprawdzie można zauważać na rysunku 1.9, że jest całkiem sporo załamków niewykrytych, lecz te, które zostały znalezione, różnią się niewiele od tych referencyjnych. Analizując fragment sygnału widać, że niejednokrotnie załamki znalezione na drodze działania algorytmu pokrywają się z referencyjnymi, lub różnią się od nich bardzo mało. Podobnie jak w przypadku poprzedniej metody, ilość wykrytych załamków jest nieco mniejsza niż ich faktyczna liczba.



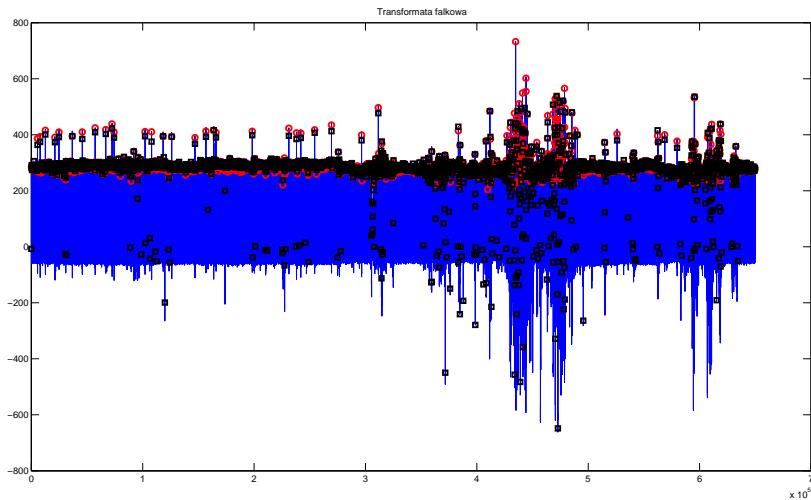
Rysunek 1.9: Sygnał nr 105 z naniesionymi pikami R - Pan - Tompkins



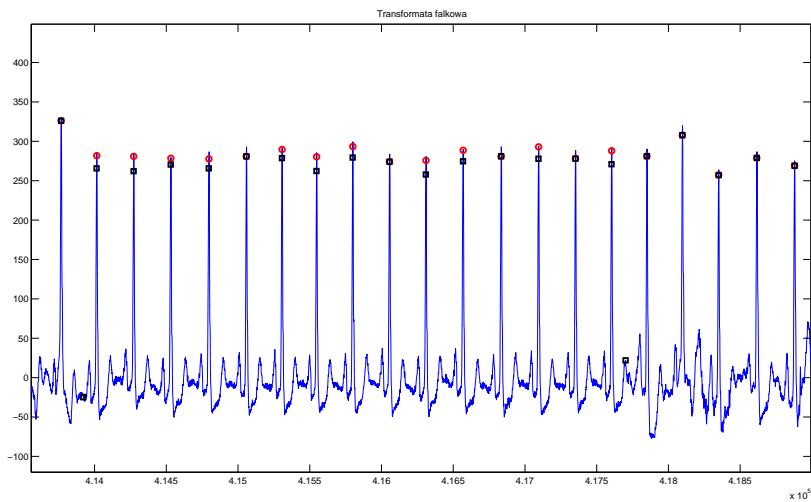
Rysunek 1.10: Fragment sygnału nr 105 - Pan - Tompkins

III. Algorytm bazujący na transformacji falkowej

Specyfika działania tego algorytmu powoduje, że za przyczyną widocznego na rysunku 1.11, mniej więcej w dwóch trzecich sygnału wysokiemu załamkowi, dosyć dużo załamków R nie zostało wykrytych. Natomiast w przypadku załamków, które zostały odnalezione, widoczna jest mała różnica w stosunku do załamków referencyjnych. Analiza fragmentu sygnału (rys. 1.12) pokazuje, że często piki wykryte przez algorytm są lepsze od referencyjnych, lub pokrywają się z nimi.



Rysunek 1.11: Sygnał nr 105 z naniesionymi pikami R - transformata falkowa



Rysunek 1.12: Fragment sygnału nr 105 - transformata falkowa

- Sygnał nr 122

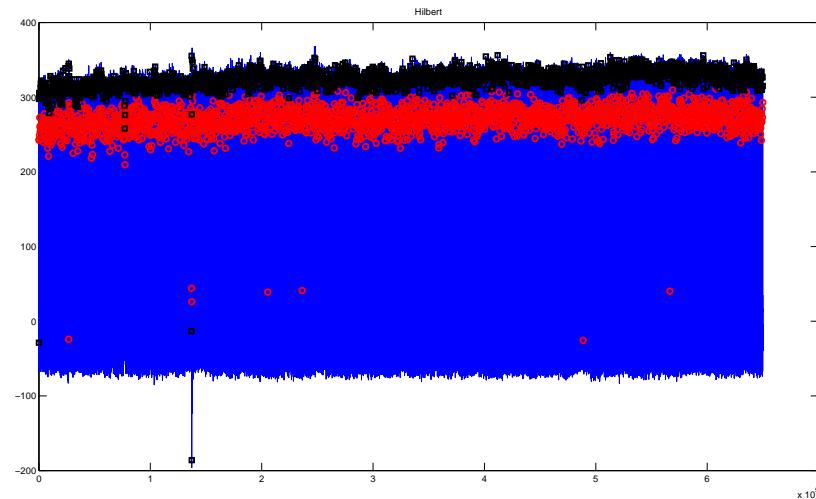
Sygnał nr 122 został wybrany ze względu na jego dosyć regularny kształt. Z tego powodu każda z metod bardzo dobrze pozwoliła na detekcję załamków R, co można odnotować czytając dane zawarte w poniższej tabeli.

Tablica 1.2: Załamki R dla sygnału 122

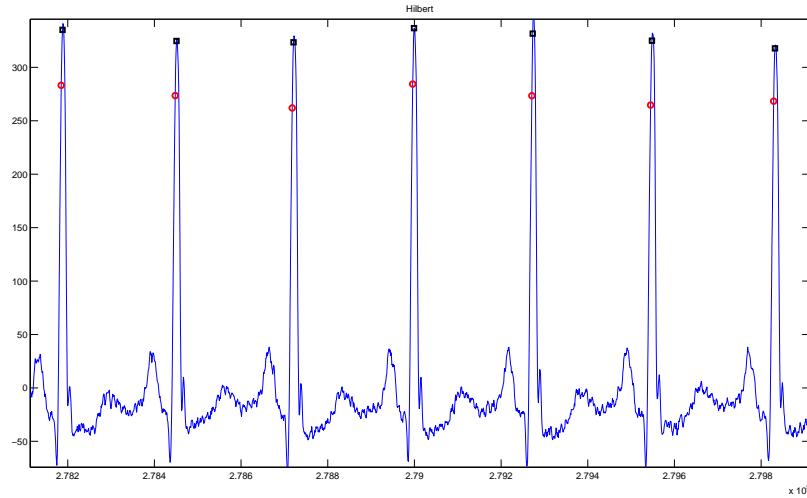
Sygnał nr 122	
Referencyjne	2479
Hilbert	2476
Pan - Tompkins	2475
Transformata falkowa	2476

I. Algorytm bazujący na transformacji Hilberta

Podobnie jak dla sygnału o nr. 105, analiza wykresu 1.13 pozornie może doprowadzić do wniosku, iż różnice między pikami R referencyjnymi oraz wykrytymi na podstawie działania algorytmu bazującego na przekształceniu Hilberta są dosyć znaczne. Wynika to jednak ponownie z faktu, że załamki R są strome i różnica kilku próbek powoduje duże różnice w wartości sygnału, jednak istotą algorytmu detekcji załamków R jest przede wszystkim dostarczenie indeksów próbek.



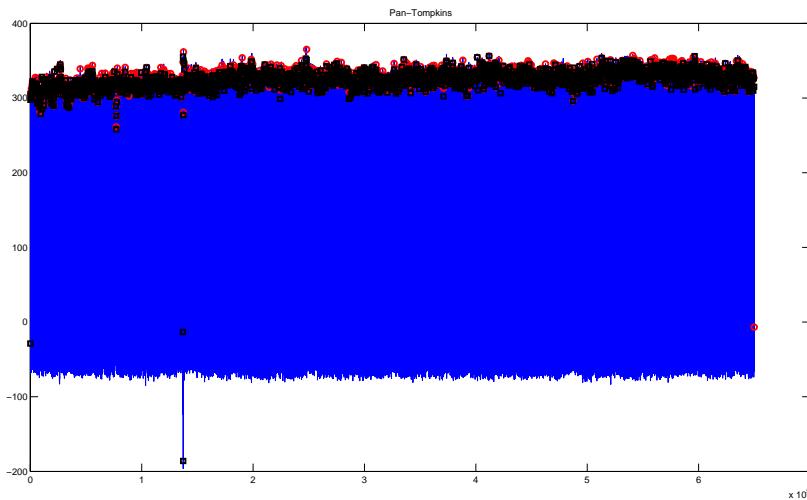
Rysunek 1.13: Sygnał nr 122 z naniesionymi pikami R - Hilbert



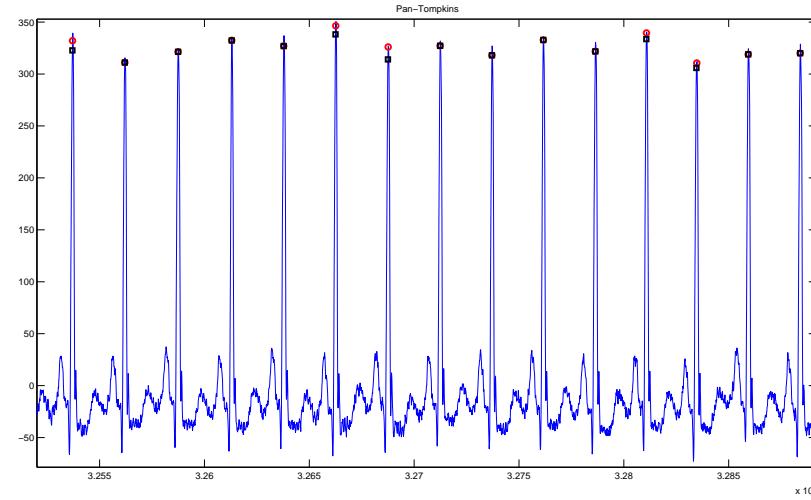
Rysunek 1.14: Fragment sygnału nr 122 - Hilbert

II. Algorytm Pan - Tompkins

Na diagramie 1.15 widać, że różnice pomiędzy załamkami wykrytymi przez algorytm Pan - Tompkins i referencyjnymi są bardzo małe. Analiza fragmentu sygnału (rys. 1.16) tylko to potwierdza. Widać nawet, że niekiedy piki R wykryte przez algorytm są lepsze.



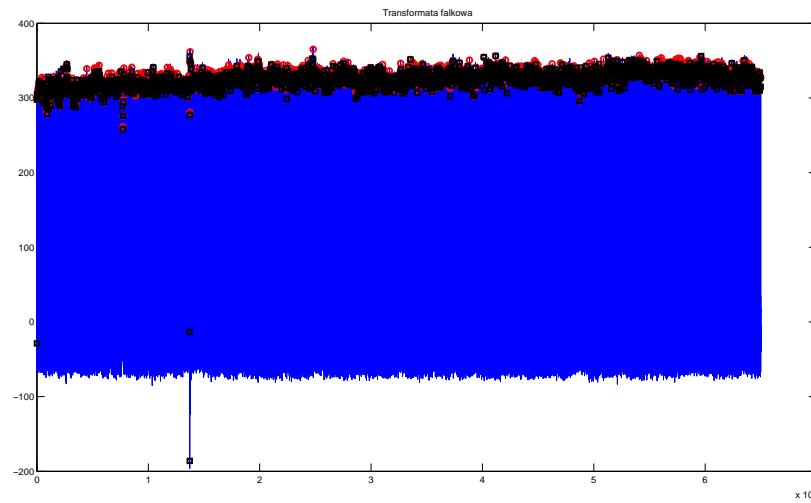
Rysunek 1.15: Sygnał nr 122 z naniesionymi pikami R - Pan - Tompkins



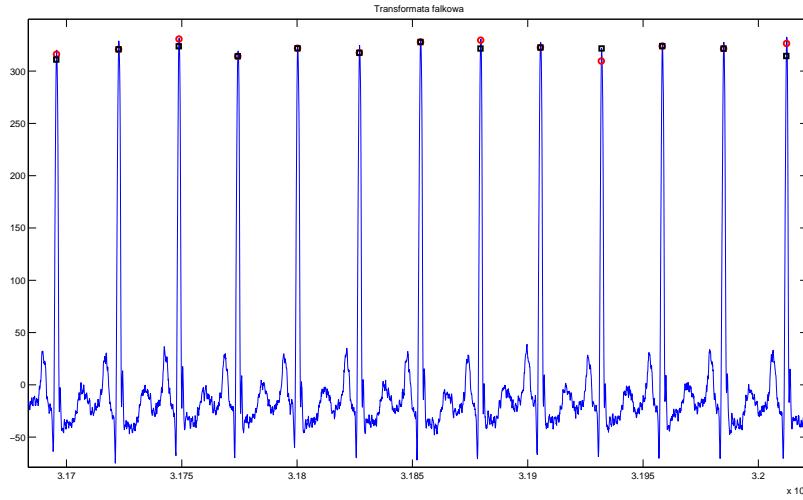
Rysunek 1.16: Fragment sygnału nr 122 - Pan - Tompkins

III. Algorytm bazujący na transformacji falkowej

Podobnie jak w przypadku algorytmu Pan - Tompkins, algorytm bazujący na transformacji falkowej dostarcza bardzo dobre rezultaty. Wszystkie załamki są o podobnej amplitudzie, więc algorytm nie wprowadza przekłamań. Na fragmencie sygnału (rys. 1.18) widać, że załamki wykryte przez algorytm pokrywają się z załamkami referencyjnymi, lub są nawet lepsze.



Rysunek 1.17: Sygnał nr 122 z naniesionymi pikami R - transformata falkowa



Rysunek 1.18: Fragment sygnału nr 122 - transformata falkowa

- Sygnał nr 228

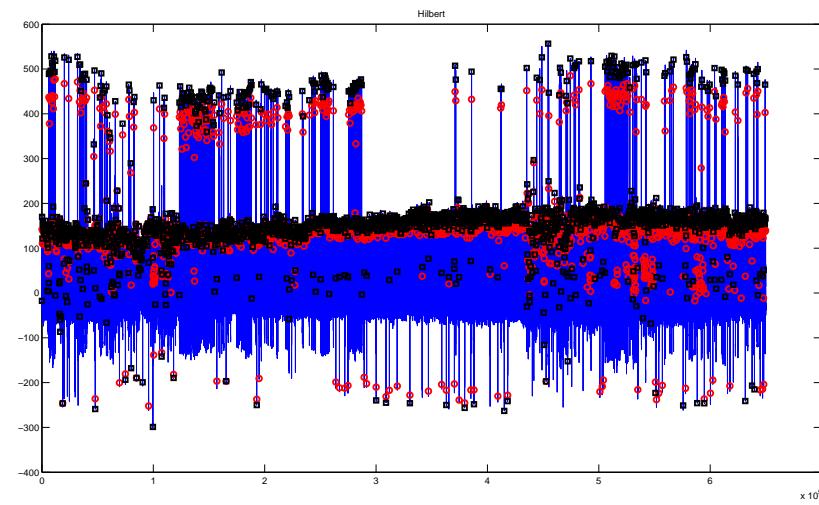
Sygnał nr 228 został wybrany ze względu na jego ciekawy, nieregularny kształt. Z tego powodu algorytm bazujący na transformacji falkowej kompletnie zawiódł - wykryto jedynie ok. 25% wszystkim załamków.

Tablica 1.3: Załamki R dla sygnału 228

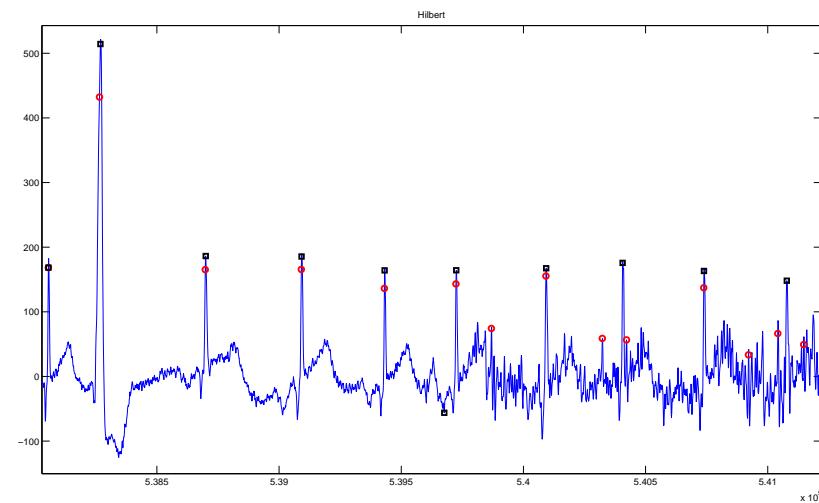
Sygnał nr 228	
Referencyjne	2141
Hilbert	2094
Pan - Tompkins	1890
Transformata falkowa	568

I. Algorytm bazujący na transformacji Hilberta

Analizując wykres całego przebiegu sygnału (rys. 1.19) można dojść do wniosku, że algorytm bazujący na transformacji Hilberta dosyć dobrze wywiązał się z zadania, aczkolwiek różnice są widoczne. Na fragmencie sygnału (rys. 1.20 widać, że w przypadku wykrytych w miarę poprawnie załamków, różnice są rzędu kilku próbek, natomiast widać, również piki, które nie zostały w ogóle wykryte, lub nawet pewne nadmiarowe załamki. Ostatni przypadek oznacza jednak, że w przekroju całego sygnału dużo załamków musiało być pominiętych - liczba wszystkich wykrytych załamków jest mniejsza od liczby pików referencyjnych.



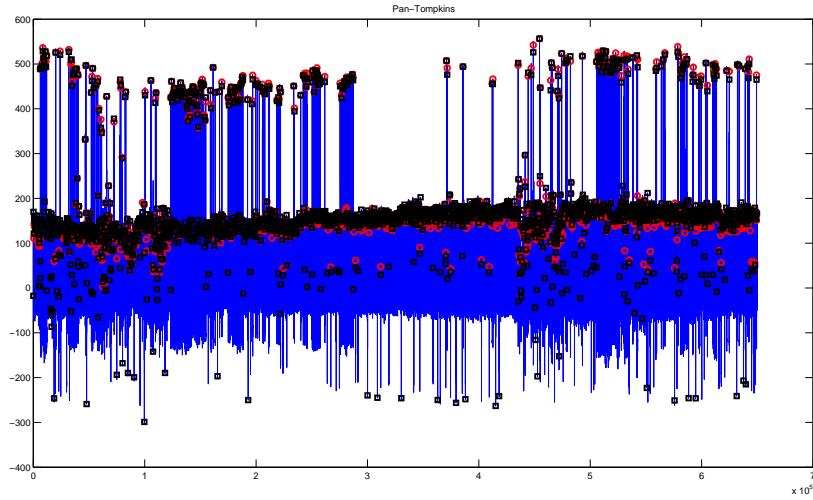
Rysunek 1.19: Sygnał nr 228 z naniesionymi pikami R - Hilbert



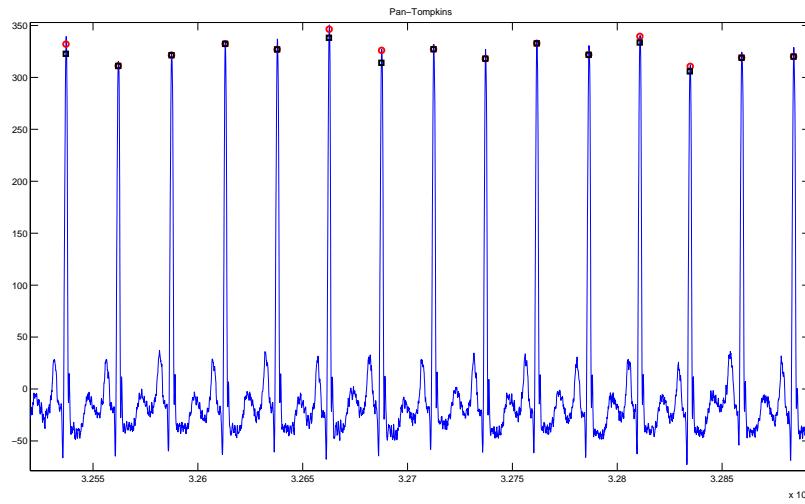
Rysunek 1.20: Fragment sygnału nr 228 - Hilbert

II. Algorytm Pan - Tompkins

Na wykresie 1.21 widać, że wykryto większość załamków i różnice między nimi i referencyjnymi są małe. Analiza fragmentu sygnału tylko to potwierdza - piki wykryte przez algorytm Pan - Tompkins pokrywają się z referencyjnymi, a nieraz nawet są lepsze.



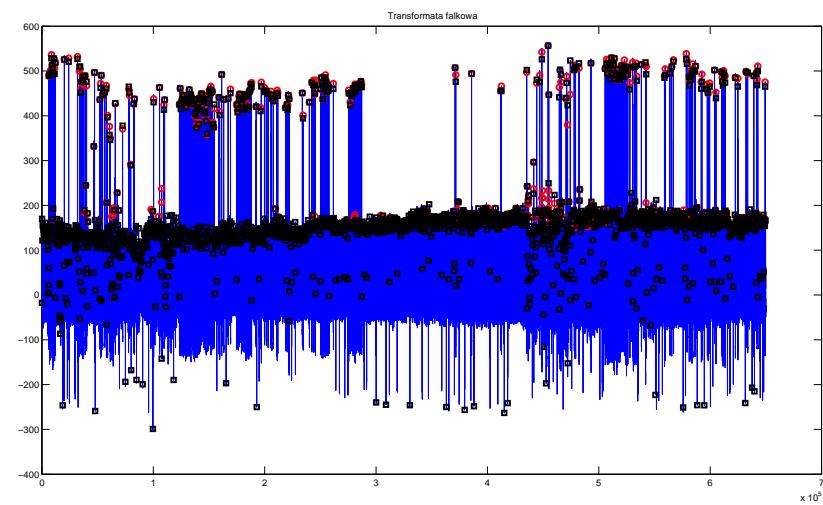
Rysunek 1.21: Sygnał nr 228 z naniesionymi pikami R - Pan - Tompkins



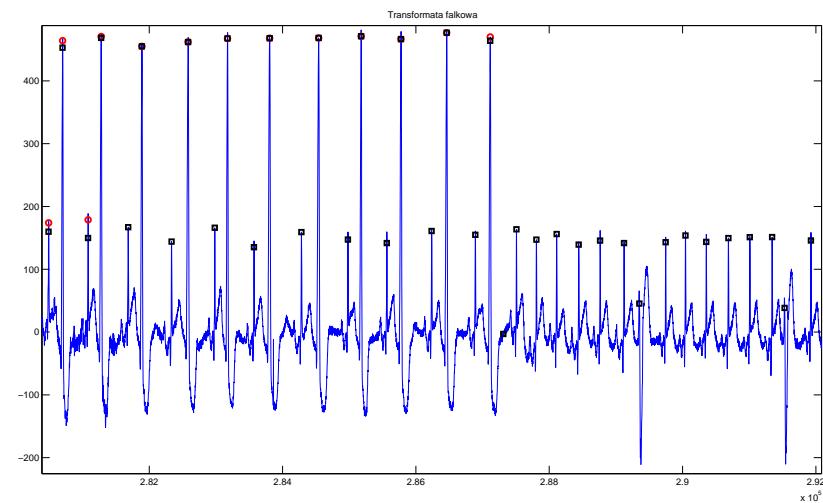
Rysunek 1.22: Fragment sygnału nr 228 - Pan - Tompkins

III. Algorytm bazujący na transformacji falkowej

Sygnał nr 228 jest dobrym przykładem sygnału, dla którego algorytm opierający swoje działanie na transformacie falkowej nie radzi sobie. Z powodu, iż różnice w amplitudzie sygnału dla niektórych załamków R są znaczne, algorytm może nie wykryć pików o małej wartości. Wprawdzie w przypadku wykrytych załamków rezultaty są dobre, to w kontekście całego sygnału wyniki są słabe. Dobre widoczne jest to na fragmencie sygnału (rys. 1.24). Widać bardzo dużo niskich załamków, które zostały pominięte przez algorytm. Piki o dużej amplitudzie zostały wykryte bardzo dobrze.



Rysunek 1.23: Sygnał nr 228 z naniesionymi pikami R - transformata falkowa



Rysunek 1.24: Fragment sygnału nr 228 - transformata falkowa

1.3.4 Diagramy klas

RpeaksModule
SAFETYCOEFFICIENT : static const double = 0.8
MAXBPM : static const unsigned short = 220
THRESHOLDDIVISOR : static const double = 20
PTLPM : static const unsigned int = 9
PTHPM: static const unsigned int = 56
PTMWN: static const unsigned int = 58
filteredSignal : QVector<double>
ecgSignal : const QVector<double> &
itVect : QVector<unsigned int>
frequency : double
fft(x, result)
ifft(x, result)
fftHilbertWindowed(x, result, windowlength, windowoverlap)
fftHilbert(x, result)
setHilbertRPeaks(peaks, dt)
PTLowPassFilter(data, reset): double
PTHHighPassFilter(data, reset): double
PTDerivative (data, reset): double
PTSquaringFunction data): double
PTMovingWindowIntegral(data, reset): double
Maximum(begin, end, values): PairDoubleUnsignedInt
PanTompkinsDo(signal, result)
PanTompkinsSetPeaks(filter, movingWindow)
PeakMS(index, signal): PairDoubleUnsignedInt
QRSCheck(rpCandidate, rpLast, threshold, slope): int
falka(x, result)
getEvenSamples(x, result)
getOddSamples(x, result)
findGreaterEqualThan(x, coeff, result)
RpeaksModule(filteredSignal, freq)
hilbert()
panTompkins()
wavelet()
getPeaksIter(): const RpeaksIterVector &
getPeaksIndex(): const QVector<unsigned int> &
ecgBegin(): const RpeaksIter

DiffSorter
DiffSorter()
operator()(a, b): bool

PeaksCutSorter
PeaksCutSorter(scope)
operator()(a, b): bool

1.3.5 Literatura

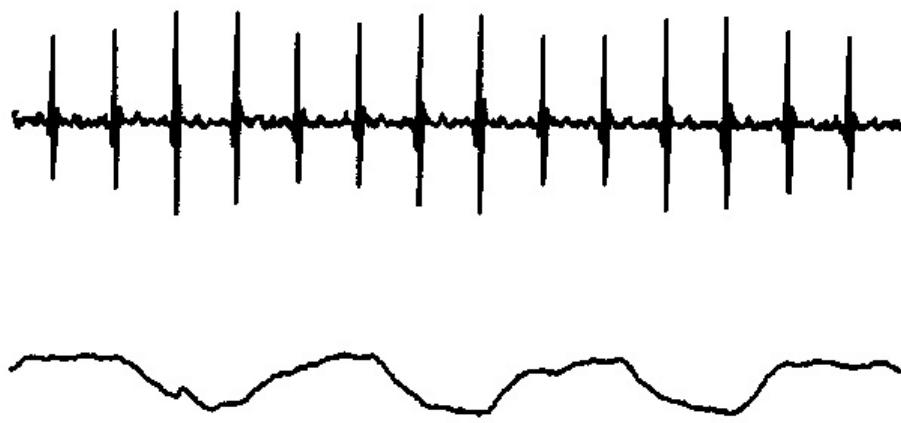
- [1] Farahabadi A., Farahabadi E., Parsa M. Mahjoob1, Rabbani Hossein. R Peak Detection in Electrocardiogram Signal Based on an Optimal Combination of Wavelet Transform, Hilbert Transform, and Adaptive Thresholding. *Journal of Medical Signals & Sensors* 2011, nr 2, s. 91-98.
- [2] Augustyniak Piotr. *Przetwarzanie sygnałów elektrodiagnostycznych*. Kraków, Uczelniane Wydawnictwo Naukowo-Dydaktyczne, 2001, ISBN 83-88400-37-2.
- [3] Pięciak Tomasz, *Elektrokardiografia (EKG)*.
- [4] Goszcz Sławomir, Kokoć Kamil. *Detekcja i synchronizacja reprezentacji uderzeń serca (zespół QRS)*.
- [5] Kasperczyk Paweł, Pękala Kamil. *Raport z modulu obliczeniowego R_PEAKS*. Kraków 2013.
- [6] Hennig Carsten, Köhler Bert-Uwe, Orglmeister Reinhold. The Principles of Software QRS Detection. *IEEE ENGINEERING IN MEDICINE AND BIOLOGY*, January/February 2002.
- [7] Bhujanga Rao A., Narayana K.V.L., Wavelet based QRS detection in ECG using MATLAB. *Innovative Systems Design and Engineering* 2011, nr 7, s. 60-69.
- [8] Simranjit Singh Kohli, Nikunj Makwana, Nishant Mishra, Balwalli Sagar. Hilbert Transform Based Adaptive ECG R-Peak Detection Technique. *International Journal of Electrical and Computer Engineering (IJECE)* 2012, nr 5, s. 639-643.

1.4 Moduł VCG_T_LOOP

1.5 Moduł SIG_EDR

1.5.1 Badania literaturowe

Moduł SIG_EDR jest odpowiedzialny za wyodrębnienie sygnału oddechu z elektrokardiogramu. Przykładowy przebieg EKG wraz z odpowiadającym mu sygnałem oddechowym zaprezentowany został na Rys. 1.25. Wiedza na temat sposobu oddychania pacjenta jest przydatna przy analizie długoterminowej w celu wykrycia np. bezdechów sennych. Otrzymana w ten sposób informacja nie wymaga użycia dodatkowego sprzętu, mogącego zakłócić naturalny rytm oddechowy.



Rysunek 1.25: EKG (przebieg górny) wraz z sygnałem oddechowym (przebieg dolny)

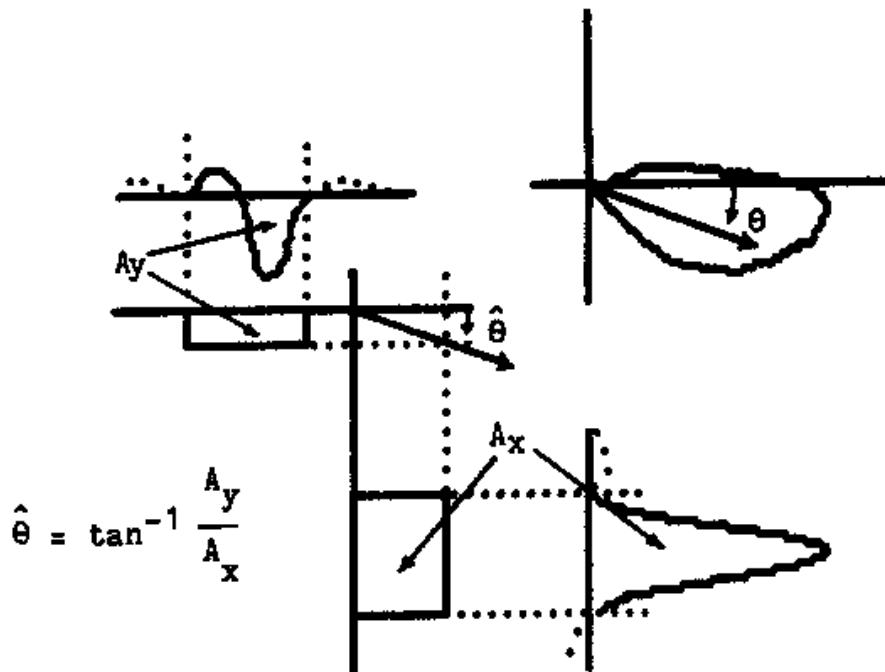
Sposób podejścia do problemu nie zmienił się znacznie w przeciagu lat. Istnieje kilka metod, które zyskały uznanie ze względu na swoją skuteczność tudzież prostotę obliczeniową i łatwość implementacji.

Pierwszą z nich jest metoda zaproponowana przez Felblinger et al. Polega ona na badaniu amplitudy pików R. Metoda ta wykorzystuje fakt, iż spośród wszystkich faz EKG piki R podlegają największej modulacji podczas oddychania (w wyniku zmiany impedancji klatki piersiowej). W momencie nabrania powietrza oraz trzymania go w klatce piersiowej jej impedancja rośnie, przez co amplituda pików R maleje, odwrotnie w momencie wydychania – amplituda rośnie. Przekłada się ona zatem bezpośrednio na sposób oddychania. Aby otrzymać ciągły sygnał oddechowy pacjenta należy wykonać interpolację pików R funkcjami sklejonymi (sygnał może zostać poprawnie odtworzony, gdyż prawie zawsze częstotliwość skurczów serca jest przynajmniej dwa razy większa od częstotliwości oddychania – zostaje spełnione twierdzenie o próbkowaniu). Otrzymany w ten sposób sygnał reprezentuje sposób oddychania pacjenta. Algorytm ten jest łatwy w implementacji oraz dosyć efektywny, wymaga jednak podania dobrze przefiltrowanego sygnału EKG na wejście (niezwykle istotna jest eliminacja falowania izolinii).

Kolejna metoda również opiera się na badaniu pików R. Tutaj jednak liczony jest czas trwania fali R. W tym celu w pierwszej kolejności liczona jest pochodna sygnału EKG w oknie czasowym o długości 40 ms (+/- 20 ms od piku R). Pochodna ta generuje jedno minimum oraz jedno maksimum lokalne (dla danego piku R). Wartości różnic czasowych pomiędzy minimum a maksimum umieszcza się na wykresie w funkcji czasu wystąpienia danego piku R. Następnie

otrzymane punkty interpoluje się funkcjami sklejonymi, otrzymując w ten sposób sygnał EDR. Obie przytoczone metody zostały porównane w pracy Murtaza M. Lakdawala „Derivation of the respiratory rate signal from a single lead ECG”. Została w niej również zaprezentowana metoda łącząca wyniki otrzymane z obu metod.

Jedną z bardziej znanych metod jest ta zaproponowana przez Moody et al. Wykorzystuje ona sygnał z dwóch odprowadzeń. Metodę tą, podobnie jak poprzednie, realizuje się na odfiltrowanym sygnale EKG. Sygnał oddechu wyznaczany jest na podstawie zmiany kąta nachylenia głównego wektora sercowego w stosunku do osi jednego z wyprowadzeń. W celu wyznaczania takiego wektora należy obliczyć pole powierzchni wartości bezwzględnej z kompleksów QRS dla każdego z wyprowadzeń. Następnie, aby wyznaczyć poszukiwane nachylenie, należy obliczyć arctg ze stosunku obu otrzymanych powierzchni (Rys. 1.26). W ten sposób dla każdego kompleksu QRS otrzymujemy jedną wartość nachylenia głównego wektora sercowego. Aby wygenerować ciągły sygnał oddechowy należy otrzymane punkty interpolować funkcjami sklejonymi. Algorytm ten zakłada ortogonalność wyprowadzeń. Jeżeli jednak warunek ten nie jest spełniony to generowany jest stały błąd, który nie ma znaczącego wpływu na wyznaczanie nachylenia głównego wektora sercowego.



Rysunek 1.26: Sposób wyznaczania nachylenia głównego wektora sercowego od osi jednego z wyprowadzeń.

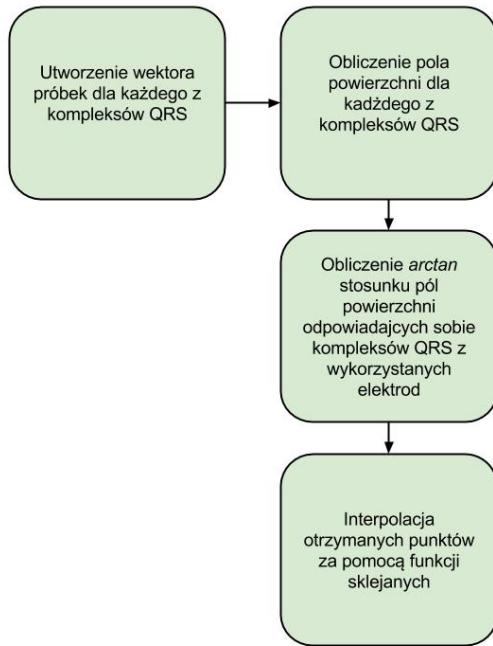
Istnieją również algorytmy wykorzystujące większą liczbę wyprowadzeń. Nie zostały one jednak rozpatrzone, ze względu na wykorzystanie w projekcie jedynie dwóch wyprowadzeń.

1.5.2 Koncepcja proponowanego rozwiązania

Z racji stosunkowo niewielkiego poziomu skomplikowania algorytmów zdecydowano się na implementację dwóch spośród wymienionych, tj. pierwszego (analiza amplitudy pików R) oraz ostatniego (badanie odchylenia głównego wektora sercowego od osi jednego z wyprowadzeń) celem porównania działania.

W przypadku pierwszego algorytmu zostały wykorzystane informacje pochodzące z modułu odpowiedzialnego za wykrywanie pików R (R_PEAKS). Moduł ten wysyła indeksy próbek, dla których został wykryty pik R. Wartości sygnału z próbek o zadanych indeksach są następnie interpolowane metodą funkcji sklejanych. Rząd wielomianu determinowany jest na podstawie najlepszego dopasowania do interpolowanego sygnału. Otrzymana za pomocą interpolacji krzywa stanowi sygnał oddechowy.

Drugi algorytm wykorzystuje informacje pochodzące z modułu odpowiedzialnego za detekcję punktów charakterystycznych sygnału EKG (WAVES). Schemat blokowy algorytmu został przedstawiony na poniższym rysunku (Rys. 1.27):



Rysunek 1.27: Schemat blokowy realizujący odczytywanie sygnału EDR z wykorzystaniem modułu WAVES

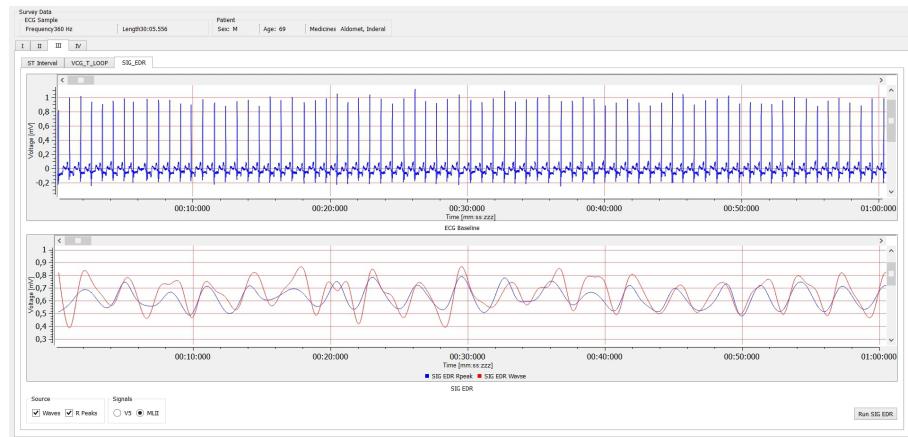
Na podstawie otrzymanych punktów QRSSonset oraz QRSSend wyznaczane jest pole powierzchni każdego z kompleksów QRS, zarówno z pierwszej, jak i z drugiej elektrody. Pole powierzchni kompleksu jest proporcjonalne do średniej amplitudy sygnału. Zakładając ortogonalność osi wyprowadzeń, \arctan ze stosunku pól powierzchni obu kompleksów daje w wyniku kąt odchylenia głównego wektora sercowego od jednej z osi wyprowadzeń. Jeżeli wyprowadzenia

nie są ortogonalne pojawia się błąd pomiaru. Jest on jednak stały dla całego sygnału i nie ma wpływu na wyznaczanie sygnału oddechu pacjenta.

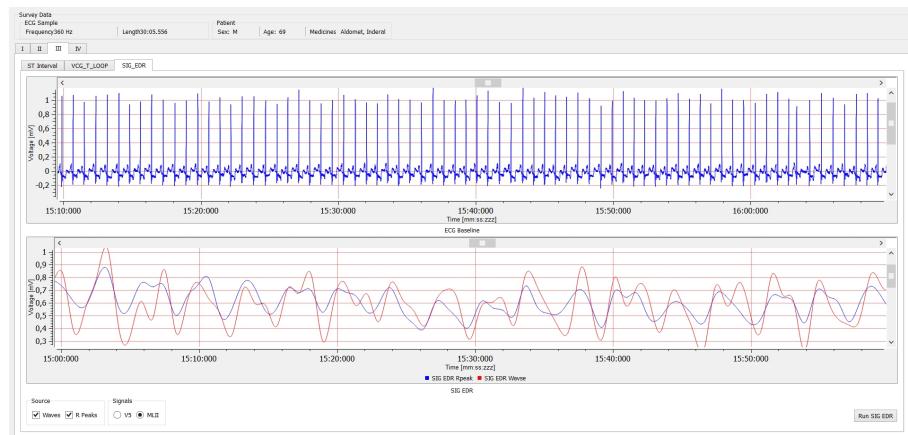
Obie metody charakteryzują się wysoką dokładnością odtwarzania sygnału oddechowego. Nie wymagają wykonywania dużej ilości obliczeń oraz są stosunkowo proste w implementacji. Dodatkową zaletą pierwszej z nich jest wykorzystanie do obliczeń sygnału z jednego odprowadzenia.

1.5.3 Rezultaty i wnioski

Działanie modułu zostało przedstawione na poniższych obrazkach (Rys. 1.28) oraz Rys. 1.29).



Rysunek 1.28: Działanie modułu dla rekordu 100, pierwsza minuta sygnału



Rysunek 1.29: Działanie modułu dla rekordu 100, piętnasta minuta sygnału

Niestety, z braku dostępu do rzeczywistego sygnału oddechowego dla posiadanych danych, przebiegi nie mogły zostać porównane w sposób analityczny. Jedynym sposobem sprawdzania poprawności otrzymanego sygnału EDR było porównanie kształtu przebiegu z sygnałem ECG, a następnie porównanie z wzorcowymi sygnałami, uzyskanymi przy pomocy aparatury do mierzenia oddechu (PRT). Z tej przyczyny skupiono się na przeprowadzeniu porównania otrzymanych danych w celu weryfikacji poprawności zastosowanych algorytmów.

Dane zostały porównane w następujący sposób:

- obliczenie różnicy pomiędzy wartościami próbek stanowiących R-piki z wartościami otrzymanymi w wyniku działania modułu
- porównanie wartości zwracanych w wyniku zastosowania drugiego algorytmu (Moody) z wartościami uzyskanymi z poprzedniego algorytmu
- obliczenie odchylenia poszczególnych różnic od średniej różnicy badanego przebiegu (odchylenie standardowe)

Obliczone wartości zostały zgromadzone w tabeli zbiorczej(Tab. 1.4).

Przeprowadzone badania miały na celu sprawdzenie czy kształt sygnału oddechowego jest taki sam dla każdej z zastosowanych metod. Badana różnica pomiędzy poszczególnymi sygnałami EDR powinna być zatem stała, bądź jak najbardziej zbliżona do stałej. Wynika to z faktu, iż w rzeczywistości mamy do czynienia z jednym sygnałem oddechowym, który jest taki sam dla każdej z zastosowanych metod. Niezależnie od zastosowanej metody, wyniki powinny mieć podobny kształt, mogą się jednak różnić wartościami, wynikającymi z różnych odległości elektrod od serca, a tym samym mocą sygnału. Na wykresie skutkuje to wertykalnym przesunięciem przebiegu i jego lekką deformacją.

Kształt przebiegu sygnału EDR w niewielkim stopniu odbiega od kształtu sygnałów wzorcowych, otrzymanych z pneumatycznej aparatury pomiarowej. Głównymi czynnikami generującymi różnice pomiędzy sygnałami EDR są nienaturalne różnice występujące pomiędzy badanymi próbками, a tym samym wykrytymi pikami R, które są filarem stosowanego algorytmu. Podobne problemy występują w przypadku drugiego z algorytmów, gdyż jest on zależny od obu próbek. Dodatkowo nie wszystkie wykryte obszary QRS, z których ten algorytm korzysta, są poprawne.

1.5.4 Diagram klasy

sigedr
<pre> signalone : const QVector<double> signal_two : const QVector<double> EDRsignal_RPeaks_one : QVector<double> EDRsignal_RPeaks_two : QVector<double> EDRsignal_Waves : QVector<double> EDRsignal : stdvector<double> Integrals_one : QVector<double> Integrals_two : QVector<double> sig_edr(&signal_one : const QVector<double>, &signal_two : const QVector<double>) sig_edr(&signal_one : const QVector<double>, &signal_two : const QVector<double>, &RPeaksIterators_one : const QVector<unsigned int>, &RPeaksIterators_two : const QVector<unsigned int>) sig_edr(&signal_one : const QVector<double>, &signal_two : const QVector<double>, &QRSonsetIterators_one : const vector_it, &QRSonsetIterators_two : const vector_it, &QRSSendIterators_one : const vector_it, &QRSSendIterators_two : const vector_it) new_RPeaks_signal(&signal_num : int, &RPeaksIterators : const QVector<unsigned int>) : void new_Waves_signal(&QRSonsetIterators_one : const vector_it, &QRSonsetIterators_two : const vector_it, &QRSSendIterators_one : const vector_it, &QRSSendIterators_two : const vector_it) : void retrieveEDR_QVec(EDR_type : int, &signal_num : int) : QVector<double>* retrieveEDR_StdVec(EDR_type : int, &signal_num : int) : stdvector<double>* integral(&value : QVector<double>) : double calculate_signal_from_QRS(&QRSIntegrals_one : const QVector<double>, &QRSIntegrals_two : const QVector<double>) : void </pre>

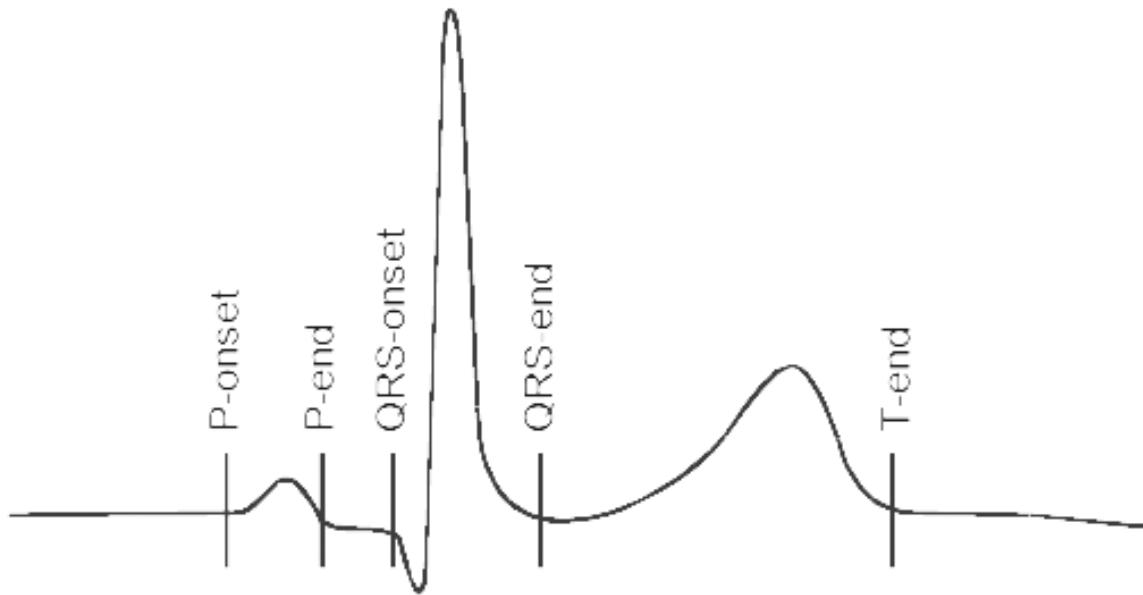
R1	R2	QRS	RR1	R1Q	R2Q	WR1	WR2	WR3
0,514	0,774	0,822	-0,260	-0,307	-0,047	0,554	0,340	0,308
0,580	0,951	0,392	-0,371	0,188	0,559	0,665	0,155	0,298
0,678	0,960	0,797	-0,282	-0,120	0,163	0,576	0,153	0,098
0,658	0,903	0,747	-0,246	-0,090	0,156	0,540	0,123	0,105
0,557	0,911	0,632	-0,354	-0,075	0,279	0,648	0,108	0,018
0,591	0,888	0,572	-0,297	0,019	0,316	0,591	0,014	0,055
0,747	0,991	0,769	-0,244	-0,022	0,222	0,538	0,055	0,039
0,643	0,921	0,713	-0,278	-0,070	0,208	0,572	0,104	0,053
0,563	0,948	0,466	-0,385	0,097	0,482	0,679	0,064	0,221
0,581	0,909	0,699	-0,328	-0,118	0,210	0,622	0,151	0,051
0,670	0,933	0,733	-0,263	-0,063	0,200	0,557	0,096	0,061
0,571	0,878	0,723	-0,307	-0,152	0,155	0,601	0,185	0,106
0,494	0,861	0,468	-0,367	0,026	0,392	0,661	0,007	0,132
0,695	0,936	0,728	-0,241	-0,032	0,208	0,535	0,066	0,052
0,652	0,908	0,685	-0,256	-0,033	0,223	0,550	0,067	0,038
0,512	0,849	0,623	-0,337	-0,111	0,226	0,631	0,144	0,035
0,557	0,919	0,536	-0,362	0,021	0,383	0,656	0,012	0,122
0,718	0,957	0,774	-0,239	-0,056	0,183	0,533	0,089	0,078
0,634	0,881	0,775	-0,247	-0,141	0,106	0,541	0,174	0,155
0,609	0,901	0,564	-0,293	0,045	0,337	0,587	0,011	0,076
0,649	0,965	0,664	-0,316	-0,015	0,301	0,610	0,048	0,041
0,693	0,908	0,772	-0,215	-0,078	0,136	0,509	0,111	0,125
0,647	0,883	0,832	-0,236	-0,185	0,050	0,530	0,219	0,210
0,558	0,929	0,486	-0,371	0,073	0,443	0,665	0,039	0,182
0,626	0,935	0,734	-0,310	-0,108	0,202	0,604	0,141	0,059
0,751	0,959	0,680	-0,208	0,072	0,280	0,502	0,039	0,019
0,581	0,897	0,733	-0,317	-0,153	0,164	0,611	0,186	0,097
0,571	0,929	0,421	-0,358	0,151	0,509	0,652	0,117	0,248
0,777	0,963	0,826	-0,186	-0,049	0,137	0,480	0,082	0,124
0,703	0,911	0,697	-0,208	0,006	0,214	0,502	0,027	0,047
0,567	0,915	0,632	-0,348	-0,065	0,283	0,642	0,098	0,022
0,564	0,968	0,472	-0,404	0,092	0,496	0,698	0,059	0,235
0,716	0,993	0,711	-0,278	0,005	0,282	0,572	0,029	0,021
0,659	0,950	0,690	-0,291	-0,032	0,259	0,585	0,065	0,001

Tablica 1.4: Porównanie wartości próbek R-pików z obu elektrod oraz metody wykorzystującej moduł WAVES. R1 - wartości próbek z pierwszej elektrody, R2 - wartości próbek z drugiej elektrody, QRS - wartości próbek otrzymanych metodą wykorzystującą moduł WAVES, RR1 - różnice pomiędzy wartościami odpowiadających próbek z R1 i R2, R1Q - różnice pomiędzy wartościami odpowiadającymi próbek z R1 i QRS, R2Q - różnice pomiędzy wartościami odpowiadającymi próbek z R2 i QRS, WR1, WR2, WR3 - odchylenia od średniej różnicy dla kolejnych par (odpowiednio RR1, R1Q oraz R2Q)

1.6 Moduł WAVES

1.6.1 Badania literaturowe

Celem niniejszej pracy było zaimplementowanie oraz przetestowanie działania algorytmu detekcji punktów charakterystycznych sygnału EKG. W tym początku i końca zespołu QRS (QRS-onset i QRS-end) oraz fali P (P-onset i P-end). Numery próbek załamków R przyjmowane są jako dane wejściowe wraz z przefiltrowanym sygnałem. Graficznie opisywane punkty przedstawione zostały na rysunku 1.30.



Rysunek 1.30: Sygnał EKG z punktami charakterystycznymi. Źródło: [2]

Zespół QRS jest fragmentem zapisu elektrokardiograficznego odpowiadającego depolaryzacji, czyli pobudzeniu prawej i lewej komory serca. Czas jego trwania oraz kształt niosą wiele informacji na temat aktualnej kondycji serca. Dzięki swojemu charakterystycznemu kształtwi oraz wysokiej energii stanowi on często podstawę wszystkich automatycznych systemów analizy sygnałów EKG. Mimo intensywnych prac na przestrzeni przeszło trzydziestu lat programowe wyznaczanie zespołu QRS wciąż stanowi wyzwanie dla projektantów. Bardzo duże zróżnicowanie morfologiczne analizowanych sygnałów powoduje, że do tej pory nie opracowano detektora działającego ze stu procentową dokładnością. W ostatnich latach dynamiczny rozwój techniki pozwolił na stosowanie czasochłonnych algorytmów o dużej złożoności obliczeniowej (wyjątkiem są tu urządzenia przenośne, ze względu na baterijne zasilanie). Zaowocowało to szeregiem nowych podejść do problemu detekcji punktów charakterystycznych zespołu QRS. W literaturze odnaleźć można algorytmy oparte o sztuczne sieci neuronowe, algorytmy genetyczne, przekształcenia falkowe, banki filtrów cyfrowych, ukryte modele Markowa, transformatę Hilberta oraz heurystyczne metody, w większości oparte o nieliniowe przekształcenia [8] [24]. Cechą wspólną niemal wszystkich rozwiązań jest odpowiednie przygotowanie próbek sygnału do analizy, czyli filtracja sygnału mająca na celu usunięcie zakłóceń mięśniowych, sieciowych 50/60 Hz, zmiennej impedancji elektrod oraz artefaktów powstały wskutek ruchu pacjentów podczas badania. Fala P odpowiada natomiast depolaryzacji przedsionków serca i w

warunkach prawidłowych występuje na schemacie zapisu elektrokardiograficznego przed zespołem QRS. Stanowi ona logiczny początek pojedynczego cyklu pracy serca. Jak w przypadku zespołu QRS automatyczne wyznaczanie początku i końca fali P cieszy się wśród projektantów dużym zainteresowaniem. Podobnie nie stworzono wciąż bezbłędnego algorytmu detekcji, gdyż wykrywanie załamka P nie jest zadaniem prostym. Ma na to wpływ parę czynników, z których można wymienić: zmienną amplitudę fali, możliwą interferencję z innymi falami występującymi w sygnale EKG, a nawet czasem brak występowania. W literaturze można znaleźć wiele przykładów badających problematykę znalezienia załamka P. Stosowane na świecie metody bazują na:

- pochodnych sygnału [9]
- transformacie fazowej sygnału [7] [14]
- filtrowaniu sygnału filtrami górnoprzepustowymi [12]
- transformacie falkowej [11]

W zależności od wybranej metody algorytmy różnią się między sobą dokładnością jak i złożonością obliczeniową. Autorzy algorytmu bazującego na 9 punktowej pochodnej sygnału [9] zapewniają o skuteczności ok 90 procent, co nie jest wartością przekonującą. W przypadku zastosowania transformaty fazowej wyniki znalezione w publikacji [7] są obiecujące: ponad 98% procentowa wykrywalność oraz mała złożoność obliczeniowa. Skuteczność algorytmu mogła być ostatecznie sprawdzona podczas realizacji Projektu EKG [14] w ramach zajęć laboratoryjnych z przedmiotu Elektroniczne systemy diagnostyki medycznej i terapii w roku 2012/2013, kiedy został on zaimplementowany właśnie do wykrywania fali P. Niestety opracowane wyniki nie dają jednoznacznej odpowiedzi na pytanie co do przydatności algorytmu. Wynika to z tego, iż w pracy [14] znajdują się jedynie procentowe wykrycia ilości początków fali i końców P w stosunku do znalezionych załamków R w sygnale. O ile w przypadku badania jakości wykrycia zespołów QRS taka metoda jest akceptowalna, to wiązanie ilości wykrytych początków i końców fali P z ilościami wykrytych załamków R jest niewłaściwe. Nie wiadomo także, czy wykryte załamki to faktycznie załamki P, czy może kawałki uchwyconego załamka T. Dlatego też, jedynym możliwym sposobem na weryfikację algorytmu jest porównanie otrzymanych punktów z wartościami referencyjnymi, którą zaproponowano w tej pracy.

1.6.2 Koncepcja proponowanego rozwiązania

1.6.2.1 QRS-onset i QRS-end

W trakcie realizacji moduły Waves, do wyznaczenia punktów charakterystycznych zespołu QRS zdecydowano się na algorytm bazujący na wyznaczaniu wskaźnika będącego wartością obszaru wyznaczonego obwiednią analizowanego zespołu. Algorytm ten, opisany szczegółowo w pracy [6], można podzielić na cztery etapy, które zostały opisane poniżej (Detekcja załamka R opisana w pracy [6] została pominięta gdyż pozycja załamka R przyjmowana jest jako parametr wejściowy z innego modułu).

Filtracja pasmowa Celem filtracji jest przekształcenie sygnału faworyzując cechy zespołu QRS, jednocześnie osłabiając cechy innych elementów elektrokardiogramu i zakłóceń. Częstotliwości odcięcia filtra pasmowego są odmienne dla detekcji QRS-onset i QRS-end i wynoszą odpowiednio [0.5 40] Hz oraz [5 30] Hz. Realizację filtra oparto o algorytm FFT

(ang. Fast Fourier Transform). W tym celu wyznaczono transformatę całego sygnału, zastąpiono niechciane częstotliwości zerami, a następnie obliczono odwrotną transformatę wykorzystując algorytm IFFT (ang. Inverse Fast Fourier Transform).

Wyznaczanie Obwiedni Obwiednia zbudowana jest z przefiltrowanego sygnału (części rzeczywistej) oraz jego przekształcenia Hilberta (części urojonej). Transformata Hilberta rzeczywistego sygnału $x(t)$ zdefiniowana jest wzorem 1.11.

$$x_H = \mathcal{H}\{x(t)\} = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{x(\lambda)d\lambda}{t - \lambda} \quad (1.11)$$

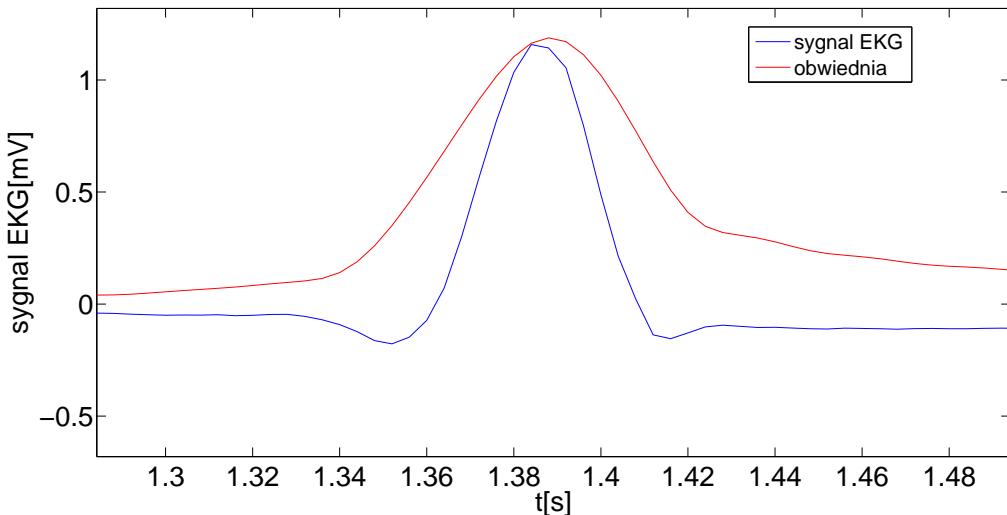
Zamiast korzystać bezpośrednio z definicji można także wyznaczyć ją w domenie częstotliwości zgodnie z 1.12.

$$X_H(j\omega) = X(j\omega) \cdot [-j \cdot \text{sgn}(\omega)] \quad (1.12)$$

Następnie jeśli jako ECG oznaczymy rzeczywistą część sygnału przefiltrowanego, a jako ECGH część urojoną przekształcenia Hilberta sygnału to obwiednie możemy zdefiniować jako 1.13.

$$ECG_e(k) = \sqrt{ECG^2(k) + ECG_H^2(k)} \quad (1.13)$$

Rysunek 1.31 pokazuje przykładowy zespół QRS wraz z wyznaczoną dla niego obwiednią



Rysunek 1.31: Sygnał EKG wraz z odpowiadającą mu obwiednią.

Należy zauważyć, że wypłaszczenia obwiedni zbiegają się z początkiem i końcem zespołu QRS. Fakt ten zostanie wykorzystany w późniejszych krokach do detekcji punktów QRS-onset i QRS-end.

Wyznaczanie Okna W celu ograniczenia zakresu poszukiwania punktów charakterystycznych zespołu QRS, zdefiniowane zostały wstępne okna w każdym cyklu pracy serca. Ze względu na czas trwania zespołu od 60 do 110 ms zdecydowano się na przeszukiwanie sygnału 100

ms przed wystąpieniem załamka R i 100 ms po nim odpowiednio dla QRS-onset i QRS-end. Taka szerokość okna powinna zapewnić objęcie pożądanych punktów, jednocześnie nie zachodząc na falę P i T. W związku z powyższym skuteczność algorytmu w sposób oczywisty zależy od poprawności wykrycia załameków R.

Wyznaczanie wskaźnika powierzchni Ostatnim i kluczowym krokiem algorytmu jest wyznaczenie wskaźnika powierzchni w oknie przesuwnym. Wszystkie poniższe czynności wykonywane są dla każdego cyklu serca, tylko w opisanych w poprzednim kroku oknach. Algorytm zostanie opisany na podstawie detekcji punktu QRS-end, dla ułatwienia rozpatrzony zostanie czas ciągły. Dla punktu QRS-onset postępowanie jest analogiczne i nie będzie w pracy opisane. Dla poniższych rozważań przyjęto następujące oznaczenia (patrz rysunek 1.32):

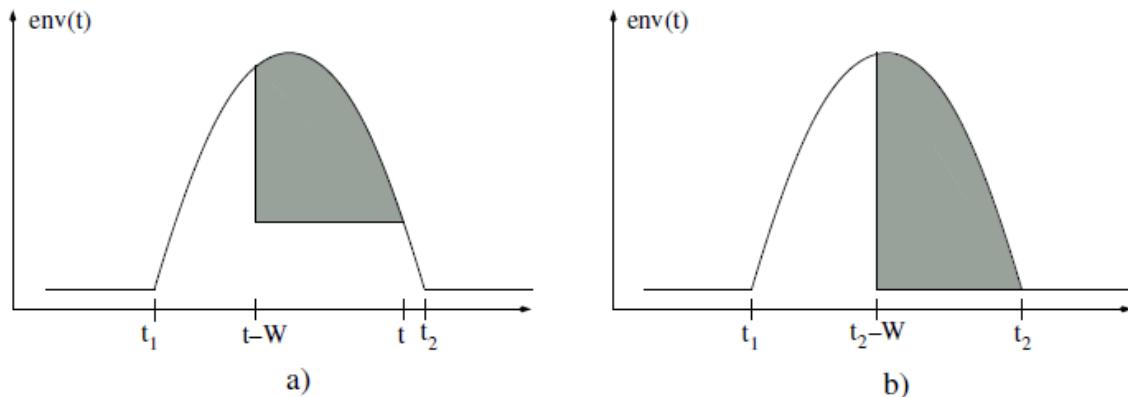
$\text{env}(t)$ - obwiednia sygnału w czasie,

t_1 i t_2 początek i koniec obwiedni,

t_p - punkt szczytowy obwiedni,

$L = t_1 - t_2$ - długość obwiedni pojedynczego zespołu QRS.

W - szerokość okna przesuwnego



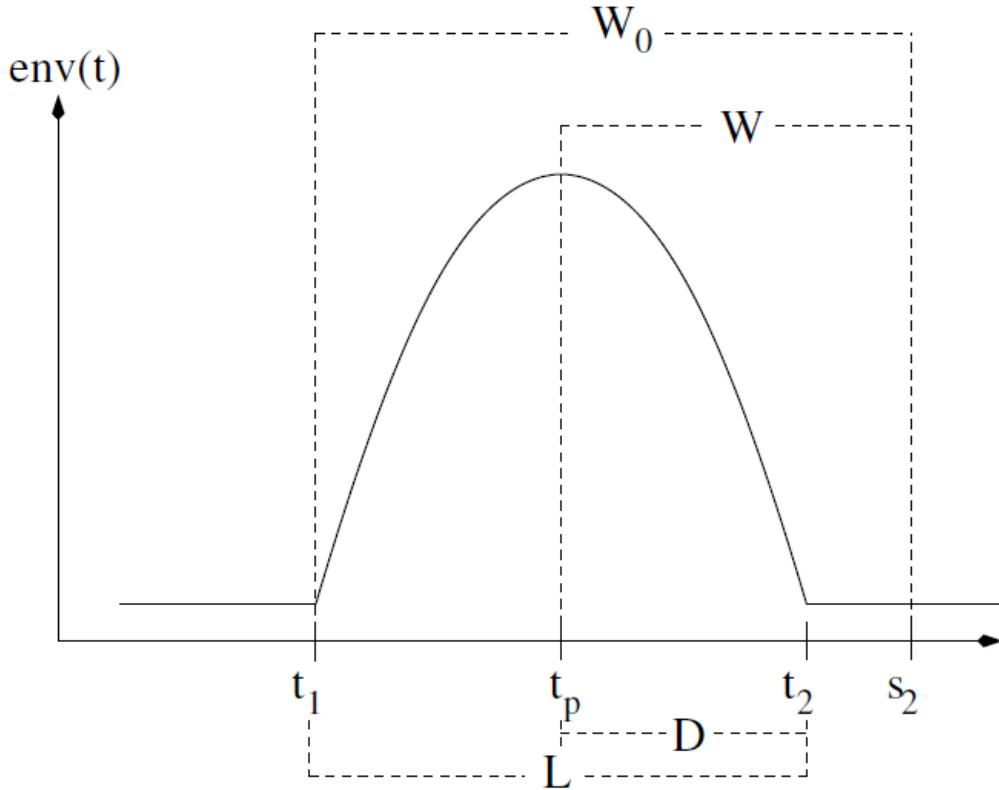
Rysunek 1.32: Ilustracja wskaźnika powierzchni $A(t)$ - szare obszary, Źródło: [6]

Wykorzystany algorytm opiera się na wyznaczeniu wskaźnika $A(t)$, który osiąga maksymalną wartość dla czasu równego t_2 . Wyznaczany jest on przez całkowanie w oknie przesuwnym o szerokości W (wyznaczenie wartości W omówione będzie w dalszej części). W każdej chwili czasu wskaźnik może być wyliczony jako 1.14:

$$A(t) = \int_{t-W}^t [\text{env}(\tau) - \text{env}(t)] d\tau \quad (1.14)$$

Wskaźnik ten jest polem powierzchnią pod wykresem obwiedni w obszarze $[t-W, t]$, ale nad poziomą linią przechodzącą przez punkt $(t, \text{env}(t))$. Zobrazowano to na rysunku 1.32. W związku z dużym zróżnicowaniem wartości L , szerokość okna W będzie osobno strojona dla każdego zespołu QRS. Jak wynika z rysunku 1.33, dla poprawnej detekcji QRS-end, wskaźnik powierzchni 2.4 powinien być wyznaczany w oknie W spełniającym zależność: $t_2 - t_p < W < L$.

Kiedy wartość W jest zbyt duża, przykładowo równa W_0 , wtedy wskaźnik przyjmuje największą wartość dla końca okna, w punkcie oznaczonym s_2 na rysunku 1.33. Jeśli W_0 nie jest zbyt duże, punkt s_2 powinien leżeć wystarczająco blisko t_2 aby spełniać warunek



Rysunek 1.33: Dwuetapowy wybór okna W , Źródło: [6]

$s_2 - t_p < L$. W związku z tym brana jest wartość szerokości okna równa $W = s_2 - t_p$ i powtórnie wyznaczany jest wskaźnik $A(t)$. Podsumowując powyższe rozważania krok algorytmu można podzielić na dwie fazy:

1. Obliczenie wskaźnika powierzchni dla okna przesuwnego $W = W_0$, gdzie W_0 jest szerokością okna opisaną w poprzednim kroku (100 ms). Wyznaczenie wartości s_2 maksymalizującej wskaźnik $A(t)$ oraz wartości t_p będącej maksimum obwiedni w oknie W_0 .
2. Ponowne obliczenie wskaźnika powierzchni dla okna przesuwnego o szerokości $W = s_2 - t_p$. Punkt QRS-end jest punktem maksymalizującym wskaźnik $A(t)$.

1.6.2.2 P-onset i P-end

Brak rzetelnej oceny we wcześniej pracy [14] spowodował, że podjęto decyzję o ponownej implementacji algorytmu wykrywania fal P za pomocą transformaty fazowej i poprawnej weryfikacji jego skuteczności. Szczegółowe informacje co do metody można znaleźć w publikacji [7] Transformata fazowa sygnału polega na zamienieniu próbki EKG we wskaz w płaszczyźnie zespolonej wg wzoru 1.15:

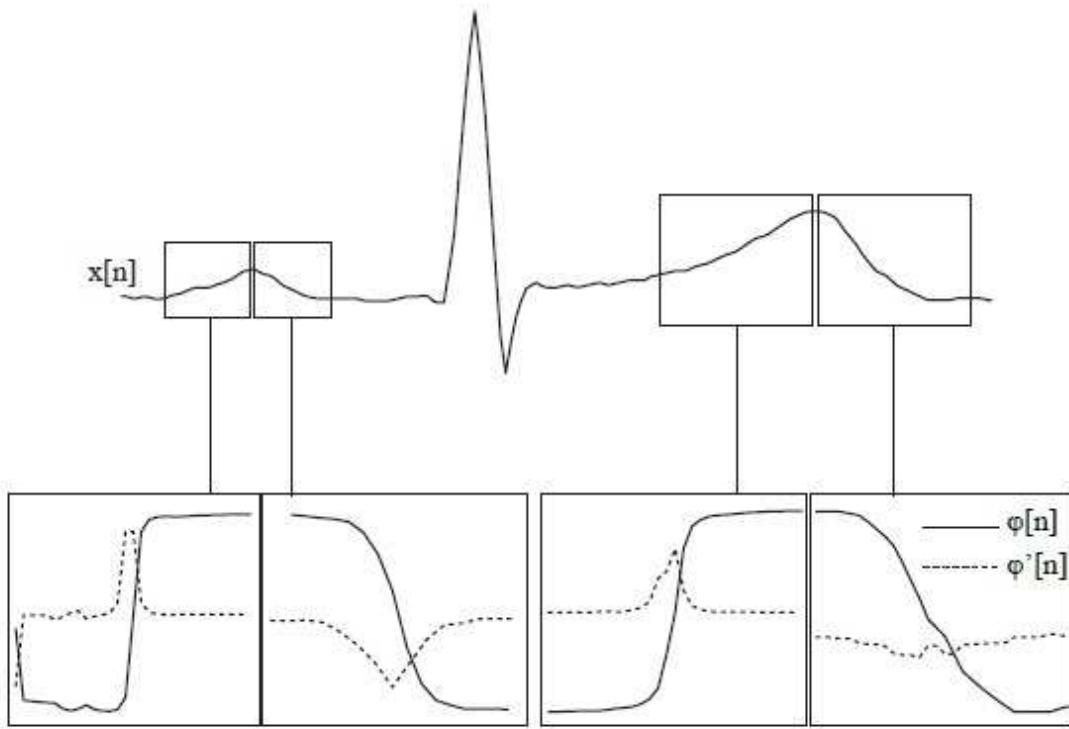
$$y[n] = Rv + j \cdot x[n], dlan = 1, 2, \dots, N \quad (1.15)$$

Sygnał po transformacie składa się zatem ze rzeczywistej stałej części Rv i urojonej części $x[n]$, będącą wartością sygnału w próbce n . Moduł i faza wskazu jest obliczany jako:

$$M[n] = \sqrt{Rv^2 + x[n]^2}$$

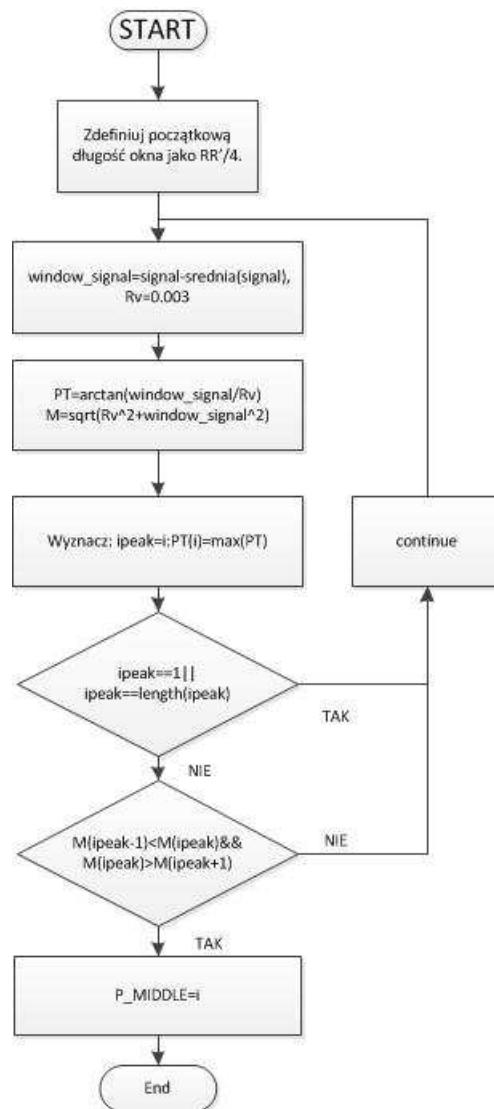
$$\phi[n] = \arctan\left(\frac{x[n]}{Rv}\right)$$

Wykrywanie fali P dokonuje się na podstawie badania wartości fazy ϕ , obliczanej w odpowiednim oknie.

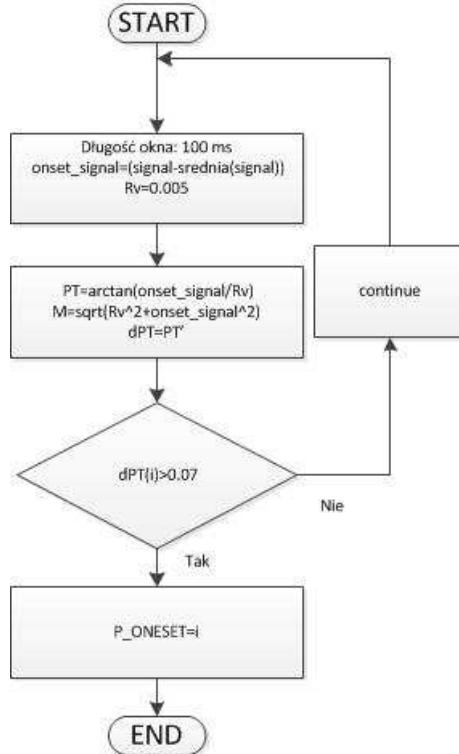


Rysunek 1.34: Zależność między fazą ϕ a badanym sygnałem EKG. Źródło: [7]

Dzięki tej transformacji można odwzorować niewielką amplitudę fali P w funkcję arcustangens zawierającym wartości z przedziału $[-\pi/2, \pi/2]$, co ułatwia jej wykrycie. Wykrywanie fali P za pomocą tego algorytmu polega najpierw na znalezieniu maksimum lokalnego fali P (P-middle), a następnie dokonania niezależnego wykrycia początku i końca fali (P-onset i P-end). Schematy blokowe wykrycia punktów P-middle i P-onset (P-end jest analogiczny) są zamieszczone na rysunkach 1.35 oraz 1.36



Rysunek 1.35: Schemat blokowy algorytmu wykrywającego P-MIDDLE



Rysunek 1.36: Schemat blokowy algorytmu wykrywającego P-ONSET (dla P-END jest podobny)

1.6.3 Rezultaty i wnioski

Ewentualne użycie oprogramowania do detekcji zespołów QRS w urządzeniach medycznych wymaga dokładnego testowania i oceny skuteczności. Zgodnie z literaturą [8] działanie algorytmów ocenia się na podstawie dwóch parametrów: czułości - S_e (ang. sensitivity) i poprawnej wykrywalności - $+P$ (ang positive predictivity). Wskaźniki te definiuje się odpowiednio jako 1.16, 1.17:

$$S_e = \frac{TP}{TP + FN} \quad (1.16)$$

oraz

$$+P = \frac{TP}{TP + FP} \quad (1.17)$$

gdzie:

TP - (ang. true positive) ilość próbek wykrytych poprawnie

FP - (ang. false positive) ilość próbek wykrytych błędnie

FN - (ang. false negative) ilość zespołów QRS dla których algorytm nie zwrócił wyników

Implementacja algorytmu zespołu QRS wyklucza możliwość występowania FN, dla każdego wejściowego załamka R zawsze zwraca parę punktów oznaczających QRS-onset o QRS-end. W związku z tym czułość algorytmu zawsze wynosi jeden. Inaczej jest dla algorytmu detekcji fali P. Ponieważ jak pisano wcześniej fala P może być przysłonięta, algorytm najpierw dokonuje oceny czy fala P istnieje. Jeśli stwierdzi brak fali P naturalnie nie wyznacza jej punktów charakterystycznych. W takiej sytuacji może zaistnieć FN, w przypadku gdy algorytm pominie faktycznie istniejącą falę P. Dlatego, w tym przypadku wyznaczane będą oba wskaźniki. Ponadto

algorytm bazuje na wartości interwału pomiędzy kolejnymi załamkami R, dlatego zwraca jedną parę mniej wykrytych początków i końców fali P niż podano wejściowych załamków R. W związku z powyższym konieczne jest też podanie kolejnych R-peaków, bez przerw. Założenie to nie jest spełnione dla wszystkich adnotacji z wykorzystanej bazy referencyjnej, dlatego do testów tego algorytmu brano tylko część adnotacji do pierwszej przerwy. Testy przeprowadzono wykorzystując sygnały z bazy QT database dostępnej na stronie <http://physionet.org>. Baza ta zawiera sto-pięć piętnastominutowych sygnałów z zaznaczonymi przez ekspertów punktami charakterystycznymi dla zespołu QRS, fali P, T oraz U. W każdym sygnale zostało oznaczone co najmniej 30 cykli. Szczegółowy opis bazy danych odnaleźć można w pracy [13]. Na potrzeby testów, w celu uniknięcia propagacji błędów z innych modułów wykorzystano referencyjne położenia załamków R jako daną wejściową. Punkt uznawano za poprawnie wykryty gdy jego odległość od odpowiedniego punktu referencyjnego była mniejsza od pięciu próbek. Wartości wskaźnika poprawnej wykrywalności (+P), a w przypadku fali P także czułości (S_e) zebrane w tabeli 1.5 oraz tabeli 1.6.

Tablica 1.5: Wyniki algorytmu poszukującego QRS-onset i QRS-end

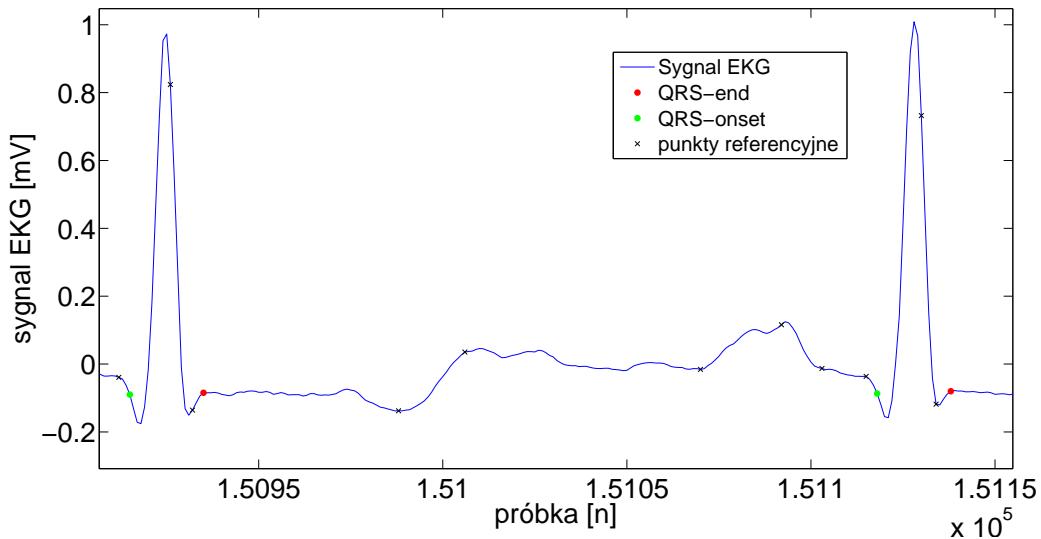
sygnał	ilość cykli	TP QRS_{onset}	TP QRS_{end}	+P QRS_{onset}	+P QRS_{end}
sel100	30	29	30	96,67	100,00
sel103	30	29	29	96,67	96,67
sel114	50	37	47	74,00	90,00
sel116	50	45	49	90,00	98,00
sel117	30	30	28	100,00	93,33
sel123	30	29	28	96,67	93,33
sel213	30	24	25	80,00	83,33
sel223	31	29	29	93,55	93,55
sel230	50	37	50	74,00	100,00
sel231	50	50	47	100,00	94,00
sel233	30	29	28	96,67	93,33
sel803	30	27	29	90,00	96,67
sel808	30	30	29	100,00	96,67
sel811	30	27	29	90,00	96,67
sel820	30	30	29	100,00	96,67
średnia:				92,00	96,40

Jak pokazuje tabela 1.5 algorytm detekcji punktów charakterystycznych zespołu QRS działa z zadowalającą dokładnością. Świadczy to o poprawności implementacji algorytmu w stworzonym module. Przykładowe działanie algorytmu detekcji punktów QRS-onset i QRS-end dla sygnału sel100 pokazano na rysunku 1.37.

Rysunek 1.37 pokazuje także że jakość dostępnych adnotacji nie jest najlepsza. Punkty oznaczone jako Qrs-end znajdują się na załamkach S, co może spowodować (dla silnie ujemnych wartości załamka S) zaklasyfikowanie poprawnie wykrytego punktu jako błędny. Najsłabsze wyniki dla QRS-onset, różniące się o 18% od średniej, uzyskano dla sygnału sel114. Wyniki +P dla punktu QRS-end nie spadły dla żadnego z testowanych sygnałów poniżej 90%. Ponownie najsłabiej algorytm poradził sobie z sygnałem sel114, dla którego wynik różnił się od średniej o 6,4%. Sytuacja taka ma miejsce ze względu na kształt sygnału sel114. Występuje tam przy załamku R drugi, mający zbliżoną wartość, peak sygnału. Powoduje to zaburzenie kształtu ob-

Tablica 1.6: Wyniki algorytmu poszukującego P-onset i P-end

sygnał	ilość cykli	TP P_{onset}	TP P_{end}	FN P_{onset}	FN P_{end}	S_e P_{onset}	S_e P_{end}	+P P_{onset}	+P P_{end}
sel100	29	26	29	0	0	100,00	100,00	89,66	100,00
sel103	29	29	24	0	0	100,00	100,00	100,00	80,76
sel114	29	18	23	2	2	90,00	92,00	66,67	85,19
sel116	49	30	31	0	0	100,00	100,00	61,22	63,27
sel117	29	20	21	0	0	100,00	100,00	68,97	72,41
sel123	29	21	22	0	0	100,00	100,00	72,41	75,86
sel213	29	3	20	0	0	100,00	100,00	10,34	68,97
sel223	30	29	27	0	0	100,00	100,00	96,67	90,00
sel230	33	13	12	20	20	39,39	37,5	100,00	92,31
sel231	49	41	47	0	0	100,00	100,00	83,67	95,92
sel233	29	15	14	0	0	100,00	100,00	51,72	48,28
sel803	29	24	26	0	0	100,00	100,00	82,76	89,66
sel808	29	10	28	0	0	100,00	100,00	34,48	96,55
sel811	29	23	28	0	0	100,00	100,00	79,31	96,55
sel820	27	21	24	1	1	95,45	96,00	77,78	88,89
		średnia:		94,99	95,03	71,71	82,97		

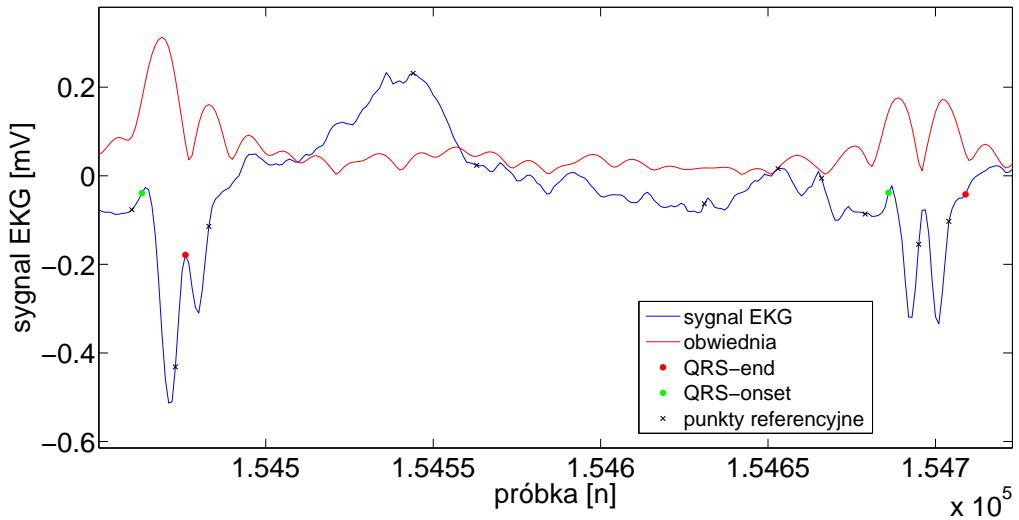


Rysunek 1.37: Sygnał sel100 z bazy QT-Database

wiedni, która powinna być funkcją dzwonową obejmującą cały zespół QRS. Skutkuje to błędny działaniem algorytmu. Sytuację przedstawia rysunek 1.38.

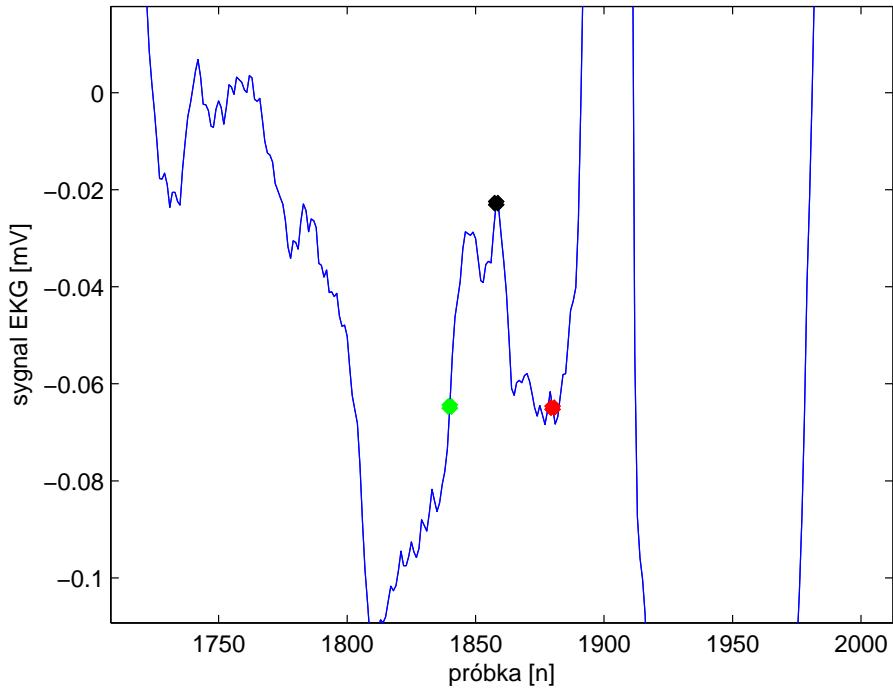
Jak widać na rysunku 1.38, algorytm dla pierwszego zespołu QRS wykrywa jego koniec w miejscu, w którym "podwojona obwiednia" osiąga minimum. W przyszłości błęd ten można rozwiązać badając czy obwiednia ma dwa maksima lokalne. Jeśli tak należy poszukiwać końca "drugiej części" obwiedni.

W przypadku fali P czułość algorytmu jest zadowalająca, choć zdarzają się rekordy, gdzie spada



Rysunek 1.38: Trudny przypadek dla algorytmu detekcji QRS-end oraz QRS-onset

ona poniżej 50% (według tabeli 1.6). Gorzej jest natomiast z wykrywalnością: średnio 71% dla fali P-ONSET i prawie 83% procent dla fali P-END. Dzieje się tak wskutek odbiegającego od oryginału kształtu załamka P. W badanych sygnałach bardzo często fala P przyjmowała kształt przedstawiony na rysunku 1.39:



Rysunek 1.39: Przykładowy załamek P z oznaczonym początkiem P-ONSET (kolor zielony), maksimum P-MIDDLE (kolor czarny) oraz końcem P-END(kolor czerwony).

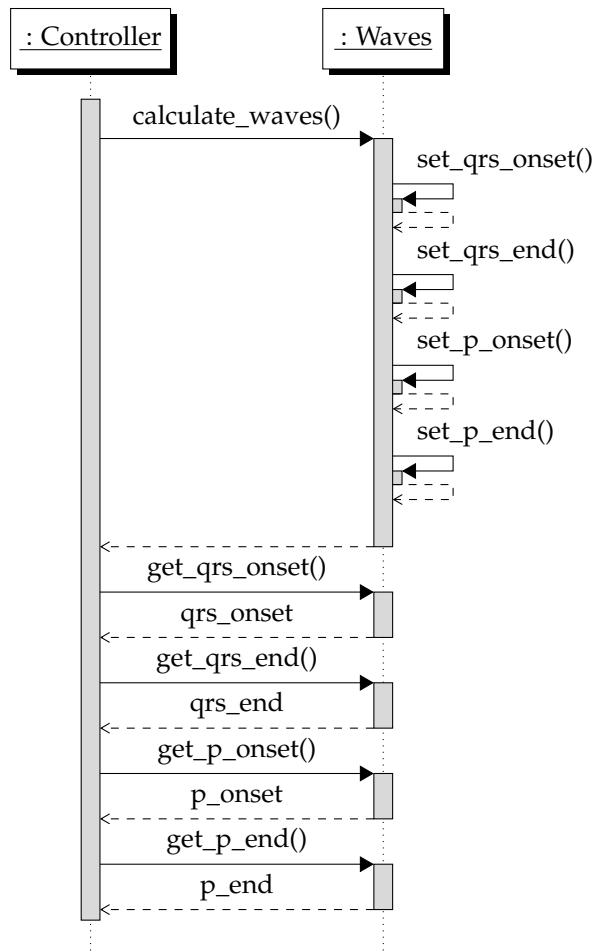
Problem tkwi w położeniu P-MIDDLE; gdy znajduje się on zaraz na początku fali lub na jej końcu, okno (o długości 100 ms) łapie za długi odcinek sygnału leżący poza falą P i na niej może znaleźć nieprawidłowe punkty. Takie sytuacje, jak się okazało w badanych sygnałach zdarzają się bardzo często, dlatego też często punkty nie zgadzają się z punktami naniesionymi przez ekspertów. Problem nie jest łatwy do rozwiązania, gdyż stawiając ostrzejszy warunek na pochodną fazy otrzymujemy punkty P-ONSET i P-END na fali P, ale są one wtedy zbyt późne (dla początku) i zbyt wcześnie dla (dla końca) w stosunku do wartości referencyjnych, więc mimo optycznego polepszenia sytuacji (wykryte punkty nie wypadają poza falę P), wykrywalność algorytmu staje się gorsza (gdyż przyjmuje się, że punkt został wykryty prawidłowo, gdy między nim a wartością referencyjną występuje maks 5 próbek różnicy). Doświadczenia pokazały, że algorytm w postaci proponowanej w publikacji [7] może być zastosowany jedynie do zgrubnej detekcji granic fali P, gdyż nie jest wystarczająco dokładny i dla bardzo zniekształconej fali P daje słabe rezultaty. Aby zwiększyć jego wykrywalność, należało by może wprowadzić zmienną długość okna po wykryciu punktu P-middle w zależności od zaistniałej sytuacji lub też po wykryciu tego punktu zastosować inny algorytm.

1.6.4 Diagramy klasy i sekwencji

1.6.4.1 Diagram klasy

Waves
<pre> - qrs_onset : QVector<QVector<double>::const_iterator> - qrs_end : QVector<QVector<double>::const_iterator> - p_onset : QVector<QVector<double>::const_iterator> - p_end : QVector<QVector<double>::const_iterator> - fs : double +calculate_waves(ECG : QVector<double> &, R_peaks : vector_it&, Fs : double) +get_qrs_onset() : const QVector<QVector<double>::const_iterator> & +get_qrs_end() : const QVector<QVector<double>::const_iterator> & +get_p_onset() : const QVector<QVector<double>::const_iterator> & +get_p_end() : const QVector<QVector<double>::const_iterator> & -set_qrs_onset(ECG : QVector<double>&,R_peaks : vector_it&) -set_qrs_end(ECG : QVector<double>&,vector_it&) -set_p_onset(ECG : QVector<double>&, R_peaks : vector_it&) -set_p_end(ECG : QVector<double>&, R_peaks : vector_it&) </pre>

1.6.4.2 Diagram sekwencji



1.7 Moduł HRV1

1.7.1 Badania literaturowe

Moduł HRV1(ang. heart rate variability) służy do analizy zmienności rytmu serca. Przeprowadza się ją, badając różnice w długościach interwałów RR wyznaczanych przez szczyty zespołów QRS sygnału EKG. Stosuje się do tego metody analizy statystycznej w dziedzinie czasu oraz częstotliwościowej. Dane dla tego modułu to ciąg próbek załamków R otrzymywany z modułu R_PEEAKS, natomiast szukane to parametry obu analiz oraz wykres postaci częstotliwościowej tachogramu wraz z naniesionymi zakresami obliczonych parametrów. Problem analizy zmienności rytmu serca rozwiązuje się korzystając z wyżej wymienionej metody analizy statystycznej (czasowej i częstotliwościowej), jednak poza zwykłym obliczaniem parametrów korzysta się również z :

- modeli parametrycznych np. periodogram –wykorzystywany jako estymator w analizie widmowej,
- modeli nieparametrycznych np. ARMA (ang. auto regressive and moving average) – regresja statystyczna przyszłych wartości szeregu czasowego.

Odpowiedni dobór modelu podczas rozwiązywania problemu ma istotny wpływ na diagnozę oraz analizę złożonych parametrów fizjologicznych człowieka.

1.7.2 Koncepcja omawianego rozwiązania

Koncepcja rozwiązania problemu odpowiedniej analizy zmienności rytmu serca jest po-dzielona na dwie metody, analizę statystyczną czasową oraz analizę statystyczną częstotliwościovą.

1.7.2.1 Analiza statystyczna czasowa

Jest to najprostsza metoda wyliczająca szereg współczynników diagnostycznych. Za pomocą biblioteki ALGLIB, w której znajdują się odpowiednie funkcje matematyczne oraz posiadając wiedę ze statystyki matematycznej wyliczono parametry diagnostyki takie jak :

- wartość średnia interwałów RR [ms],

$$\overline{RR} = \frac{1}{N} \sum_{i=1}^N RR_i$$

- odchylenie standardowe interwałów RR [ms],

$$SDNN = \sqrt{\frac{1}{N-1} \sum_{i=1}^N (\overline{RR} - RR_i)^2}$$

- pierwiastek kwadratowy ze średniej kwadratów różnic pomiędzy kolejnymi dwoma interwałami RR [ms],

$$RMSSD = \sqrt{\frac{1}{N-1} \sum_{i=1}^{N-1} (RR_{i+1} - RR_i)^2}$$

- liczba interwałów RR, których różnica przekracza 50 [ms],

$$NN50 = \sum_{i=1}^{N-1} f_i$$

$$f_i = \begin{cases} 1 & \text{gdy } |RR_{i+1} - RR_i| > 50 \\ 0 & \text{gdy } |RR_{i+1} - RR_i| \leq 50 \end{cases}$$

- odsetek różnic pomiędzy interwałami RR, które przekraczają 50 ms [%],

$$pNN50 = \frac{NN50}{N-1} \times 100\%$$

- odchylenie standardowe ze wszystkich średnich interwałów RR w 5 minutowych segmentach czasu całego zapisu [ms],

SDANN

- średnia z odchyleń standardowych interwałów RR w 5 minutowych segmentach czasu całego zapisu [ms],

SDANNindex

- odchylenie standardowe różnic pomiędzy dwoma sąsiadującymi interwałami RR [ms].

SDSD

gdzie:

- N - ilość próbek tachogramu
- RR_i - wartość i-tego interwału RR

1.7.2.2 Analiza statystyczna częstotliwościowa

Jest to bardziej złożona metoda analizy parametrów zmienności rytmu serca, operująca się o dyskretną transformację Fouriera, rzadziej funkcję autokorelacji. Do jej zastosowania użyto biblioteki ALGLIB. Przed zastosowaniem samej transformacji Fouriera skorzystano z krzywej splajn odnośnie danych wejściowych. Dzięki takim operacjom obliczono następujące parametry analizy częstotliwościowej:

- całkowita moc widma (≤ 0.4 Hz) [ms^2],

TP

- moc widma w zakresie wysokich częstotliwości (0.15 - 0.4 Hz) [ms^2],

HF

- moc widma w zakresie niskich częstotliwości (0.04 - 0.15 Hz) [ms^2],

LF

- moc widma w zakresie bardzo niskich częstotliwości (0.003 - 0.04 Hz) [ms^2],

VLF

- moc widma w zakresie ultra niskich częstotliwości (≤ 0.003 Hz) [ms^2],

ULF

- stosunek mocy widm w zakresie niskich częstotliwości do wysokich częstotliwości.

$$LHFH = \frac{LF}{HF}$$

Po dokonaniu wyliczeń wszystkich parametrów analizy częstotliwościowej ustawiono dane potrzebne do narysowania wykresu widma mocy sygnału EKG z odpowiednimi przedziałami mocy. Przykłady takich wykresów są umieszczone w sekcji 1.7.3.

1.7.3 Realizacja i wnioski

Po zaprogramowaniu wszystkich koniecznych funkcji wyliczających odpowiednie parametry czasowe i częstotliwościowe oraz funkcji odpowiedzialnych za rysowanie wykresów widma mocy sygnału EKG przystąpiono do przetestowania działania modułu na 5 różnych danych wejściowych. W tabeli 1.7 widoczne są wszystkie wyliczone parametry analizy statystycznej dla 5 różnych sygnałów EKG.

Tablica 1.7: Analiza statystyczna

	1	2	3	4	5
Mean[ms]	794.15	865.42	932.61	811.53	752.76
SDNN[ms]	48.89	48.70	331.29	135.28	192.25
RMSSD[ms]	63.36	35.72	497.84	135.56	282.50
RR50	219.00	211.00	1266.00	534.00	335.00
RR50 Ratio[%]	9.64	10.13	65.49	24.03	13.98
SDANN[ms]	33.05	20.74	72.86	14.80	19.38
SDANN Index[ms]	42.83	42.27	307.49	117.46	122.20
SDSD[ms]	54.69	24.01	377.84	118.57	273.63

Jak widać z tabeli 1.7 parametry dla pomiaru 1 oraz 2 zgadzają się z typowymi wartościami dla sygnału EKG przeprowadzonego na zdrowym pacjencie. Pozostałe pomiary a szczególnie

pomiar 3 oraz 5 posiadają wartości znaczco odbiegające od odpowiednich, co może świadczyć o tym, że pacjent posiada zaburzenia pracy serca. Dla porównania na rysunku 1.40 przedstawiono parę parametrów analizy czasowej zmienności rytmu serca u zdrowego człowieka w wieku od 40 do 69 lat wg. Biggera

Parametry	Wartość średnia \pm SD
Odstęp <i>RR</i> (ms)	817 ± 103
SDNN (ms)	141 ± 39
Wskaźnik SDANN (ms)	127 ± 35
Wskaźnik SDNN (ms)	54 ± 15
rMSSD (ms)	27 ± 12
pNN50 (%)	9 ± 7

Rysunek 1.40: Analiza czasowa wg. Biggera

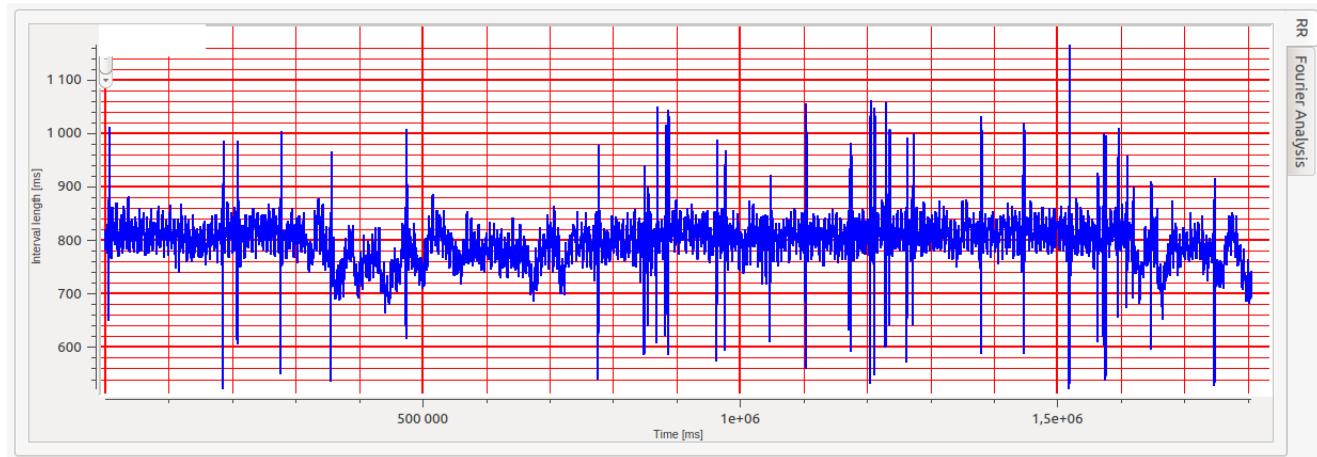
Powyższe rozważania dotyczą jedynie charakterystyki czasowej statycznej, natomiast dużo więcej można się dowiedzieć przeprowadzając charakterystykę częstotliwościową. W tabeli 1.8 umieszczone są wyliczone parametry analizy częstotliwościowej dla takich samych sygnałów jak w przypadku analizy statycznej.

Tablica 1.8: Analiza częstotliwościowa

	1	2	3	4	5
TP[ms ²]	3190.00	4132.00	208550.00	17268.00	238756.00
HF[ms ²]	1846.00	1384.00	83800.00	8581.00	11153.00
LF[ms ²]	179.00	478.00	38673.00	4440.00	127614.00
VLF[ms ²]	627.00	1750.00	73051.00	3209.00	87569.00
ULF[ms ²]	536.00	519.00	13024.00	1036.00	12419.00
LFHF[%]	9.73	34.56	46.15	51.75	1144.17

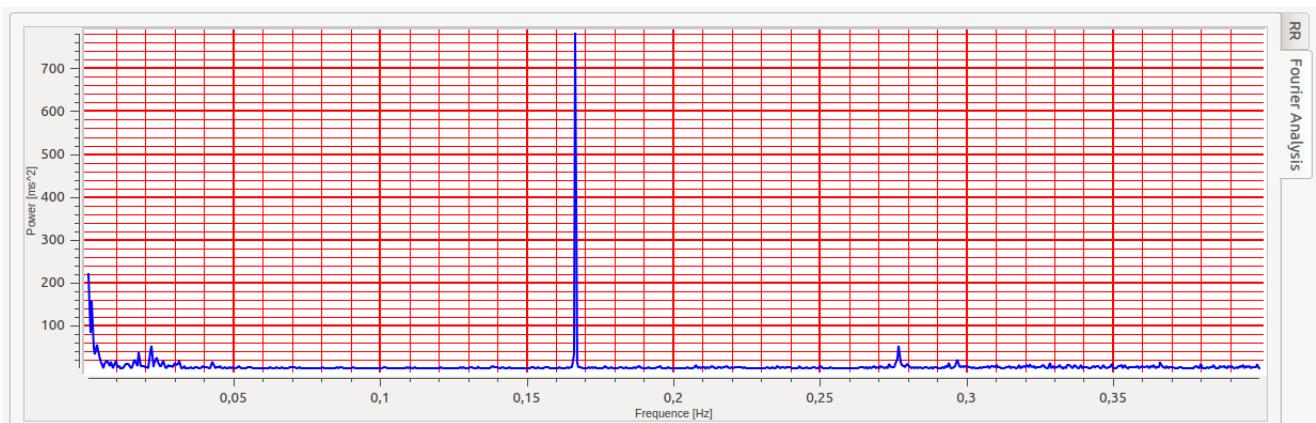
Oceniając wartości poszczególnych parametrów analizy częstotliwościowej widoczne w tabeli 1.8 dochodzi się do podobnego wniosku jak w przypadku analizy czasowej. Wartości w pomiarach 3, 4 oraz 5 odbiegają od parametrów zmienności rytmu serca u zdrowych dorosłych osób.

Kolejnym krokiem w analizie częstotliwościowej jest narysowanie wykresu mocy widma wyrażonego w milisekundach do kwadratu w stosunku do częstotliwości odpowiedniego zakresu widma. Jak zostało opisane w sekcji 1.7.2 najpierw skorzystano z krzywej splajn odnośnie danych wejściowych co obrazuje rysunek 1.41. Jest to wykres RR dla pierwszego sygnału EKG użytego podczas wyliczania parametrów czasowych oraz częstotliwościowych.

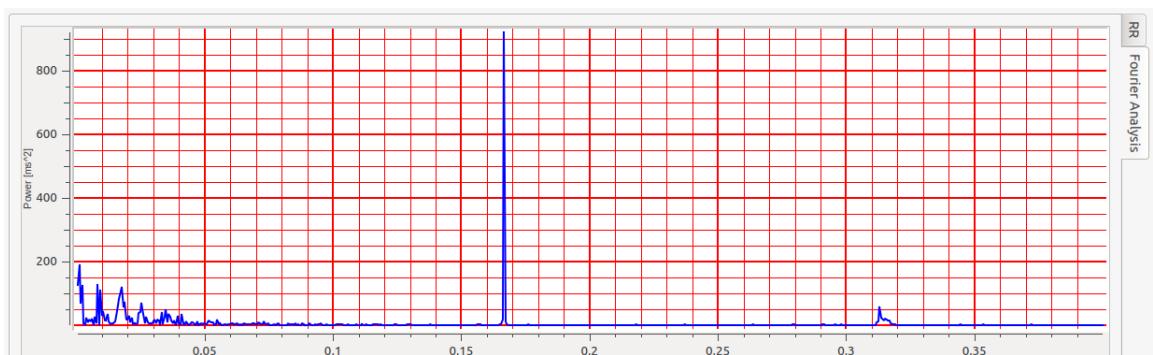


Rysunek 1.41: Wykres RR dla pierwszego pomiaru

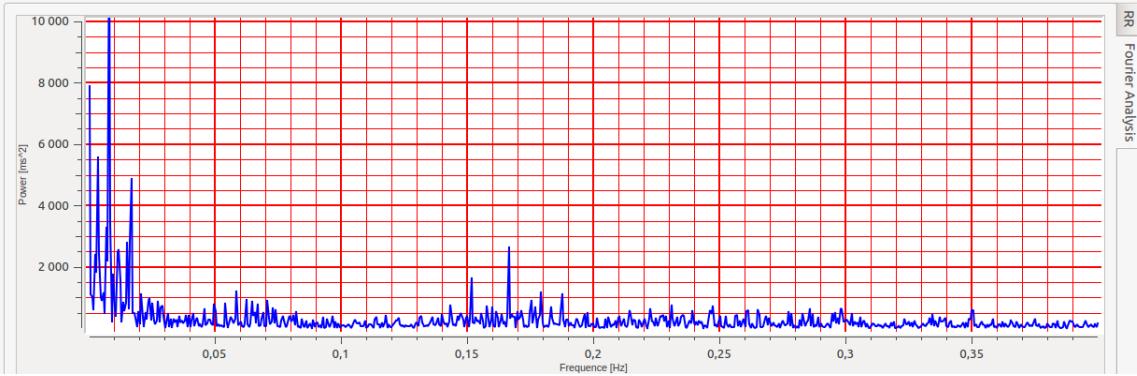
Następnie można przeprowadzić szybką transformatę Fouriera, która jeżeli wykres RR był w miarę okresowy powinna dać jeden wyróżniający się pik w środkowej części wykresu gęstości mocy widma. Na rysunkach 1.42, 1.43, 1.44, 1.45, 1.46 widoczne są kolejno przeprowadzone transformaty Fouriera na sygnałach EKG kolejnych 5 pomiarów. Cały czas korzystano z tych samych pomiarów co przy wyliczaniu parametrów czasowych i częstotliwościowych.



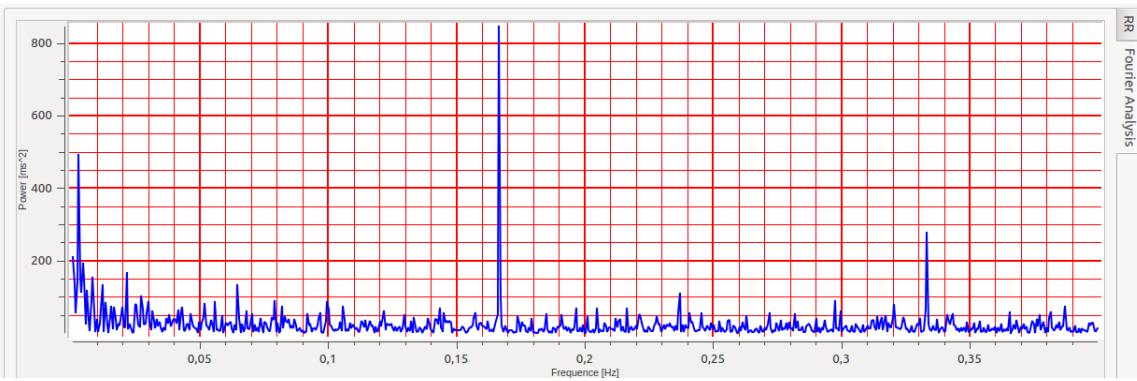
Rysunek 1.42: Wykres transformaty Fouriera dla pierwszego pomiaru



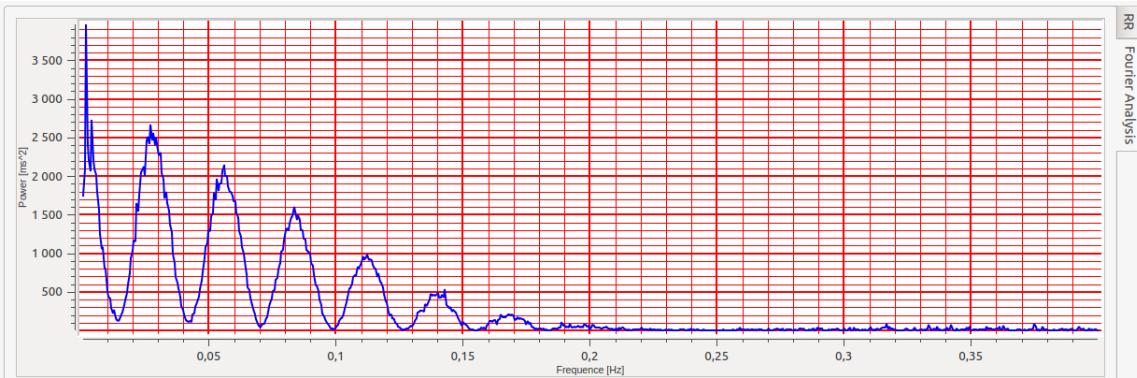
Rysunek 1.43: Wykres transformaty Fouriera dla drugiego pomiaru



Rysunek 1.44: Wykres transformaty Fouriera dla trzeciego pomiaru



Rysunek 1.45: Wykres transformaty Fouriera dla czwartego pomiaru



Rysunek 1.46: Wykres transformaty Fouriera dla piątego pomiaru

Po obejrzeniu wykresów transformaty Fouriera dla 5 różnych sygnałów EKG można dojść do wniosku, że najlepszy wynik, a co za tym idzie najlepsza praca serca występuje dla pacjentów z sygnałem EKG pierwszym oraz drugim. Są one najbardziej zbliżone do normalnej pracy serca zdrowego dorosłego człowieka. Gorsze wyniki uzyskujemy przy sygnale czwartym, natomiast pomiar trzeci oraz piąty świadczą o tym, że dane wejściowe dotyczyły człowieka z prawdopodobnymi zaburzeniami serca. Wykresy Fouriera w tych obu przypadkach nie pokazują

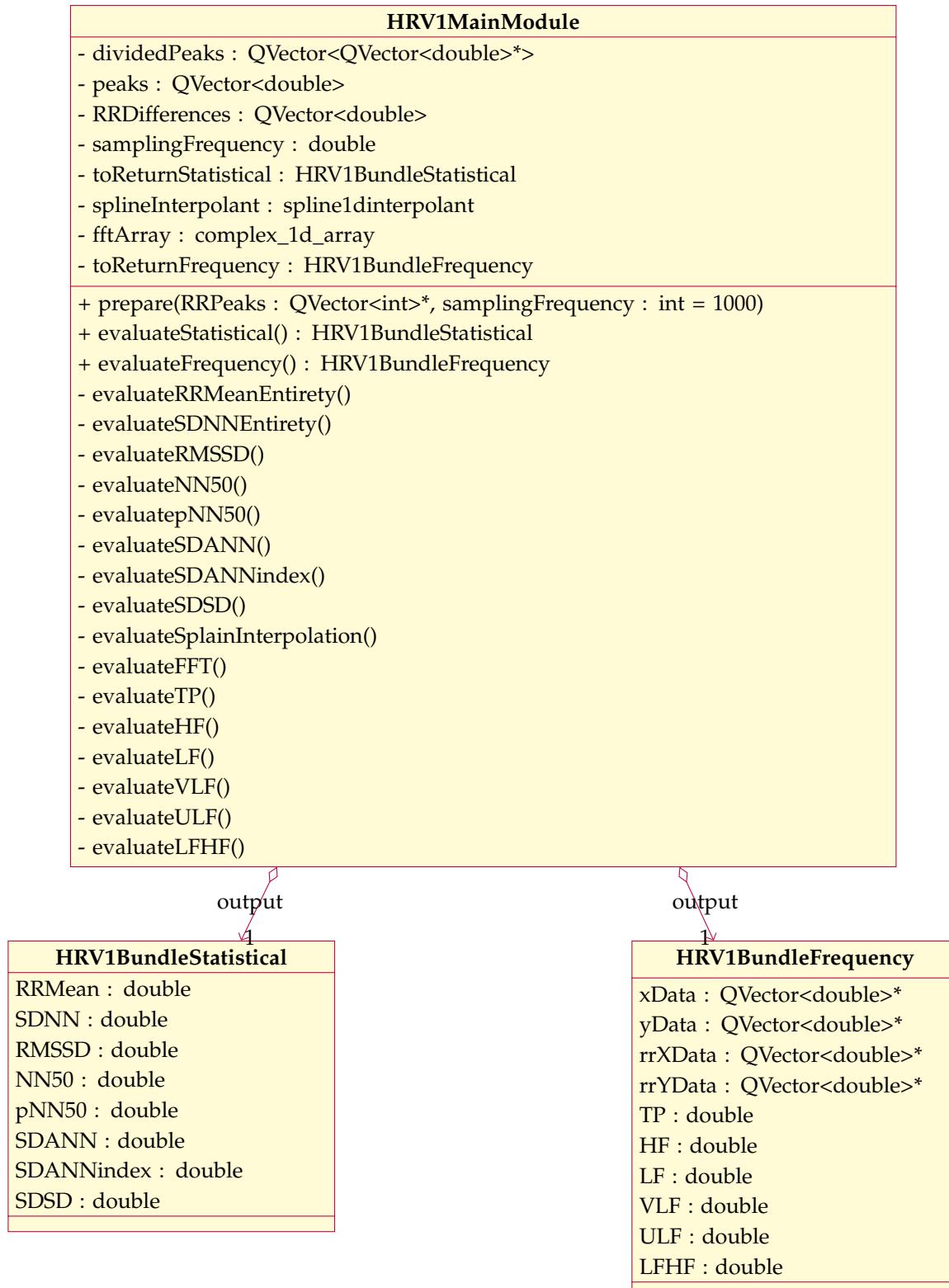
zują jednoznacznej częstotliwości oraz wartość mocy widma jest zdecydowanie za duża w obu przypadkach.

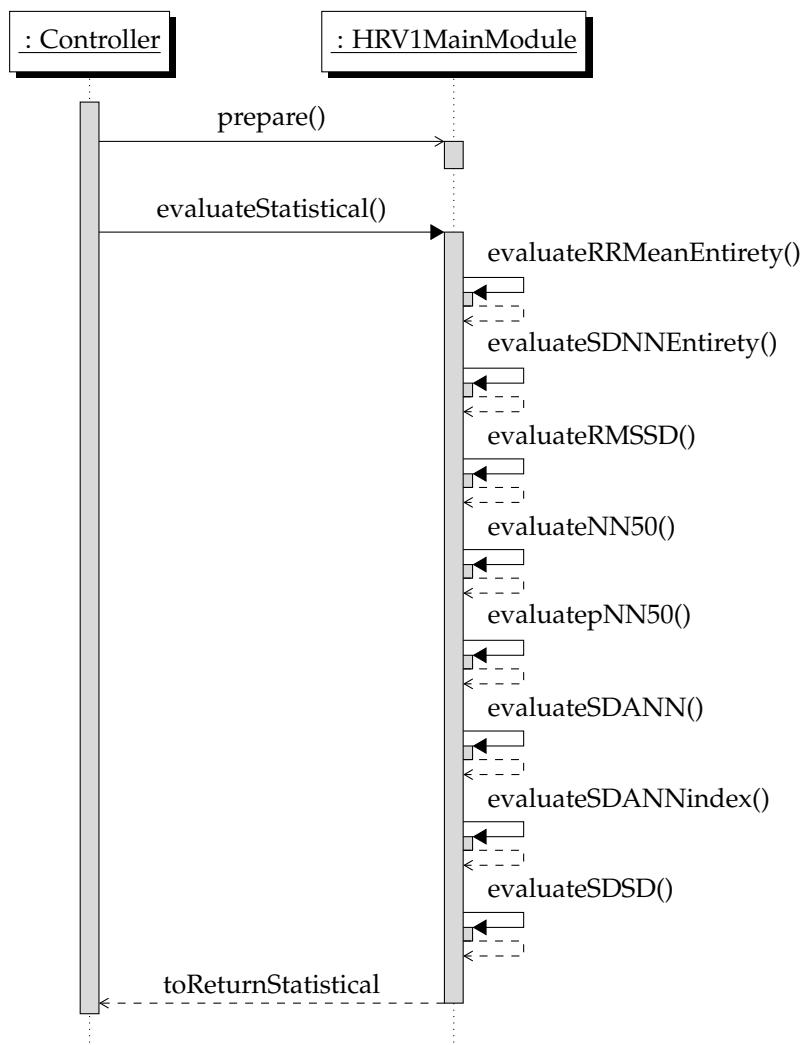
Moduł HRV1 według oceny autorów został poprawnie zinterpretowany i zaprogramowany. Świadczą o tym wyniki parametrów obu analiz oraz wykresy częstotliwościowe. Porównując wyniki modułu z przykładowymi wynikami zaczerpniętymi z internetu można znaleźć duże podobieństwo, szczególnie dla sygnałów EKG zdrowego, dorosłego człowieka.

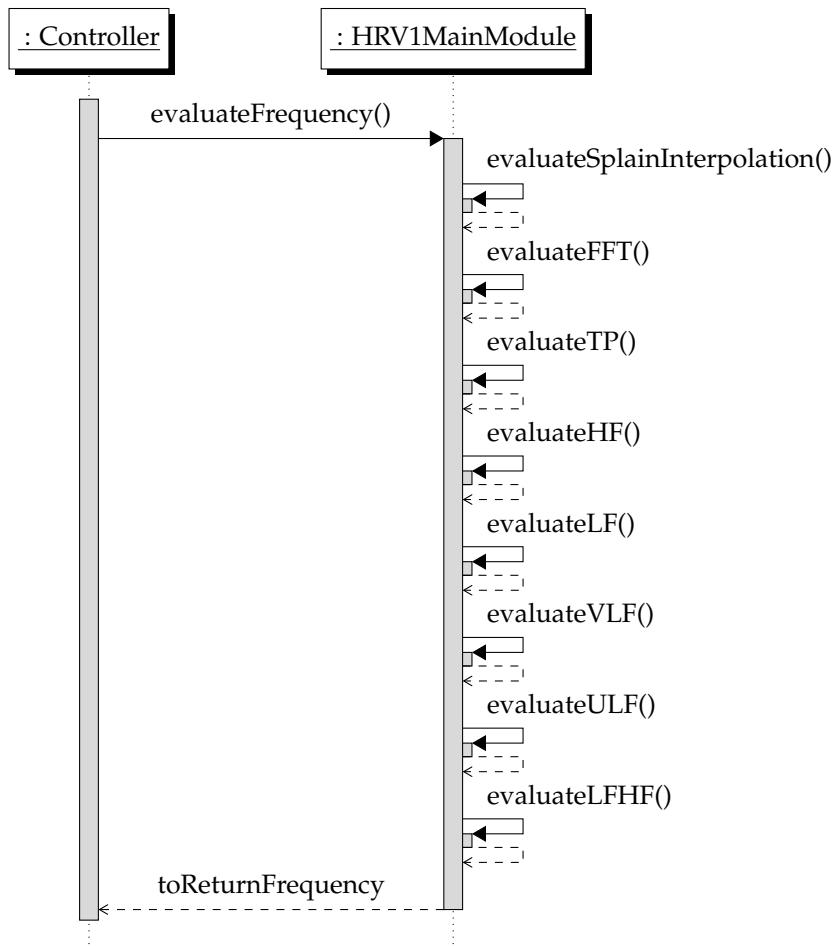
W ostatniej sekcji 1.7.4 został przedstawiony diagram klas oraz sekwencji programu zaprogramowanego na potrzeby modułu HRV1.

1.7.4 Diagramy klas i sekwencji

1.7.4.1 Diagram klas



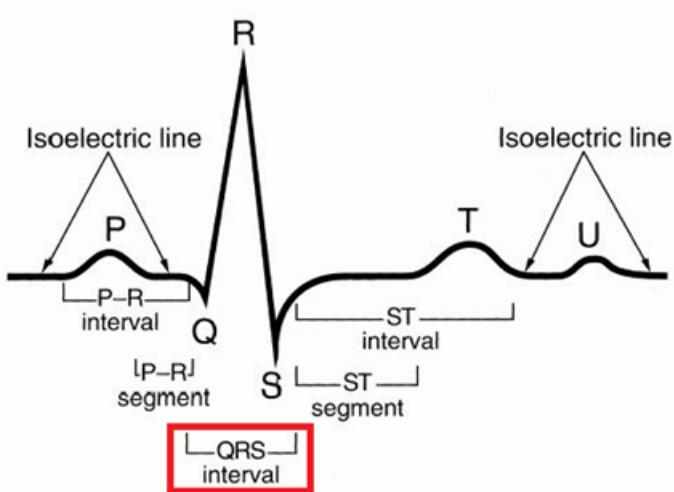
1.7.4.2 Diagram sekwencji



1.8 Moduł QRS_CLASS

1.8.1 Wstęp

Zadaniem niniejszego modułu jest klasyfikacja zespołów QRS, polegająca na wyodrębnieniu grup podobnych do siebie zespołów na podstawie przebiegu sygnału elektrokardiograficznego wraz z zaznaczonymi punktami charakterystycznymi.



Rysunek 1.47: Sygnał EKG wraz z zaznaczonym zespołem QRS (kolor czerwony). Źródło: [1]

Celem klasyfikacji zespołów QRS jest określenie ośrodków bodźcotwórczych w sercu. Odmienny kształt zespołu wynika z pobudzenia o innym przebiegu, którego źródło położone jest poza podstawowym generatorem rytmu. Grupowanie zespołów odbywa się zatem ze względu na ich morfologię (kształt) przy jednoczesnym określaniu typu pobudzenia (komorowe lub nadkomorowe). Koniecznie jest także wyodrębnienie zespołów, których pobudzenia nie można określić oraz artefaktów – przebiegówomyłkowo rozpoznanych jako zespół QRS.

1.8.2 Koncepcja proponowanego rozwiązania

Klasyfikacja zespołów QRS może być wykonywana na podstawie przebiegu sygnału w analizowanym przedziale lub z wykorzystaniem wektora cech reprezentujących dany zespół. Ze względu na szybkość obliczeń w opisywanym module wykorzystano drugą metodę, opisując zespół QRS za pomocą trzech współczynników kształtu:

- stosunek pola powierzchni do obwodu,

$$h_1 = 10 \cdot \frac{\sum_{i=0}^N |s(n)|}{\sum_{i=1}^N (|s(n) - s(n-1)|)}$$

- stosunek maksymalnej prędkości do maksymalnej amplitudy,

$$h_2 = 10 \cdot \frac{\max_{n=2..N} (|s(n) + s(n-2) - 2s(n-1)|)}{|\max_{n=0..N} (s(n)) - \min_{n=0..N} s(n)|}$$

- procentowy udział próbek, w których prędkość przekracza 40% prędkości maksymalnej, do długości zespołu QRS.

$$h_{10} = \frac{\sum k : \{|s(k) - s(k-1)| > 0,4 \cdot \max_{n=1..N} |s(n) - s(n-1)|\}}{N}$$

Grupowanie poszczególnych zespołów odbywa się z wykorzystaniem algorytmu G-means, będącego ulepszeniem klasycznego algorytmu k-średnich (k-means). Główną zaletą wybranego rozwiązania jest brak konieczności określania z góry liczby klas, do których przydziela się reprezentantów.

Podstawowe założenie algorytmu stanowi hipoteza, że zbiór reprezentantów każdej klasy posiada rozkład Gaussa. Metoda G-średnich uruchamiana jest z niewielką początkową liczbą centroid (klas), która może być równa 1 lub więcej, w zależności od posiadanej wiedzy na temat analizowanego problemu. Następnie wykonywany jest kolejno algorytm k-średnich dopóki wszystkie klasy będą posiadały rozkład normalny lub gdy osiągnięta zostanie maksymalna liczba klas.

Do prawidłowego działania algorytmu konieczne jest określenie, czy dane przyporządkowane do określonej klasy mają rozkład gaussowski. Rozróżniamy zatem dwie hipotezy:

- dane mają rozkład normalny – klasa jest wystarczająca do reprezentacji wszystkich przedstawicieli,
- dane nie mają rozkładu normalnego – klasa powinna być podzielona na dwie podklasy.

W celu sprawdzenia normalności rozkładu danych w określonej klasie, przeprowadzono test Andersona-Darlinga, opisany statystyką:

$$A^2(x) = -n - \frac{1}{n} \sum_{i=1}^n (2i-1)(\log(\Phi(y_i)) + \log(1-\Phi(y_{n+1-i})))$$

gdzie: n - liczba przedstawicieli danej klasy,

y_i - wartości reprezentantów przekształcone do rozkładu normalnego,
 $\Phi(x)$ - dystrybuanta rozkładu normalnego $N(0,1)$.

W przypadku, gdy wartości wariancji i odchylenia standardowego są szacowane, wykorzystuje się zmodyfikowaną statystykę:

$$A^*2(x) = A^2(x) \cdot \left(1 + \frac{4}{n} + \frac{25}{n^2}\right)$$

Rozkład uznaje się za normalny jeżeli wartość statystyki jest mniejsza od zadanego progu tolerancji α .

Detekcja i eliminacja artefaktów następuje w dwóch etapach:

- wyodrębnienie próbek o zerowej długości (punkty QRS_onset i QRS_end pokrywają się ze sobą),
- wykreślenie przedstawicieli klas, których odległość od danej centroidy jest większa niż dwukrotność wariancji.

1.8.3 Diagramy klas

RpeaksModule
a : static const double = 0.8
b(x, result)

1.8.4 Rezultaty i wnioski

1.8.4.1 Możliwości rozszerzenia

W celu ulepszenia modułu (poprawy klasyfikacji zespołów QRS) możliwe jest dodanie kolejnego algorytmu grupującego, który dawałby możliwość porównania otrzymanych wyników. Przykładowym rozwiązaniem jest zastosowanie metody Expectation Maximization – algorytmu iteracyjnego stosowanego do tworzenia modeli danych statystycznych, możliwego do wykorzystania także gdy istnieje ryzyko niekompletnych danych.

Wejścia metody Expectation Maximization (EM) stanowią: wektor danych (x), liczba klas (M), dopuszczalny błąd (e) oraz maksymalna liczba iteracji. Algorytm może być podzielony na dwa etapy: etapinicjalizacji i etap iteracyjny, który składa się z dwóch kroków: estymacji - expectation step (E-step) i maksymalizacji - maximization step (M-step), wykonywany iteracyjnie do czasu osiągnięcia zbieżności.

W kroku E-step szacuje się prawdopodobieństwo przynależności wektora do danej klasy, a następnie w kroku M-step ponownie określa się wektor parametrów rozkładu prawdopodobieństwa każdej klasy. Wykonanie algorytmu kończy się, gdy osiągnięto dopuszczalną wartość błędu lub po osiągnięciu maksymalnej liczby iteracji.

1.8.4.2 Wnioski

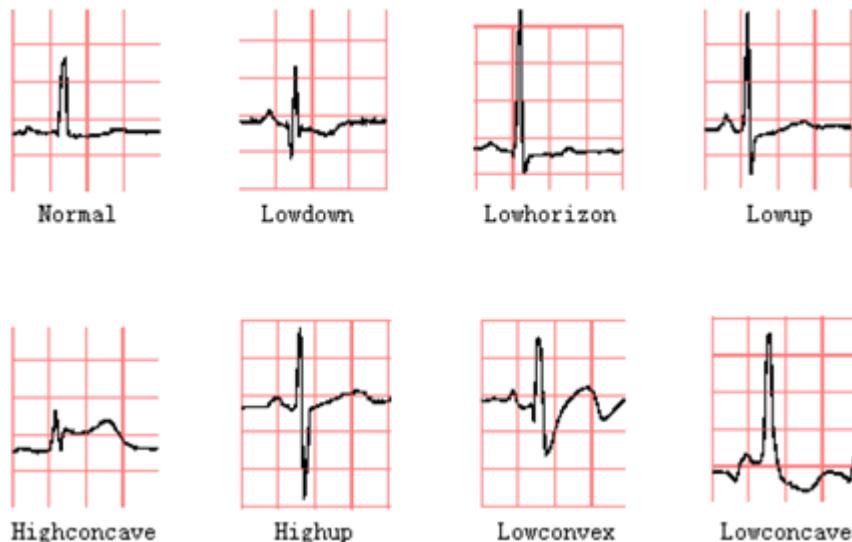
Klasyfikacja zespołów QRS stanowi ważny element detekcji i przetwarzania sygnału elektrokardiograficznego. Wykonanie projektu w tej dziedzinie pozwoliło na zapoznanie się z kilkoma zaawansowanymi metodami klasyfikacji wzorców i wykorzystanie narzędzi programistycznych, które znajdują zastosowanie nie tylko w szeroko pojętej medycynie.

Z powodu wysokiego stopnia skomplikowania zastosowanych metod, dużej ilości czasu przeznaczonej na synchronizację modułu z pozostałymi częściami aplikacji oraz licznych problemów podczas kompilacji kodu nie udało się z powodzeniem zaimplementować dodatkowej metody klasyfikacji. Do wykonanego modułu można jednak w prosty sposób dodać nowe algorytmy, które w przyszłości posłużą do jeszcze dokładniejszej analizy zespołów QRS.

1.9 Moduł ST_INTERVAL

1.9.1 Badania literaturowe

Zadaniem modułu ST_INTERVAL jest detekcja oraz analiza tzw. odcinków ST w zarejestrowanym sygnale EKG. Pojedynczy odcinek ST jest fragmentem sygnału ograniczonym przez punkt (oznaczany jako ST_{on}) znajdujący się w okolicach końca poprzedzającego go zespołu QRS (zwanego dalej punktem J) oraz punkt początkowy fali T (oznaczany jako T_{on}). Analiza kształtu tego obszaru oraz jego położenia względem linii izoelektrycznej elektrokardiogramu pozwala na stwierdzenie wystąpienia u pacjenta schorzeń takich jak niedokrwienie lub zawał mięśnia sercowego. W czasie normalnej pracy serca, odcinek ST przyjmuje kształt linii prostej, równoległej do izolinii sygnału i oddalonej od niej o około 0.1 mV. Podczas występowania zaburzeń można zaobserwować zmiany położenia odcinka względem izolinii (tzw. depresje i elewacje) oraz jego kształtu (linie proste o zmiennym nachyleniu, krzywe wklęsłe i wypukłe).



Rysunek 1.48: Rodzaje kształtu odcinka ST [30].

Postawiony problem można zatem zdekomponować na następujące zagadnienia:

1. Detekcja odcinków ST w sygnale EKG:
 - (a) wyznaczenie punktu T_{on} ,
 - (b) wyznaczenie punktu ST_{on} .
2. Analiza kształtu odcinków ST:
 - (a) położenie odcinka względem linii izoelektrycznej,
 - (b) klasyfikacja kształtu.
3. Wykrywanie i ocena epizodów związanych z odcinkami ST wykorzystanie danych o kształcie fragmentów w celu wykrycia potencjalnych zaburzeń.

Najważniejszym zadaniem w pierwszym podproblemie jest poprawne wyznaczenie położenia początku fali T. W dostępnej literaturze można najczęściej spotkać się z algorytmami detekcji

bazującymi na transformacji falkowej, wykorzystującymi fakt, iż punkt T_{on} w większości przypadków jest lokalnym ekstremum sygnału EKG. W opracowaniu [30] zaproponowano analizę zmienności znaku parametrów falkowej dekompozycji sygnału. Odmienne rozwiązanie zostało przedstawione w [16]. Algorytm ten wykorzystuje proste poszukiwanie ekstremów odpowiednio dobranego wskaźnika jakości, bazującego na morfologii (ksztalcie) fali T.

Wadą metod wykorzystujących falki jest zwiększoną złożoność obliczeniową, a ponadto trudność w doborze parametrów transformacji (tj. wyboru falki-matki, rzędu dekompozycji), w taki sposób, by uwzględniona została każda morfologia fali T. Algorytm bazujący na optymalizacji wskaźnika jakości wykorzystuje znacznie prostsze metody obliczeniowe, dodatkowo w jego konstrukcji uwzględniona została informacja o kształcie fali T, co pozwala na dokładniejszą detekcję pożdanego punktu. Założeniem rozważanych metod jest operowanie na zaszumionym sygnale EKG, stąd w każdej z nich znajduje się pewna forma filtracji (poprzez średnią kroczącą lub w wyniku transformacji falkowej).

Intuicyjna metoda klasyfikacji kształtu odcinka ST opisana została w pracy [19]. Opiera się ona na aproksymacji analizowanego fragmentu dwoma prostymi, umożliwiając jednoczesne wyznaczenie położenia oraz morfologii ST.

1.9.2 Koncepcja proponowanego rozwiązania

Wyznaczenie punktu T_{on} odbywa się poprzez analizę odcinków znajdujących się pomiędzy punktami J oraz T_{end} (koniec fali T). Odpowiedni dobór przedziału poszukiwania ogranicza wpływ morfologii pozostałych zespołów (QRS, P) w sygnale EKG.

Punkt T_{on} można wyznaczyć poprzez poszukiwanie ekstremów wskaźnika jakości A_k , opierającego się na morfologii fali T [16]. Formuła ta wyznacza dla każdego rozważanego punktu w badanym obszarze wartość stanowiącą analogię pola powierzchni pod sygnałem EKG w oknie przesuwnym o szerokości w próbek, rozpoczynającym się badaną próbką k oraz ograniczonego od dołu wartością tej próbki, co ukazane jest na rysunku 1.49.

$$A_k = \sum_{j=k}^{k+w-1} (s_j - \bar{s}_k) \quad (1.18)$$

Gdzie:

$k \in [k_a, k_b]$, $k_a = Q_{end}$, $k_b = T_{end}$ - przyjęty obszar detekcji ST,

s_i - wartość sygnału EKG w i-tej próbce,

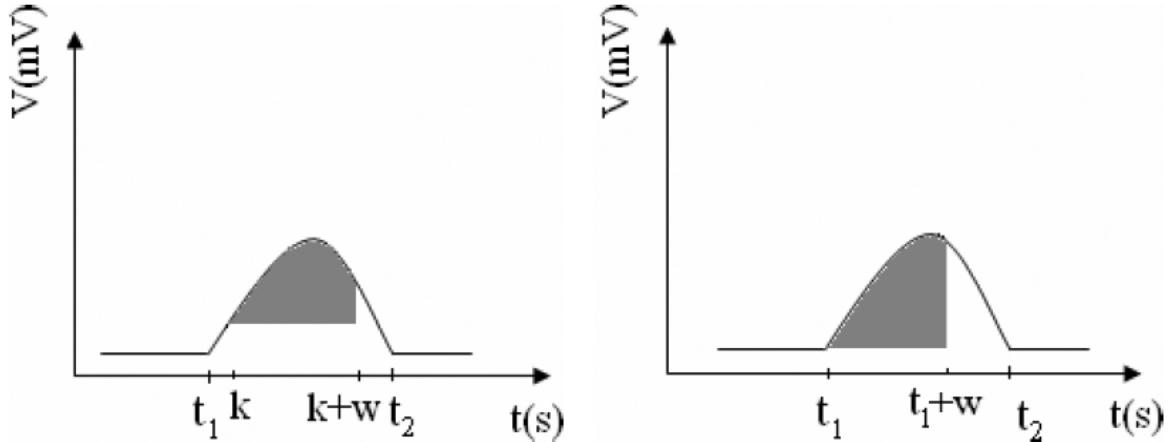
w - szerokość okna analizy.

Wskaźnik wprowadza także prostą filtrację sygnału (wartość \bar{s}_k), poprzez zastosowanie średniej w ruchomym oknie o szerokości $2p$:

$$\bar{s}_k = \frac{1}{2p+1} \sum_{j=k-p}^{k+p} s_j \quad (1.19)$$

Gdzie:

p - połowa szerokości okna uśredniającego.

Rysunek 1.49: Graficzna interpretacja wskaźnika A_k [16].

W przypadku analizy nominalnego obszaru zawierającego odcinek ST (linia prosta równoległa do izolinii) oraz falę T (krzywa wypukła), ekstrema wskaźnika A_k znajdują się w punkcie początkowym fali T (maksimum) oraz jego szczytce (minimum). Wyznaczenie obydwu punktów ekstremalnych k' oraz k'' jest konieczne w celu uwzględnienia innych morfologii tej fali (wkłesłości, dwubieguności – zob. rys. 1.50) [16].

$$k' = \arg \max_{k_a \leq k \leq k_b} A_k \quad (1.20a)$$

$$k'' = \arg \min_{k_a \leq k \leq k_b} A_k \quad (1.20b)$$

Wybór formuły na wyznaczenie punktu T_{on} zdeterminowany jest przez parametr λ stanowiący wartość progową „przełączającą” algorytm na odpowiednie morfologie fali T. Jeśli spełniony jest warunek:

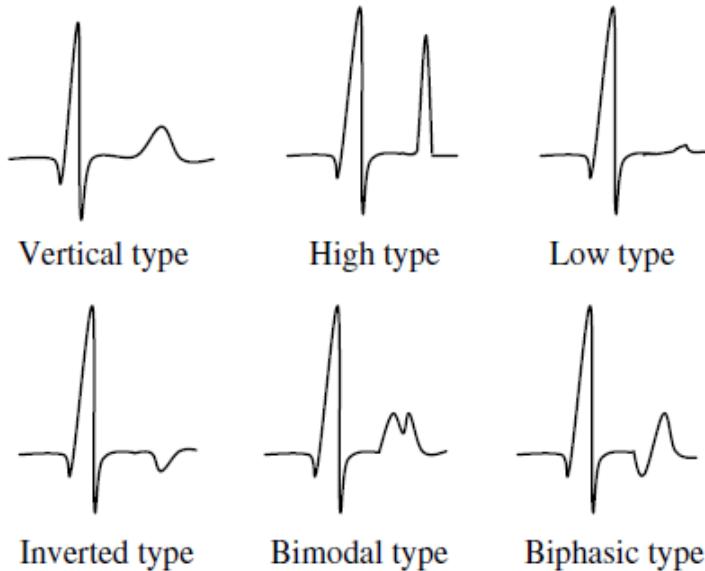
$$\frac{1}{\lambda} < \frac{|A_{k'}|}{|A_{k''}|} < \lambda \quad (1.21)$$

to:

$$T_{on} = \min \{k', k''\} \quad (1.22)$$

w przeciwnym wypadku:

$$T_{on} = \arg \max_{k \in \{k', k''\}} |A_k| \quad (1.23)$$



Rysunek 1.50: Morfologie fali T [16].

Aby uwzględnić sygnały EKG, w których występuje zjawisko dominacji fali T utrudniające poprawną detekcję krótkich odcinków ST, można wprowadzić następującą modyfikację wskaźnika A_k :

$$A_k = \sum_{j=k}^{k+w-1} (s_j - \bar{s}_k)^2 \quad (1.24)$$

Wprowadzenie kwadratu do wskaźnika pozwala na zwiększenie wpływu próbek sygnału znacznie oddalonych od linii izoelektrycznej, poprawiając jakość detekcji dla specyficznych przypadków.

Pozycja punktu początkowego fali ST uzależniona jest od szybkości pracy serca [16]. W celu poprawnego wykrycia punktu ST_{on} konieczne jest obliczenie interwału RT oraz oszacowanie tężna HR osoby badanej. Interwał RT wyznaczany jest jako odległość między punktami T_{on} i R_{peak} :

$$RT = T_{on} - R_{peak} \quad (1.25)$$

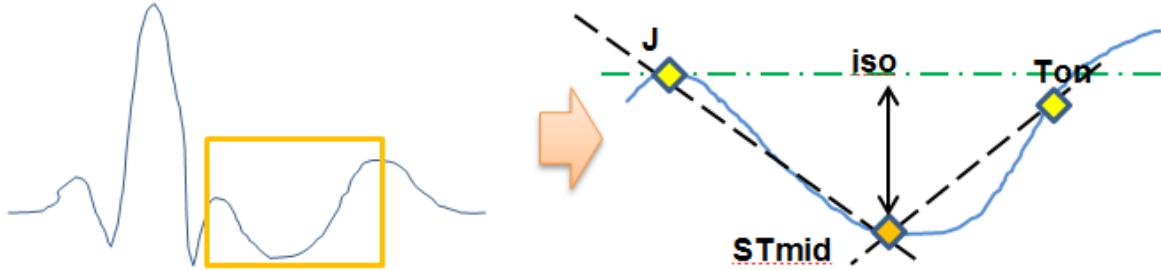
Tętno można określić na podstawie odległości, w jakiej znajdują się sąsiednie punkty R_{peak} . Posiadając te dane można wyznaczyć punkt ST_{on} :

$$ST_{on} = R_{peak} + xRT \quad (1.26)$$

Gdzie:

$$x = \begin{cases} 0.4 & \text{dla } HR < 100 \\ 0.45 & \text{dla } 100 \leq HR < 110 \\ 0.5 & \text{dla } 110 \leq HR < 120 \\ 0.55 & \text{dla } HR \geq 120 \end{cases} \quad (1.27)$$

Na podstawie położenia punktów J oraz T_{on} można wyznaczyć punkt środkowy ST_{mid} . Odległość tego punktu od linii izoelektrycznej określa odległość a_{ST} całego odcinka od tej izolinii, która następnie posłuży do oceny położenia fragmentu ST (normalna, depresja, elewacja).



Rysunek 1.51: Aproksymacja odcinka ST prostymi.

Odcinek ST może być aproksymowany dwoma prostymi łączącymi kolejno punkty J i ST_{mid} oraz ST_{mid} i T_{on} (rys. 1.51). Kąty nachylenia tych prostych względem izolinii (oznaczane jako d_1 i d_2) pozwalają na klasyfikację kształtu odcinka [19].

Wprowadzić można zatem dwa parametry progowe determinujące kształt interwału ST: a_{tol} (odległość od izolinii), d_{tol} (kąt nachylenia). Klasyfikacja kształtu zależna od tych parametrów przedstawiona została w tabeli 1.9. Typowymi wartościami zgodnymi z normami kardiologicznymi [19] są:

$$a_{tol} = 0.1mV, d_{tol} = 35^\circ$$

Tablica 1.9: Klasyfikacja kształtu odcinka ST.

Położenie	Kształt	a_{st}	d_1	d_2
nominalne	-	$ a_{st} \leq a_{tol}$	-	-
elewacja	-	$a_{st} > a_{tol}$	-	-
depresja	-	$a_{st} < -a_{tol}$	-	-
-	horyzontalny	-	$ d_1 \leq d_{tol}$	$ d_2 \leq d_{tol}$
-	wzrastający	-	$d_1 > d_{tol}$ $ d_1 \leq d_{tol}$ $d_1 > d_{tol}$	$d_2 > d_{tol}$ $d_2 > d_{tol}$ $ d_2 \leq d_{tol}$
-	opadający	-	$d_1 < -d_{tol}$ $ d_1 \leq d_{tol}$ $d_1 < -d_{tol}$	$d_2 < -d_{tol}$ $d_2 < -d_{tol}$ $ d_2 \leq d_{tol}$
-	wklęsły	-	$d_1 < -d_{tol}$	$d_2 > d_{tol}$
-	wypukły	-	$d_1 > d_{tol}$	$d_2 < -d_{tol}$

1.9.3 Rezultaty i wnioski

1.9.3.1 Rezultaty testów algorytmu

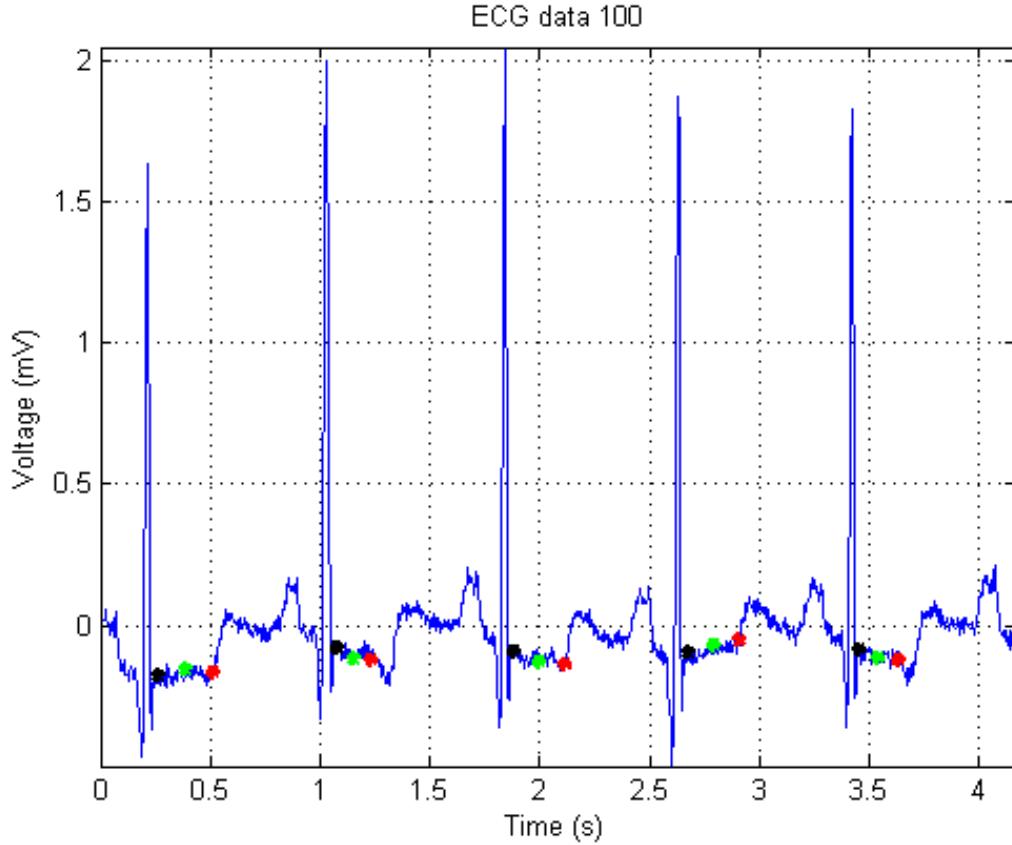
Zaimplementowany algorytm w środowisku MATLAB został wielokrotnie przetestowany z różnymi zestawami danych z bazy MIT-BIH oraz European ST-T. Detekcję i analizę odcinków ST przeprowadzono dla pięciu pierwszych uderzeń serca zarejestrowanych na elektrokardiogramie. Wyniki eksperymentów zostały zamieszczone w niniejszym rozdziale wraz z ich interpretacją. Położenie odcinka ST względem linii izoelektrycznej zostało oznaczone w tabelach jako a_{st} , natomiast nachylenia pierwszej i drugiej połowy jako odpowiednio d_1, d_2 . Wszystkie

testy zostały przeprowadzone przy wykorzystaniu niezmodyfikowanego wskaźnika detekcji (poza przypadkiem zestawu e1304) dla następującego zestawu parametrów:

$$p = 4, w = 30, \lambda = 6, a_{tol} = 0.15mV, d_{tol} = 35$$

Zestaw danych „100”

Zestaw ten jest wzorcowym przypadkiem, nie zawierającym jakichkolwiek zaburzeń pracy serca.



Rysunek 1.52: Rezultaty detekcji dla zestawu „100”.

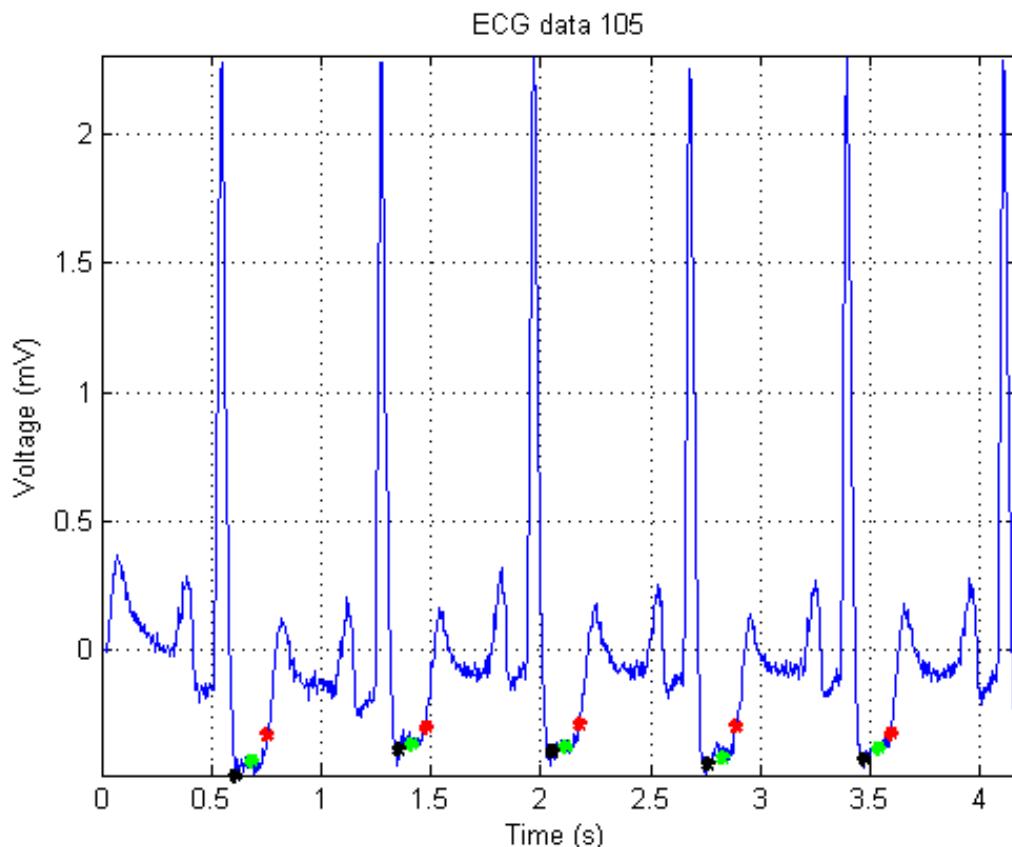
Wszystkie charakterystyczne punkty odcinków ST zostały prawidłowo wykryte. Klasyfikacja kształtu i położenia interwałów ST jest zgodna z oczekiwaniami (tabela 1.10).

Tablica 1.10: Rezultaty analizy dla zestawu „100”.

Lp	Typ oczekiwany	Typ wykryty	$a_{st}[mV]$	$d_1[^\circ]$	$d_2[^\circ]$
1	Depresyjny horyzontalny	Depresyjny horyzontalny	-0.155	10.426	-5.301
2	Normalny horyzontalny	Normalny horyzontalny	-0.106	-12.113	-23.966
3	Normalny horyzontalny	Normalny horyzontalny	-0.126	-17.299	-5.941
4	Normalny horyzontalny	Normalny horyzontalny	-0.070	12.451	5.194
5	Normalny horyzontalny	Normalny horyzontalny	-0.099	-7.443	-14.902

Zestaw danych „105”

Zestaw 105 jest przypadkiem zawierającym znaczącą ilość depresyjnych i wzrastających odcinków ST, co sugerować może występowanie niedokrwienie mięśnia sercowego pacjenta.



Rysunek 1.53: Rezultaty detekcji dla zestawu „105”.

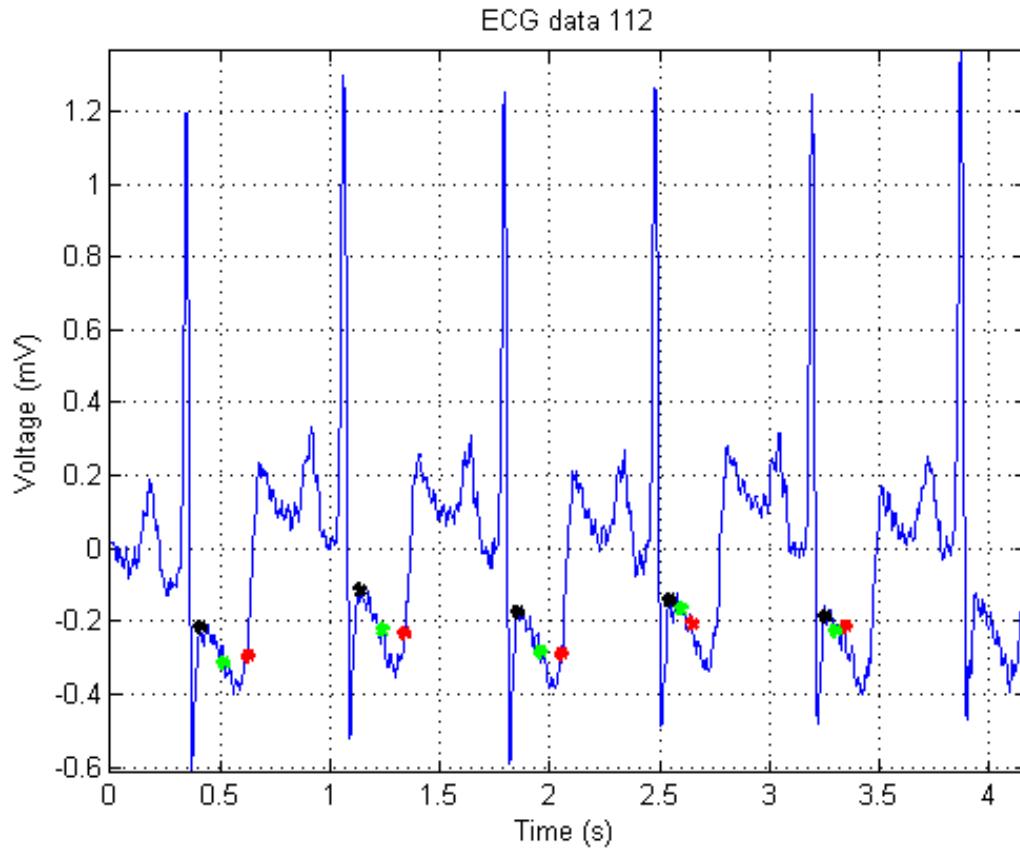
Tabela 1.11 ukazuje sukces detekcji i klasyfikacji w badanym przypadku.

Tablica 1.11: Rezultaty analizy dla zestawu „105”.

Lp	Typ oczekiwany	Typ wykryty	$a_{st} [mV]$	$d_1 [^\circ]$	$d_2 [^\circ]$
1	Depresyjny wzrastający	Depresyjny wzrastający	-0.429	37.821	53.347
2	Depresyjny wzrastający	Depresyjny wzrastający	-0.366	17.149	46.937
3	Depresyjny wzrastający	Depresyjny wzrastający	-0.398	1.416	59.307
4	Depresyjny wzrastający	Depresyjny wzrastający	-0.385	43.056	53.207
5	Depresyjny wzrastający	Depresyjny wzrastający	-0.372	46.728	44.352

Zestaw danych „112”

Zestaw ten charakteryzuje się występowaniem odcinków ST o morfologii depresyjnej opadającej.



Rysunek 1.54: Rezultaty detekcji dla zestawu „112”.

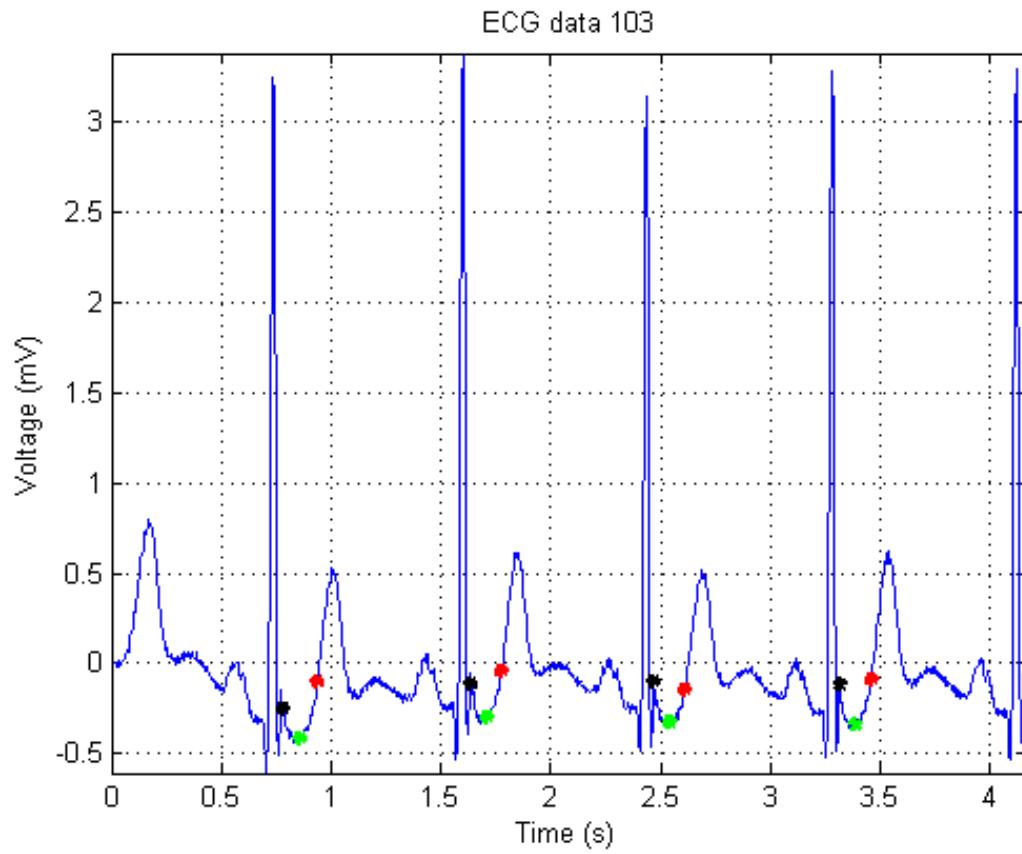
Detekcja i klasyfikacja zestawu 112 również zakończyła się pełnym sukcesem (tabela 1.12).

Tablica 1.12: Rezultaty analizy dla zestawu „112”.

Lp	Typ oczekiwany	Typ wykryty	$a_{st} [mV]$	$d_1 [^{\circ}]$	$d_2 [^{\circ}]$
1	Depresyjny opadający	Depresyjny opadający	-0.313	-42.806	19.021
2	Depresyjny opadający	Depresyjny opadający	-0.223	-47.684	-5.535
3	Depresyjny opadający	Depresyjny opadający	-0.282	-47.752	-17.172
4	Depresyjny opadający	Depresyjny opadający	-0.252	-40.834	24.248
5	Depresyjny opadający	Depresyjny opadający	-0.300	-45.349	0.460

Zestaw danych „103”

Przypadek został wykorzystany do zbadania skuteczności wykrywania odcinków o morfologii wkleślej.



Rysunek 1.55: Rezultaty detekcji dla zestawu „103”.

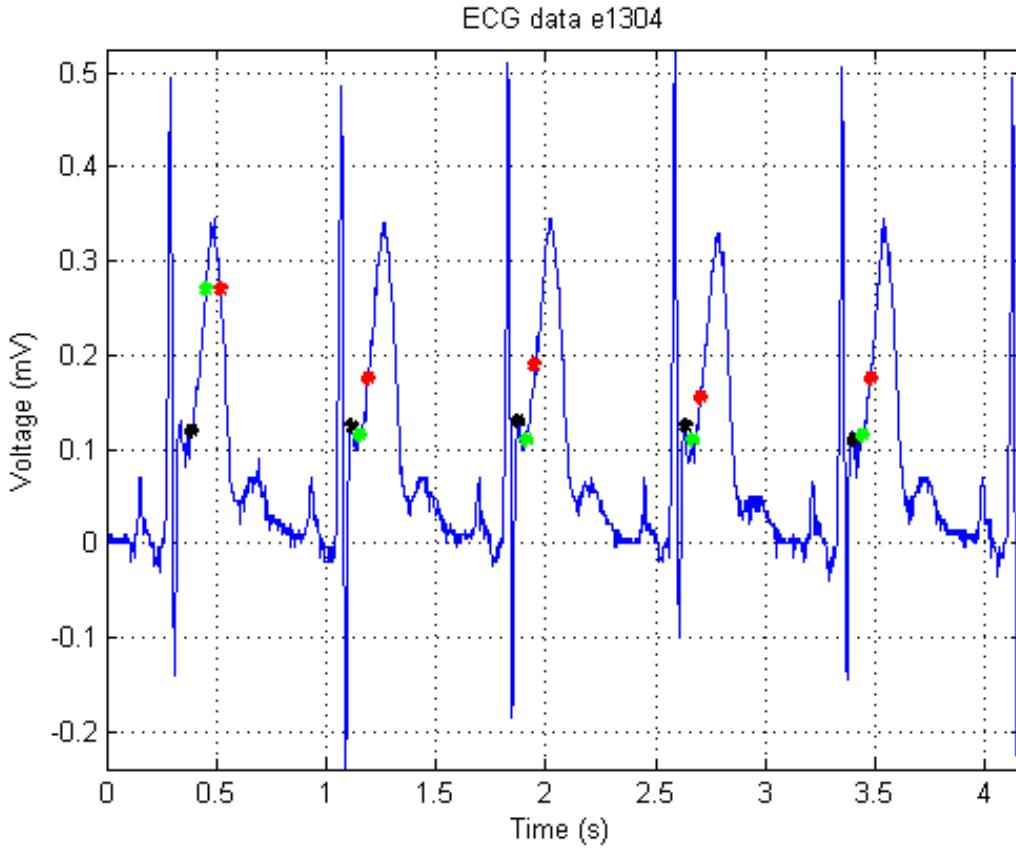
Podobnie jak w poprzednich przypadkach, odcinki wykryto i sklasyfikowano poprawnie (tabela 1.13).

Tablica 1.13: Rezultaty analizy dla zestawu „103”.

Lp	Typ oczekiwany	Typ wykryty	$a_{st}[mV]$	$d_1[^\circ]$	$d_2[^\circ]$
1	Depresyjny wklęsły	Depresyjny wklęsły	-0.408	-63.650	75.819
2	Depresyjny wklęsły	Depresyjny wklęsły	-0.293	-68.990	75.094
3	Depresyjny wklęsły	Depresyjny wklęsły	-0.323	-73.143	69.071
4	Depresyjny wklęsły	Depresyjny wklęsły	-0.335	-71.824	73.186

Zestaw danych „e1304”

Cechą charakterystyczną elektrokardiogramu *e1304* jest dominacja fali T przy większości uderzeń serca. Powoduje to znaczne zawężenie odcinków ST, utrudniając ich poprawną detekcję.



Rysunek 1.56: Rezultaty detekcji dla zestawu „e1304”.

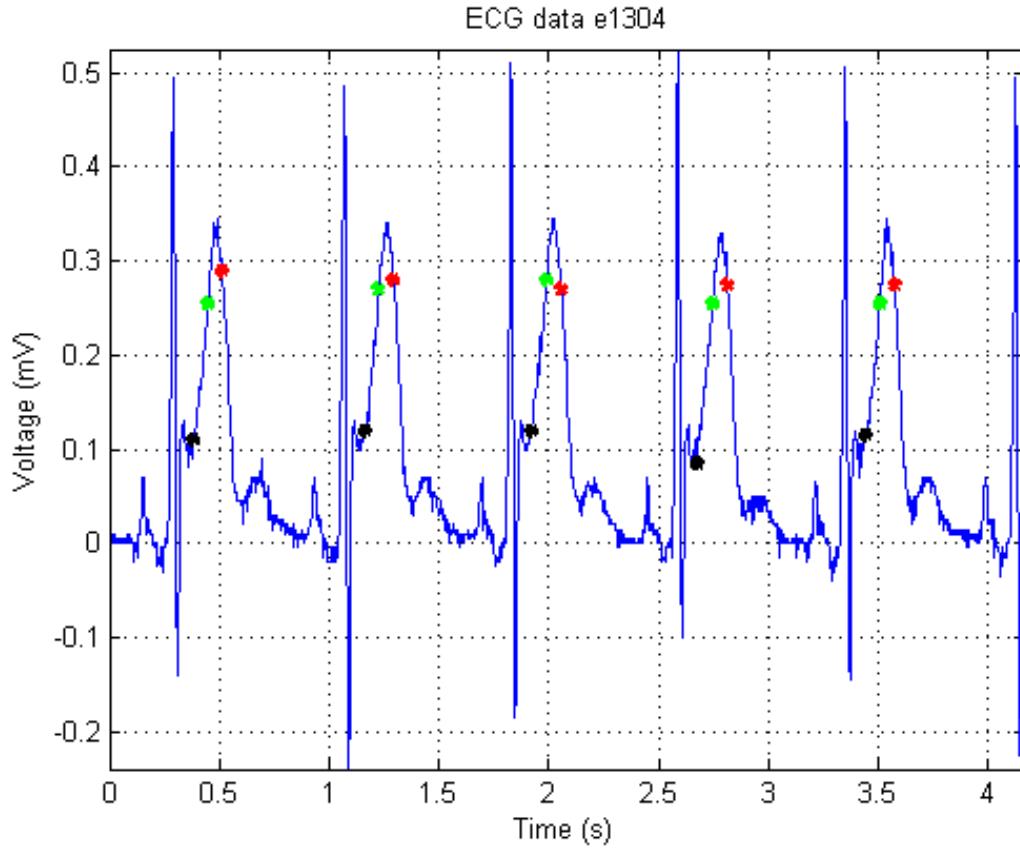
Jak ukazuje tabela 1.14, każdy z badanych odcinków ST został sklasyfikowany błędnie. Ponadto, większość z nich uznana została jako przypadek bez zaburzeń, co stwarza ryzyko niewykrycia poważnego schorzenia, które wyraźnie sugeruje zwiększoną amplitudę sygnału EKG (głównie fali T).

Tablica 1.14: Rezultaty analizy dla zestawu „e1304”.

Lp	Typ oczekiwany	Typ wykryty	$a_{st} [mV]$	$d_1 [^\circ]$	$d_2 [^\circ]$
1	Elewacyjny wypukły	Elewacyjny wzrastający	0.270	65.158	-0.000
2	Elewacyjny wypukły	Normalny wzrastający	0.115	-15.479	58.958
3	Elewacyjny wypukły	Normalny wzrastający	0.110	-28.980	65.706
4	Elewacyjny wypukły	Normalny wzrastający	0.110	-22.557	53.471
5	Elewacyjny wypukły	Normalny wzrastający	0.115	7.326	57.051

Zestaw danych „e1304” ze zmodyfikowanym algorytmem

W celu poprawy analizy badanego przypadku wykorzystana została wspomniana wcześniej modyfikacja algorytmu detekcji.



Rysunek 1.57: Rezultaty detekcji dla zestawu „e1304” po modyfikacji algorytmu.

Po wprowadzeniu modyfikacji otrzymany został częściowo poprawny wynik analizy (tabela 1.15). Położenie każdego z odcinków zostało prawidłowo rozpoznane jako elewacyjne, natomiast niewłaściwie określony został kształt. Mimo błędu, wyniki sugerują zaburzenie pracy serca w przeciwieństwie do rezultatów uzyskanych dla podstawowej wersji algorytmu.

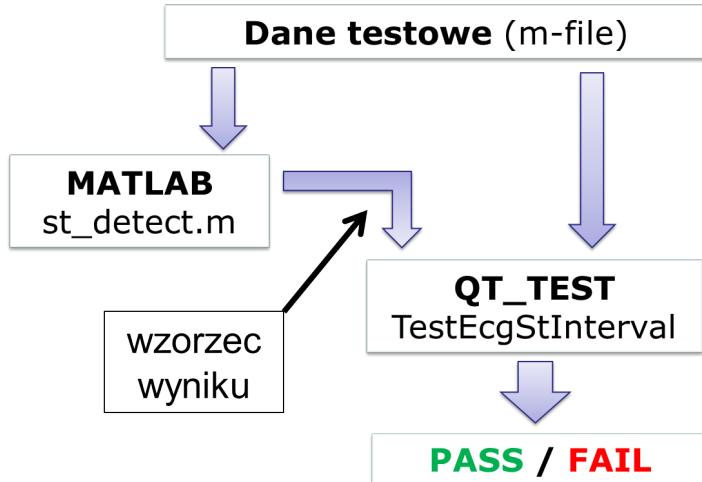
Tablica 1.15: Rezultaty analizy dla zestawu „e1304” po modyfikacji algorytmu.

Lp	Typ oczekiwany	Typ wykryty	$a_{st}[mV]$	$d_1[^\circ]$	$d_2[^\circ]$
1	Elewacyjny wypukły	Elewacyjny wzrastający	0.255	65.308	28.715
2	Elewacyjny wypukły	Elewacyjny wzrastający	0.270	66.038	8.531
3	Elewacyjny wypukły	Elewacyjny wzrastający	0.280	66.538	-8.531
4	Elewacyjny wypukły	Elewacyjny wzrastający	0.255	67.780	16.699
5	Elewacyjny wypukły	Elewacyjny wzrastający	0.255	64.537	16.699

1.9.3.2 Weryfikacja poprawności implementacji w C++

Na podstawie algorytmu zbudowanego w środowisku MATLAB utworzona została implementacja w języku C++ z wykorzystaniem bibliotek Qt. Ze względu na odmienne metodyki tworzenia programów w tych środowiskach, przydatna okazuje się automatyczna weryfikacja poprawności działania modułu. Do badania zgodności implementacji w C++ z algorytmem wzor-

cowym wykorzystano mechanizm testów automatycznych, które udostępnia środowisko Qt – QT_TEST. Pozwala on na utworzenie wyspecjalizowanego programu, którego zadaniem jest weryfikacja poprawnego funkcjonowania wybranej części projektu (tzw. *unit tests*).



Rysunek 1.58: Przepływ danych testowych w procedurze weryfikacji.

Utworzone testy automatyczne wykorzystują ten sam zestaw danych wejściowych, co implementacja w środowisku MATLAB, gdzie generowane są także wzorce poprawnych wyników analizy (rys. 1.58). Każdy z przypadków testowych przeprowadzany jest według poniższej procedury:

- weryfikacja liczby wyników analizy,
- weryfikacja klasyfikacji pozycji interwałów ST,
- weryfikacja klasyfikacji kształtów ST,
- weryfikacja punktów brzegowych i środkowych interwałów.

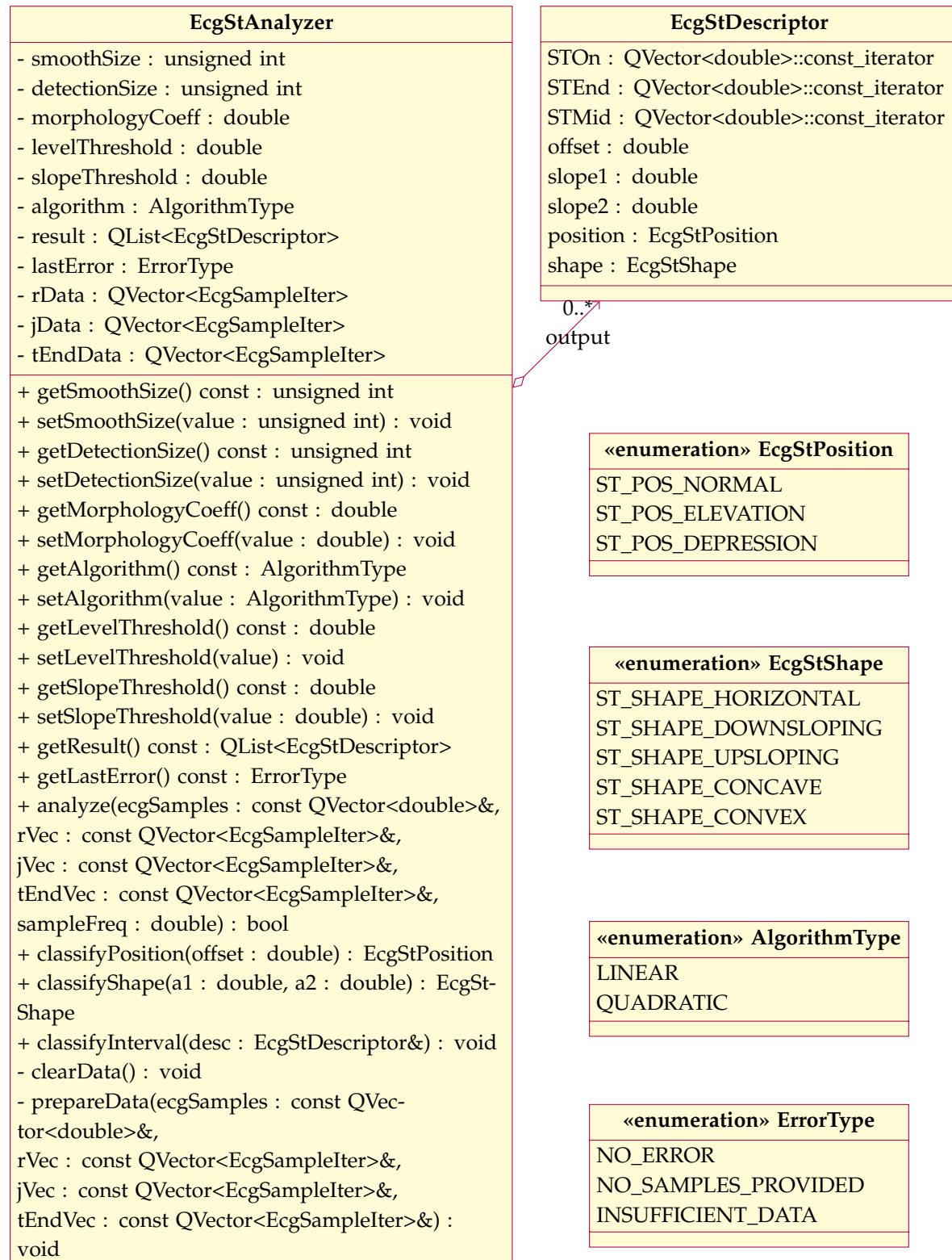
Badanie poprawności detekcji punktów brzegowych i środkowych przeprowadzane jest z zadaną tolerancją, gdyż m.in. ze względu na różnice w błędach obcięcia i zaokrąglenia między C++ a MATLAB pozycje tych punktów mogą się minimalnie przesunąć.

W ramach eksperymentów opisana procedura testowania została przeprowadzona dla czterech zestawów danych: 100, 103, 105, 112 (przedstawionych w poprzedniej sekcji). Implementacja w C++ okazała się zgodna z algorytmem wzorcowym. Wyniki testów ukazane są w tabeli 2.1.3.7. Kolejne kolumny tabeli przedstawiają procent zgodności każdego z etapów weryfikacji.

Tablica 1.16: Wyniki weryfikacji implementacji w C++.

Zestaw testowy	Zgodność pozycji	Zgodność kształtu	Zgodność punktów brzegowych
100	100%	100%	100%
103	100%	100%	90%
105	100%	100%	90%
112	100%	100%	100%

1.9.4 Diagramy klas



1.10 Moduł SLEEP_APNEA

1.10.1 Badania literaturowe

Bezdech senny, okresowe zaprzestanie oddychania podczas snu, jest często nieroznawalną wadą dróg oddechowych występującą u milionów ludzi na całym świecie, powodującą zwiększenie zachorowalności oraz śmiertelności. Za bezdech senny uważa się przerwy w wentylacji płuc dłuższe niż 10 sekund lub spłycenie oddechu poniżej 50%.

Obecna technologia wykrywania bezdechu sennego wymaga monitorowania pacjenta podczas snu w specjalnie wyposażonych laboratoriach. Ze względu na koszty i niedogodności związane z badaniem polisomnograficznym, dużo tańszym i łatwiejszym sposobem jest wdrażanie technik wykrywania dolegliwości za pomocą algorytmów specjalnych algorytmów. W ostatnich kilku latach, niektóre badania skupiały się na rozwijaniu automatycznych narzędzi do detekcji bezdechu sennego, na podstawie analizy sygnału EKG z różnymi technikami przetwarzania sygnału i rozpoznawania dolegliwości [1]. Niektóre badania stosowały funkcje wyodrębniania z sygnału EKG informacji jak HRV (zmienność rytmu zatokowego), amplitudy zespołu QRS, obszary pików R. Uzyskane wyniki pozwoliły stworzyć różne stopnie klasyfikacji między osobami zdrowymi, a cierpiącymi na bezdech senny. Spośród różnych metod klasyfikacji można wyróżnić transformację Fouriera, transformację Hilberta oraz analizę czasowo-częstotliwościową. Bardzo popularną metodą jest wykorzystywanie sieci neuronowych[2], klasyfikacja na podstawie sąsiedztw – algorytm KNN (algorytm k-najbliższych sąsiadów), klasyfikacja przy pomocy SVM (maszyna wektorów nośnych) oraz Naive Bayes(naiwny klasyfikator bayesowski)[3].

Algorytmy wykorzystujące sieci neuronowe oraz metody klasyfikacji opisane wyżej, cechują się dużym skomplikowaniem. By stworzyć poprawne kryterium klasyfikacji potrzebna jest duża ilość cech, które mogą wskazać czy badana osoba cierpi na bezdech senny czy nie. Implementacja tak złożonego i rozbudowanego algorytmu pochłonęłaby zbyt dużo czasu. Ze względów czasowych oraz technicznych zdecydowano się skorzystać z algorytmu opisanego w dokumencie [4]. Metoda ta cechuje się stosunkowo łatwą i zrozumiałą implementacją. Dodatkowo osiąga zadowalające efekty - twórcy publikacji uzyskali około 90% skuteczności w wykrywaniu dolegliwości. Autorzy metody zauważili, że bezdech senny zmienia dynamikę pracy serca. U ludzi chorych, w okresach długotrwałego bezdechu sennego, tępno zwykle pokazuje cykliczne wzrosły i spadki związane z fazą bezdechu i wznowienia oddychania. Cykle, które mają tendencje do drgań przy częstotliwości pomiędzy 0,01 i 0,04 Hz, są charakterystyczne dla osób cierpiących na bezdech senny i nie występują u zdrowych pacjentów.

1.10.2 Koncepcja proponowanego rozwiązania

Na podstawie algorytmu opisanego w dokumencie [4] powstała koncepcja rozwiązania problemu wykrycia bezdechu sennego:

1. Obliczenie interwałów RR

Danymi wejściowymi, które były podstawą rozważań, były piki typu R. By przejść do kolejnego etapu wyznaczania bezdechu sennego należało wyznaczyć tzw. interwały RR, a więc odległości pomiędzy kolejnymi pikami R.

2. Zastosowanie filtra uśredniającego interwały RR

Wyznaczony sygnał okazał się mocno zaszumiony. Dalsza praca na nieczytelnym, zaszumionym sygnale wiązałaby się ze stopniowym narastaniem błędów w każdym kolejnym kroku algorytmu. By uniknąć tego niepożądanego efektu zastosowany został filtr uśredniający.

3. Przepróbkowanie sygnału na częstotliwość 1Hz

Początkowo, uzyskany sygnał był próbkowany nierównomiernie. Uniemożliwiło to dalsze poprawne rozważania m.in. zadowalające wyznaczenie transformaty Hilberta, a w konsekwencji błędne działanie modułu. Dlatego sygnał został przepróbkowany na częstotliwość jednego herca, tak by próbki występowały w odstępach sekundowych.

4. Filtracja dolno i górnoprzepustowa

W celu eliminacji nagłych, nienaturalnie dużych skoków amplitudy oraz wstępnej normalizacji przepróbkowanego sygnału, dokonano filtracji dolno i górnoprzepustowej. Sygnał po wykonaniu wszystkich poprzednich kroków był gotowy do zastosowania transformaty Hilberta.

5. Zastosowanie transformaty Hilberta oraz wyznaczenie sygnału amplitudowego i częstotliwościowego

Sygnał poddany został transformacie Hilberta, którą dla funkcji $g(t)$ przedstawia równanie:

$$\hat{g}(t) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{g(\tau)}{t - \tau} d\tau \quad (1.28)$$

Jest to splot funkcji $g(t)$ z funkcją:

$$h(t) = \frac{1}{\pi t} \quad (1.29)$$

W każdym punkcie sygnału wyznaczane jest widmo amplitudowe i częstotliwościowe Hilberta, z których to bezpośrednio wyznaczone zostanie występowanie bezdechu sennego.

6. Filtracja uśredniająca oraz normalizacja sygnału amplitudowego

By uniknąć zaszumień i błędnych wyników ponownie dokonano filtracji uśredniającej. Natomiast amplituda została dodatkowo poddana operacji normalizacji. Operacja ta została zastosowana tak, by średnia wartość amplitudy w przedziale czasu równa była 1.

7. Obliczenie minimalnej amplitudy Hilberta oraz maksymalnej częstotliwości Hilberta

Obliczenie minimalnej amplitudy Hilberta, która była wskaźnikiem na podstawie którego stwierdzano występowanie dolegliwości wyliczono z wzoru:

$$ampl = a + \frac{b(mid + 1)}{2} \quad (1.30)$$

gdzie: $a=0.3$, $b=1.85$, mid - średkowa wartość między maksymalną a minimalną amplitudą.

Obliczenie minimalnej częstotliwości Hilberta, która również była wskaźnikiem na podstawie którego stwierdzano występowanie bezdechu sennego:

$$freq = 0.4mid \quad (1.31)$$

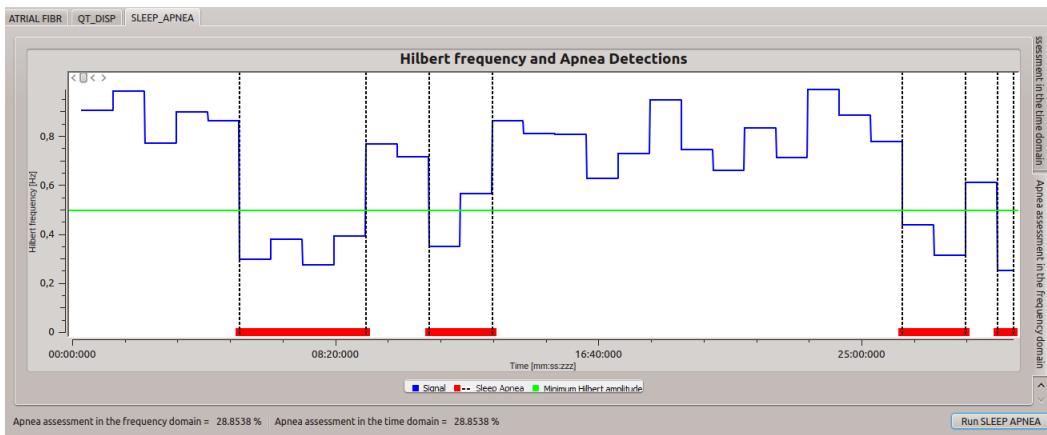
gdzie: mid - średkowa wartość między maksymalną a minimalną częstotliwością.

8. Oznaczenie próbek, w których wystąpił bezdech senny

Bezdech senny był wyznaczany na podstawie amplitudy i częstotliwości Hilberta. Przekroczenie przez amplitudę wyznaczonej w poprzednim punkcie, minimalnej amplitudy Hilberta oznaczało występowanie bezdechu sennego. Podobnie z częstotliwością. W miejscach gdzie spadała poniżej maksymalnej częstotliwości Hilberta występował bezdech senny.

1.10.3 Rezultaty i wyniki

Działanie programu oraz wizualizację danych wyjściowych przedstawiono na Rys. 1.59



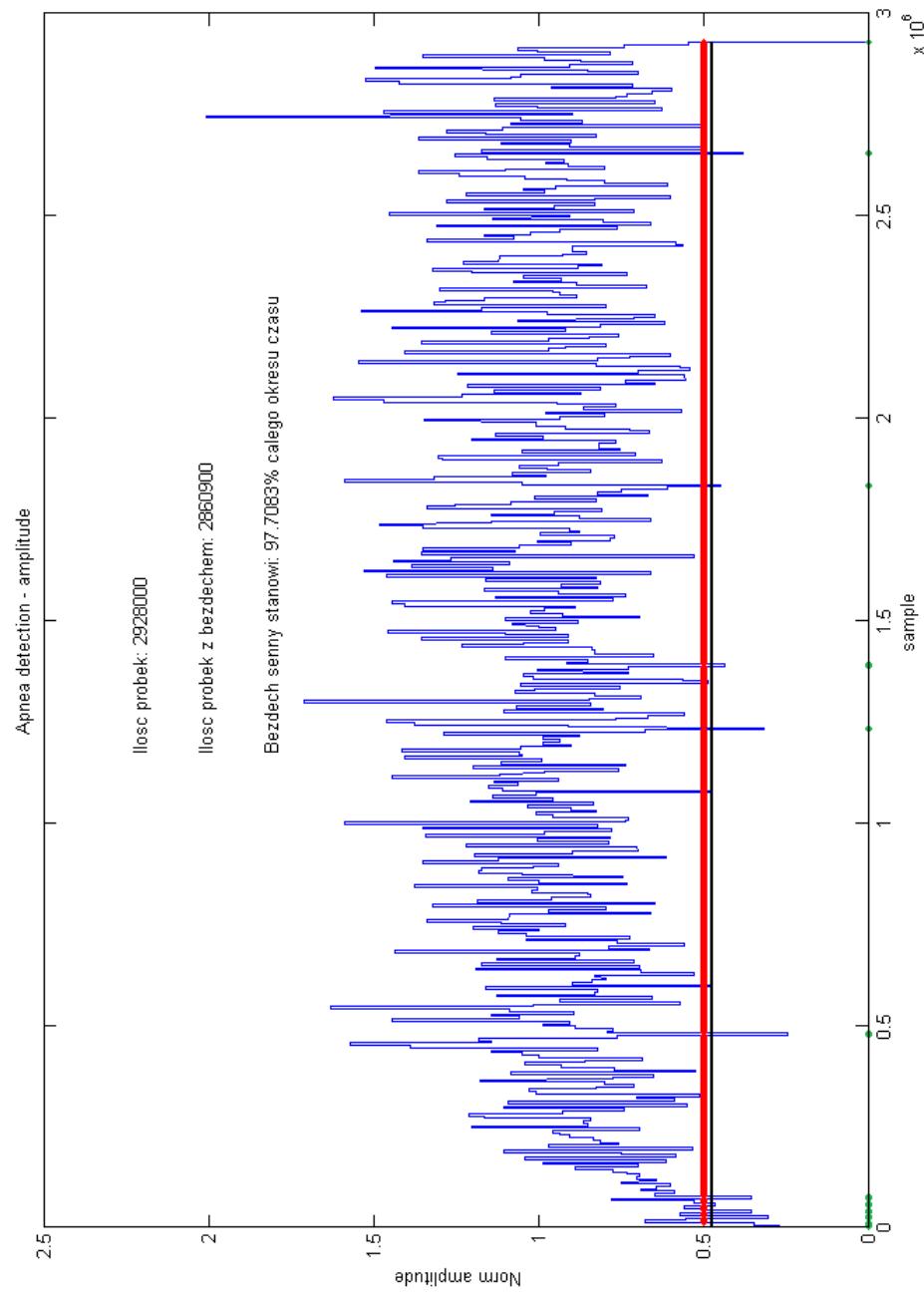
Rysunek 1.59: GUI aplikacji dla modułu SLEEP APNEA

Na wykresie powyżej linią niebieską zaznaczono znormalizowany sygnał amplitudy, linią zieloną wartość maksymalnej amplitudy hilberta oraz czerwoną linią - okres bezdechu sennego (czarne, pionowe linie oznaczają początek i koniec bezdechu). Pod krzywymi przedstawiono również ocenę bezdechu sennego w dziedzinie czasu i częstotliwości.

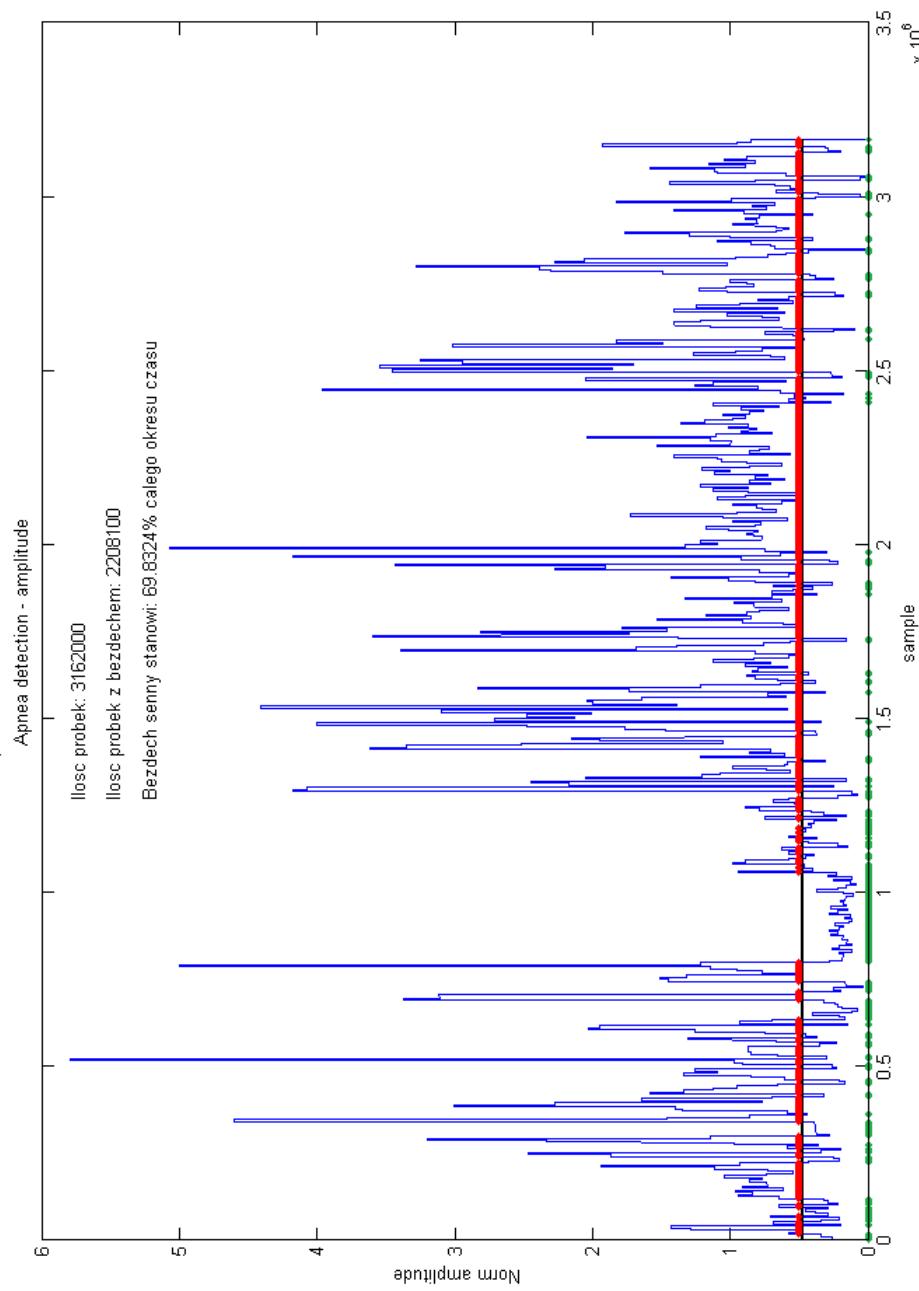
Program został również przetestowany dla pięciu próbek danych, pobranych z internetowej bazy danych Physionet. Algorytm analizował dane próbki z częstotliwością 100Hz. Na każdym z wykresów zostały przedstawione następujące elementy:

1. **Znormalizowana amplituda sygnału**
2. **Maksymalna amplituda Hilberta**
3. **Okres bezdechu sennego**
4. **Okres bezdechu sennego**
5. Podsumowanie danych wyjściowych

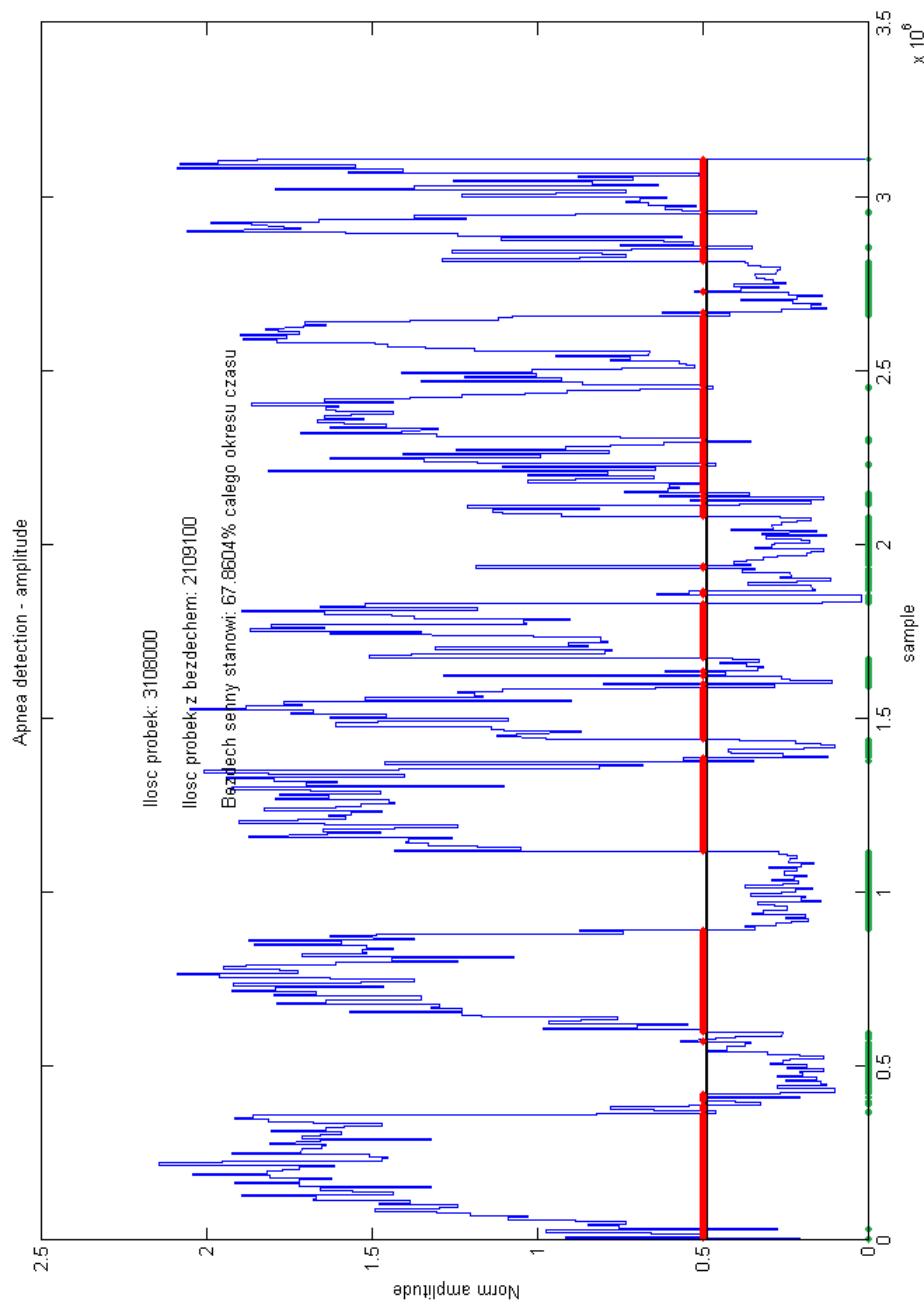
Poniżej zamieszczono wykresy dla pięciu zestawów danych.



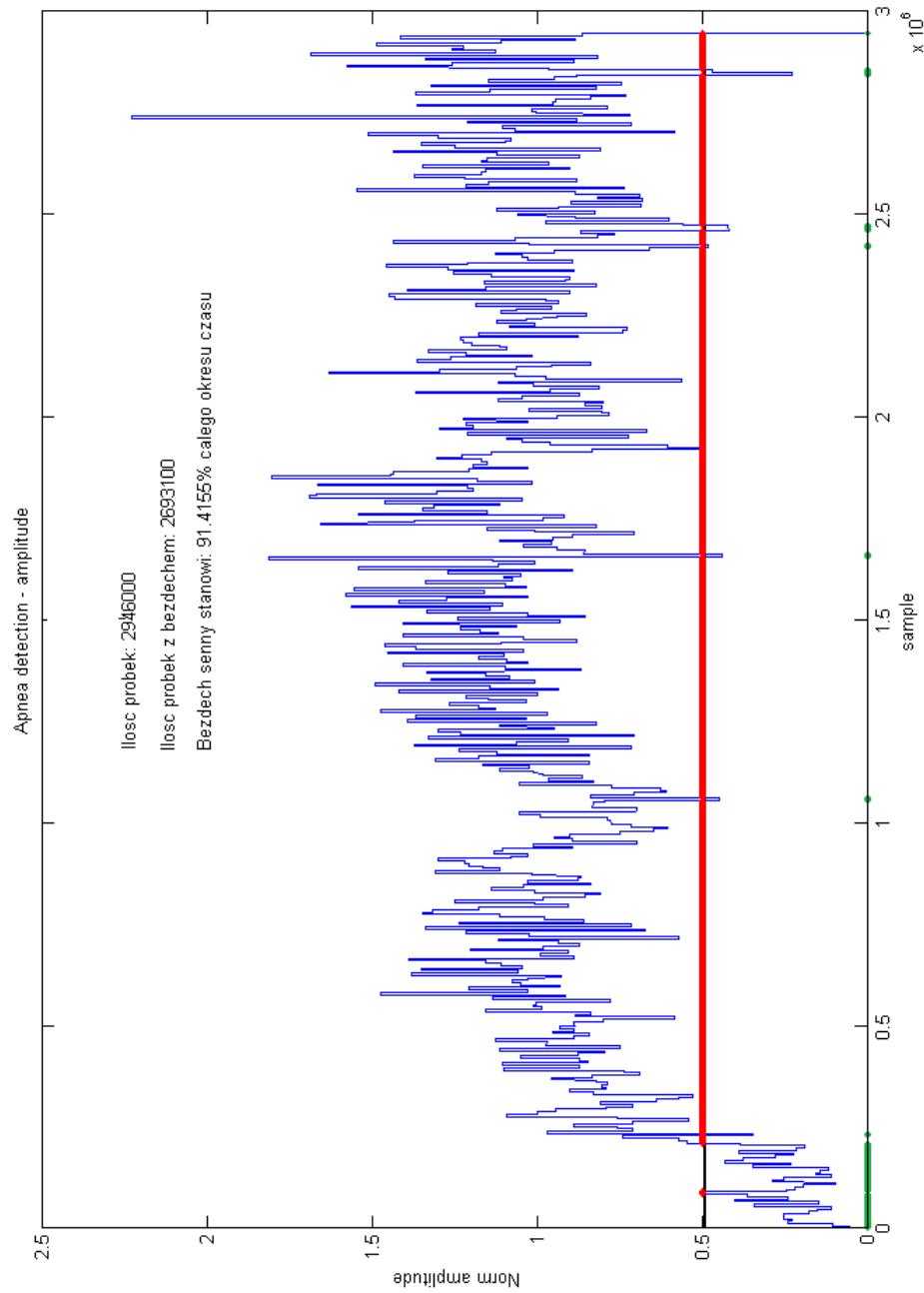
Rysunek 1.60: Wykres końcowy programu dla danych a01



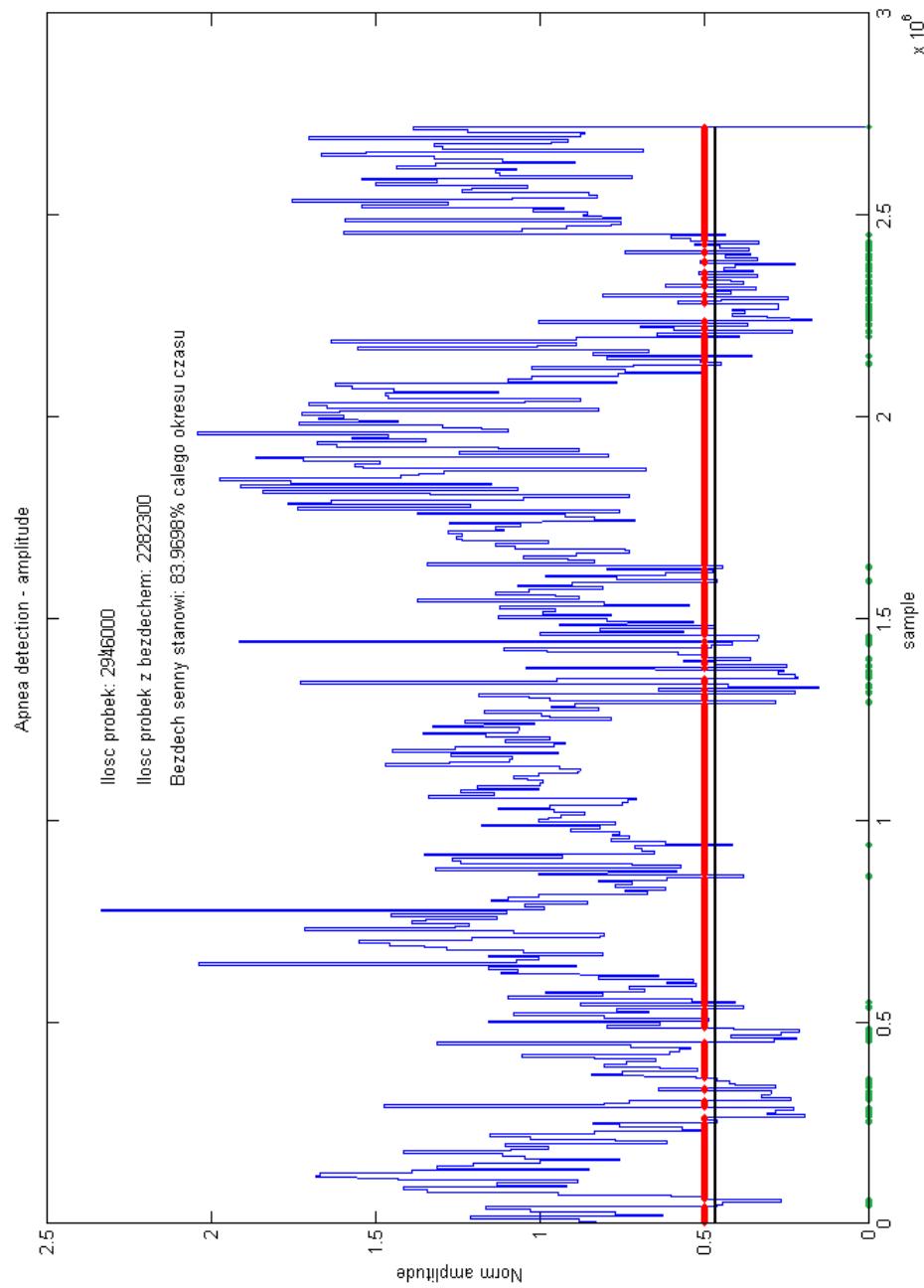
Rysunek 1.61: Wykres końcowy programu dla danych a02



Rysunek 1.62: Wykres końcowy programu dla danych a03



Rysunek 1.63: Wykres końcowy programu dla danych a04



Rysunek 1.64: Wykres końcowy programu dla danych a05

Poniżej przedstawiono tabelę porównawczą dla omawianych wyżej przykładów. Są w niej zawarte wyniki badań specjalistów z Physionetu oraz wyniki sporzązonego algorytmu.

Współczynnik opisywany w tabeli jest liczony następująco:

$$wsp = \frac{apnea}{(end - begin)} \quad (1.32)$$

gdzie:

begin - numer pierwszej próbki rekordu

end - numer ostatniej próbki rekordu

apnea - ilość próbek w których wykryto bezdech senny

Ilość przedziałów mówi o liczbie niezależnych odcinków czasu w całym sygnale, gdzie wykryto bezdech senny

Dane	Physionet		Algorytm	
	Współczynnik	Ilość przedziałów	Współczynnik	Ilość przedziałów
a01	95,69	3	97,71	10
a02	77,61	11	69,83	40
a03	45,37	11	67,86	28
a04	91,65	3	91,41	8
a05	57,62	15	83,97	36

Tablica 1.17: Tabela porównawcza dla danych pobranych z Physionetu oraz danych wyjściowych algorytmu

Jak można zauważyć, zaimplementowany przez nas algorytm w większości przypadków identyfikuje bezdech senny w większej ilości próbek w porównaniu z badaniami pochodząymi z bazy Physionetu. Taka zależność może być spowodowana:

1. Błędnym wykryciem R-pików
2. Uтратą części danych po etapach filtracji i przepróbkowania
3. Wyliczaniem progowej wartości amplitudy na podstawie stałych współczynników

W przypadku rekordu zdrowego człowieka algorytm wykrywa bezdech senny, ze względu na to, iż wartość progowa amplitudy jest ściśle powiązana z maksymalną i minimalną wartością amplitudy przetworzonego sygnału. Aby wyeliminować ten błąd należało by wykrywać bezdech senny nie tylko na podstawie wyliczonej wartości progowej, ale także uwzględniając zewnętrze dane statystyczne, niezależne od badanego rekordu.

1.10.4 Literatura

[1] T. Penzel, J. McNames, A. Murray, P. de Chazal, G. Moody, and B. Raymond, "Systematic comparison of different algorithms for apnoea detection based on electrocardiogram recordings," Med. Biol. Eng. Comput., vol. 40, no. 4, pp. 402–407, Jul. 2002.

[2] Mendez M.O., Bianchi A.M., Matteucci M., Cerutti S., Penzel T., "Sleep Apnea Screening by Autoregressive Models From a Single ECG Lead", IEEE Transactions on Biomedical Engineering, Vol. 52, No. 12, 2009, pp. 2838-2850

[3] Isa S.M., Fanany M.I., Jatmiko W., Arymurthy A.M., "Sleep Apnea Detection from ECG Signal: Analysis on Optimal Features, Principal Components", and Nonlinearity, International Conference on Bioinformatics and Biomedical Engineering, 2011, pp. 1-4

[4] J. E. Mietus, C. K. Peng, P. Ch. Ivanov, A. L. Goldberger, "Detection of Obstructive Sleep Apnea from Cardiac Interbeat Interval Time Series", Beth Israel Deaconess Medical Center and Harvard Medical School, Boston, USA

1.10.5 Diagram klas

SleepApnea
<pre> dataFreq : int=samplingFrequency window : int = 41 lfilt : int = 32 windowMedian : int = 60 sleepApnea(samplingFrequency : const int) sleepApnea plots(tabRpeaks : QVector<unsigned int>) sleepApnea output(tabRpeaks : QVector<unsigned int>) guiOutput(tabRpeaks : QVector<unsigned int>) RRintervals(samplingFrequency : const int) averangeFilter(tab RR : QVector<QVector<double> >) resample(tabRRnew : QVector<QVector<double> >) HpLpFilter(&tabRes : QVector<QVector<double> >) hilbert(tabRes : QVector<QVector<double> >,&hAmp : QVector<QVector<double> >,&hFreq : QVector<QVector<double> >) freqAmpFilter(&hAmp : QVector<QVector<double> >,&hFreq : QVector<QVector<double> >) medianFilter(&hAmp : QVector<QVector<double> >,&hFreq : QVector<QVector<double> >) apneaDetection(tabAmp : QVector<QVector<double> >, tabFreq : QVector<QVector<double> >) </pre>

1.11 Moduł ATRIAL_FIBR

1.11.1 Badania literaturowe

Celem jest wykrycie u osoby badanej migotania przedsionków (ang. *Atrial Fibrillation*). Jest to zaburzenie polegające na nieskoordynowanym pobudzeniu przedsionków serca. Ryzyko zgonu u osoby chorej wzrasta dwukrotnie, a problem dotyczy 1 – 2% ogółu populacji. Migotanie przedsionków zazwyczaj jest bezobjawowe, zaburzenie jest wykrywane przypadkowo, lub po udarze niedokrwieniennym mózgu. Zwykle występuje u pacjentów z organiczną chorobą serca, ale również może wystąpić u osób zdrowych. Możliwe do zaobserwowania objawy to duszności, kołatanie serca, ból w klatce piersiowej czy zawroty głowy.

Sygnal EKG u chorego charaktryzuje się następującymi własnościami:

- brak załamków P
- odstępy R-R nieregularne
- obecna fala migotania
- zespoły QRS zwykle wąskie



Rysunek 1.65: Sygnał EKG dla osoby chorej i prawidłowy rytm zatokowy

Rysunek 1.65 na górnym zapisie przedstawia sygnał EKG dla osoby chorej na migotanie przedsionków, natomiast dolny zapis prezentuje prawidłowy rytm zatokowy. Wykres osoby chorej wyraźnie przedstawia wszystkie podane symptomy.

Literatura podawała kilka czynników mogących wskazywać na występowanie u pacjenta migotania przedsionków. Jeden artykuł [1] przedstawał metodę opartą jedynie na detekcji peaków R. Autorzy uważali, że na podstawie jedynie nieregularnych interwałów R-R można z powodzeniem wykryć u badanego chorobę. Metoda ta polega na zaklasyfikowanie odległości jako małych, dużych i normalnych. Następnie przedstawienie jako procesu Markowa i porównanie do wyników osoby zdrowej miało pozwolić na uzyskanie ostatecznego wyniku. Niestety, po wyliczeniu parametrów dotyczących serca, okazało się, że jeden z próbnych sygnałów dostępnych w bazie Physiobank - iaf4_ivc - wykazuje własności bardzo zbliżone do zdrowego człowieka.

Autorzy drugiego źródła [2], oprócz wspomnianej rozszerzyli sposób rozwiązania problemu dodając następujące metody:

1. Wykrycie absencji załamka P

Po ustaleniu pewnego wzorca P (na podstawie dużej liczby próbek), zostaje porównany do sygnału osoby badanej. Ilość wystąpień załamka zostaje wyznaczona na podstawie wyznaczonego progu.

2. Analiza aktywności przedsionków

Po usunięciu QRS-T, sygnał u osoby chorej charakteryzuje się występowaniem fal o częstotliwości 4-10 Hz. Należy wykonać analizę widma – po zastosowaniu FFT (Fast Fourier Transform) trzeba sparametryzować widmo częstotliwości. U osoby chorej można zaobserwować koncentrację wokół wierzchołka położonego w przedziale [4,10] Hz.

1.11.2 Koncepcja proponowanego rozwiązania

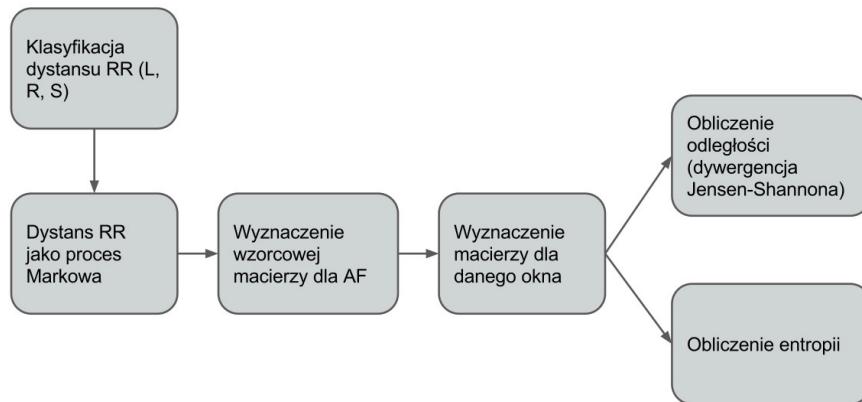
Początkowo zasugerowani artykułem z MIT postanowiliśmy wypróbować metodę tam przedstawioną i ograniczyć się do analizy interwałów R-R. Jednak po niepowodzeniu zmuszeni byliśmy wybrać inną metodę. Ze względu na istotę problemu – zdecydowaliśmy się również wykorzystać metodę wykrywania absencji P.

Analiza sygnału będzie więc polegała na zbadaniu występowania 2 parametrów:

1. Odległości pomiędzy załamkami R
2. Brak załamków P

Gdy oba warunki zostaną spełnione, uznamy, że występuje migotanie przedsionków w danym fragmencie sygnału. Okno analizowanego sygnału wynosi 100 uderzeń serca - 60 załamków P. Wartość ta musiała być wystarczająco duża, by móc odpowiednio zbadać wszystkie wartości. Jednak przy zbyt dużym oknie wynik mógłby zostać błędnie zinterpretowany.

Parametry związane z rytmem serca Informację na temat położenia w czasie peaków R otrzymamy z modułu QRS Complex. Na jej podstawie zakwalifikujemy każdy przedział RR jako krótki (poniżej 85% średniej), średni, oraz długi (powyżej 115% średniej). Następnie wyliczymy prawdopodobieństwo przejść między poszczególnymi stanami. Finalnie otrzymamy z tego dwa parametry - entropię sygnału oraz odchylenie sygnału od średniej macierzy zdrowego człowieka metodą Jensa-Shannona.



	S	M	L
S	0.005	0.023	0.006
M	0.007	0.914	0.013
L	0.019	0.006	0.003

Tablica 1.18: Prawdopodobieństwo zmiany długości dystansu RR w przypadku zdrowego człowieka. S - dystans krótki (mniej niż 85% średniego), M - średni, L - długi (ponad 115% średniego)

	S	M	L
S	0.056	0.114	0.062
M	0.101	0.350	0.096
L	0.060	0.114	0.035

Tablica 1.19: Uśredniona macierz Markowa dla ludzi chorych na migotanie przedsionków

Prawdopodobieństwo przejścia ze stanu R (regular) do kolejnego R wynosi:

$$P(R|R) = \frac{P(R,R)}{P(R)} \quad (1.33)$$

Jako miary opisującej odległość między macierzami początkowo, zgodnie z zaleceniami pozycji [?], chcieliśmy użyć dywergencji Kullbacka-Leiblera:

$$F_{KL}(P(E_i, E_j), \overline{P_{AF}(E_i, E_j)}) = \sum_{i=1}^3 \sum_{j=1}^3 P(E_i, E_j) \log \left(\frac{P(E_i, E_j)}{\overline{P_{AF}(E_i, E_j)}} \right) \quad (1.34)$$

natomast ze względu na to, że jest asymetryczna, czyli de facto nie jest metryką, postanowiliśmy skorzystać z dywergencji **Jensena-Shannona**. Bazuje na powyższej dywergencji, z pewnymi znaczącymi różnicami. Jest symetryczna i zawsze ma skońzoną wartość. Pierwiastek kwadratowy dywergencji Jensena-Shannona jest metryką zwaną dystansem Jensena-Shannona.

$$F_{JS} = \frac{1}{2} F_{KL}(P(E_i, E_j), M) + \frac{1}{2} F_{KL}(\overline{P_{AF}(E_i, E_j)}, M) \quad (1.35)$$

gdzie

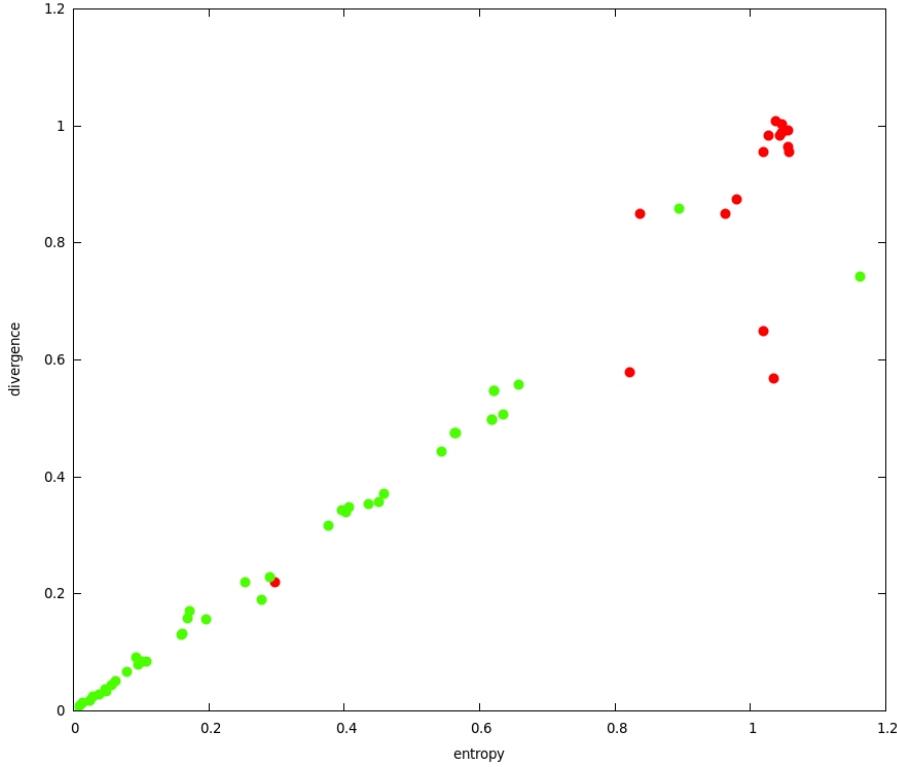
$$M = \frac{1}{2}(P(E_i, E_j) + \overline{P_{AF}(E_i, E_j)})$$

W przypadku entropii skorzystaliśmy ze standardowej definicji:

$$H = \sum_{i=1}^3 P(E_i | E_j) \quad (1.36)$$

$$H(E_j) = - \sum_{j=1}^3 P(E_j | E_i) \log_2 P(E_j | E_i) \quad (1.37)$$

Metoda została przez nas wstępnie przetestowana przy użyciu aplikacji napisanej w języku programowania Julia. Pozwoliło to nam wysnuć pierwsze wnioski z wykresu 1.66. Zielone kropki przedstawiają parametry dla sygnałów z bazy Physiobank zebranych od zdrowych ludzi, czerwone natomiast pochodzą od chorych na migotanie przedsionków. Niestety nie jest możliwe oddzielenie puli kropek zielonych od czerwonych jedną linią, dlatego potrzebujemy dodatkowego parametru.



Rysunek 1.66: Entropia i dywergencja dla danych Physiobank

Detekcja absencji P Z modułu WAVES otrzymamy informację na temat położenia początkowego i końcowego załamka P. Na tej podstawie wyliczymy średni przebieg tej fali u zdrowego człowieka. Następnie przy użyciu oszacowanego progu wyliczymy procentową absencję fali P w zadanym sygnale, która będzie stanowiła kolejny współczynnik.

Wyliczenie współczynnika korelacji:

$$cc(i) = |\text{corr coef}(P_{\text{wave}}(i), \overline{P_{\text{wave}}(i)})|, \quad i = 1, \dots, n \text{ beats} \quad (1.38)$$

$$|\text{corr coef}(x, y)| = \frac{\sum_{i=1}^n (x(i) - \bar{x})(y(i) - \bar{y})}{\sqrt{\sum_{i=1}^n (x(i) - \bar{x})} \sqrt{\sum_{i=1}^n (y(i) - \bar{y})}} \quad (1.39)$$

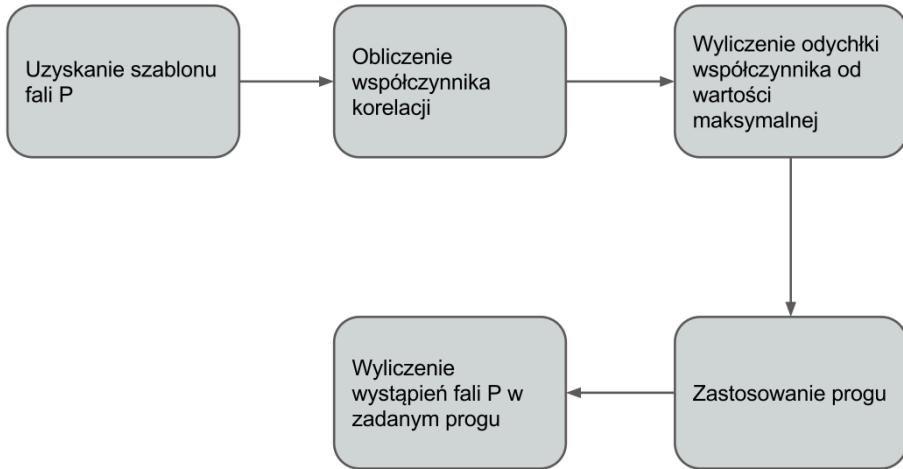
Za wykrytą falę P uznajemy tę, dla której współczynnik korelacji wynosi co najmniej 0.2. Stosunek liczby wykrytych fal P do liczby wszystkich okresów PQRST jest poszukiwanym współczynnikiem.

Badanie przeprowadzane jest dla pewnych okien czasowych

1.11.3 Rezultaty i wnioski

Zaproponowane przez nas metody zostały poprawnie zaimplementowane w języku C++ oraz zintegrowane z główną linią projektu. Integralną częścią dostarczonego przez zespół kodu był zestaw testów jednostkowych napisanych z wykorzystaniem framework'a QtTest.

Zgodnie z koncepcją proponowanego rozwiązania, ocena prawdopodobieństwa zachorowania pacjenta na migotanie przedsiornków jest średnią ważoną z następujących parametrów:



Rysunek 1.67: Algorytm wyliczenia współczynnika występowania fali P.

- **Dywergencja Jensena-Shannona** między wzorcową macierzą Markowa opisującą prawdopodobieństwo przejść między stanami, a macierzą wyliczoną dla danego pacjenta. Dywergencja 0 oznacza, że macierz jest identyczna z wzorcową.
- **Entropia macierzy Markowa.** Wartość 0 oznacza, że wszystkie wartości dystansów między falami R mieszczą się w przedziale [85%, 115%] wartości średniej. Z drugiej strony, wartość 1 mówi o dużej nieregularności występowania fali R, co jest symptomatyczne dla osób cierpiących na migotanie przedśionków.
- **Współczynnik zaniku fali P.** Stosunek liczby okresów PQRST w których fala P występuje, do całkowitej liczby tych okresów. Za występowanie fali P uznajemy sytuację, w której korelacja między sygnałem w przedziale otrzymanym z modułu WAVES, a sygnałem wzorcowym jest większa od 0.2.

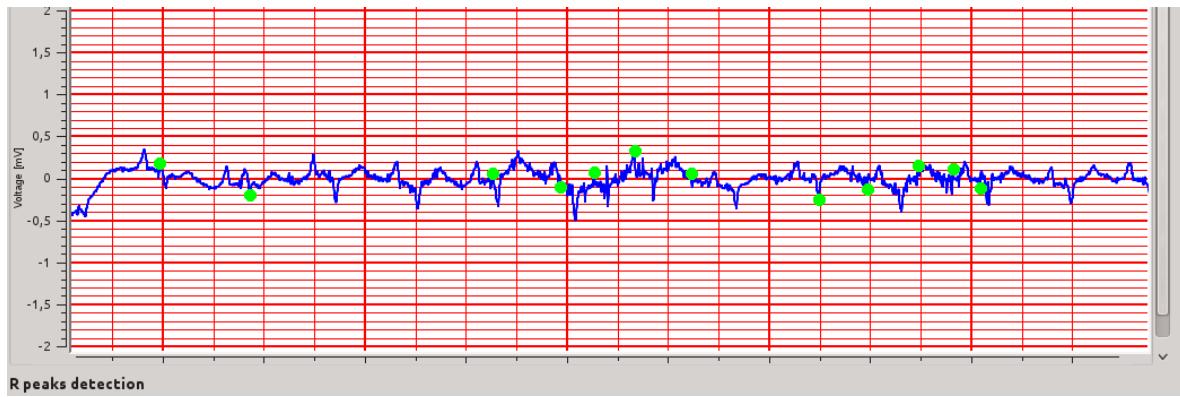
Po zaaplikowaniu wyżej wspomnianych metod dla wzorcowych sygnałów z bazy Physiobank, można stwierdzić, że program dokonuje detekcji poprawnie. Właściwie dla większości przypadków wystarczająca jest sama analiza rytmu bicia serca (dystansów R-R). Zastosowanie drugiej metody związanej z absencją fali P dodatkowo zwiększyło skuteczność wykrywania choroby.

Weryfikacja na podstawie otrzymanych danych Moduł jest zależny od wyników uzyskanych od innych. Gdyby wszystkie dane były dostarczone wcześniej, można by było wygenerować średnią falę P ze wszystkich sygnałów. Dodatkowo, dla różnych elektrod fala P różni się kształtem i nie da się dobrać jednej uniwersalnej. Konieczne byłoby uśrednienie na podstawie otrzymanych początków i końców fal P, by otrzymać średni wzorzec. Ze względu na charakter tego projektu - wszystkie moduły pracowały równolegle i dopiero pod koniec wszystkie zależności były gotowe - zrealizowanie tego wykracza poza nasze możliwości. Jest to jednak doskonały punkt, w którym można kontynuować projekt. Metody przez nas zaimplementowane są dokładnie przetestowane, należałyby stworzyć odpowiedni algorytm do uzyskania średniej fali P. Dlatego w tym momencie zaimplementowane jest rozwiązanie dla sygnału V1, w przypadku pozostałych, współczynnik zaniku fali P nie jest obliczany.

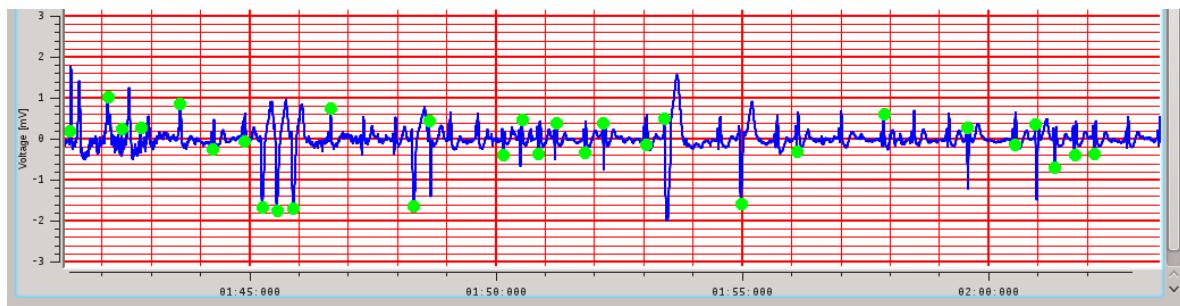
Numer próbki	Absencja fali P	Dywergencja	Entropia	Chory
101	0.20	0.10	0.71	NIE
105	0.97	0.04	0.34	NIE
106	0.83	0.49	0.95	TAK
107	0.78	0.05	0.57	NIE
108	0.87	0.56	0.95	TAK
109	0.86	0.07	0.65	NIE
111	0.18	0.34	0.96	NIE
112	0.68	0.05	0.00	NIE
113	0.13	0.07	0.72	NIE
115	0.46	0.14	0.78	NIE
116	0.85	0.07	0.59	NIE
118	0.81	0.06	0.65	NIE
119	0.99	0.46	1.00	TAK
121	0.46	0.39	0.94	NIE
122	0.88	0.04	0.10	NIE
200	0.67	0.42	1.01	NIE
201	0.64	0.52	0.92	NIE
202	0.66	0.22	1.01	NIE
203	0.72	0.43	1.04	TAK

Tablica 1.20: Rezultaty uzyskane dla poszczególnych danych

Tablica 1.20 powstała w oparciu o dane istniejące już w formacie *.dat*. Wynik 1.04 dla entropii jest wynikiem błędu numerycznego. Nie było możliwości sprawdzenia wyników w programie z danych zimportowanych z bazy Physiobank. Po podanym na stronie sposobie konwersji, sygnał nie może być wczytany. Przez to nie jesteśmy w stanie zaprezentować wyników jak w przypadku testowego programu w języku Julia. Sporządziliśmy więc listę wyników możliwych do osiągnięcia, jednak żadnen z nich nie zawierał w sobie danych osoby chorej na migotanie przedsionków. Te wyniki, które program zakwalifikował pozytywnie, okazały się błędne ze względu na charakterystykę sygnału. Przykład stanowi rysunek 1.68 lub 1.69. Wykresy przedstawiają całkowicie zakłócone i niepoprawne przebiegi dla EKG. Prawidłowe działanie naszego modułu wymaga odpowiednie przefiltrowanie i usunięcie fragmentu sygnału, który może spowodować błędny wynik.



Rysunek 1.68: Fragment próbki nr 108



Rysunek 1.69: Fragment próbki nr 203

1.12 Moduł QT_DISP

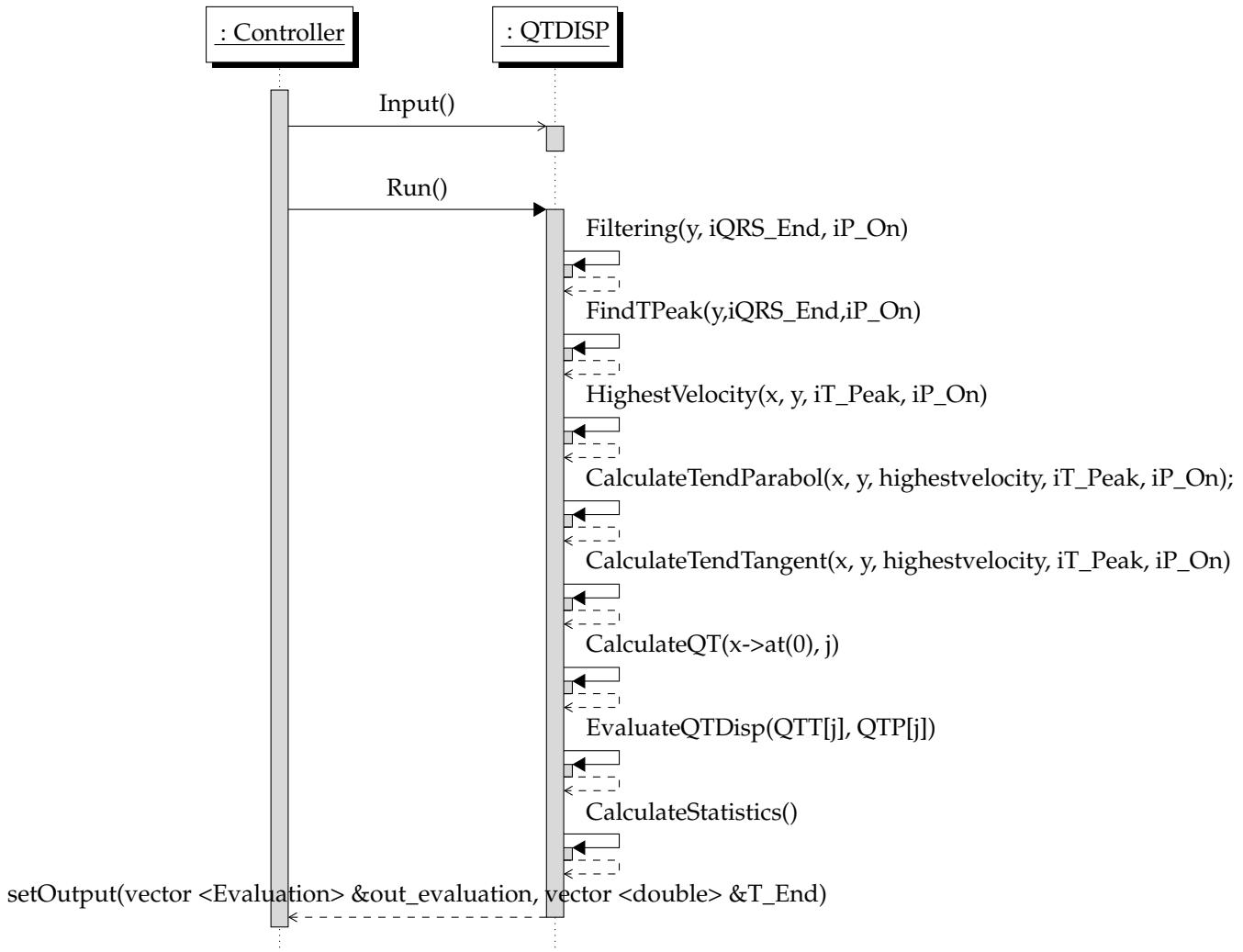
1.12.1 Diagramy klas i sekwencji

1.12.1.1 Diagram klas

QTDISP
<ul style="list-style-type: none"> - QRS_On : vector <int> - QRS_End : vector <int> - P_On : vector <int> - heartBeats : int - T_Peak : vector <int> - T_EndP : vector <double> - T_EndT : vector <double> - heartBeats : int - signals2 : vector <double> - QTP : vector <double> - QTT : vector <double> + getInput(vector <double> in_signals2, vector <int> in_QRS_On, vector <int> in_QRS_End, vector <int> in_P_On, double in_samplingFrequency) + getInput(string path) + setOutput(vector <Evaluation> &out_evaluation, vector <double> &T_End) + Run() - CalculateTend(vector<double> x, vector<double> y, int QRS_End, int P_On, int T_Peak) - CalculateQT(double QRS_OnTime, int number_T_End_QT) - Filtering(vector<double> *y, int QRS_End, int P_Onset) - FindTPeak(vector<double> *y, int QRS_End, int P_Onset) - poliFitting(double* a, double* b, double* c, vector <double> x, vector <double> y, int Tpeak, int P_Onset) - Tangent(double* a, double* b, vector <double> *x, vector <double> *y,int HighestVelocityPoint) - CalculateTendTangent(vector <double> *x, vector <double> *y, int highestVelocity, int tPeak, int P_Onset) - CalculateTendParabol(vector <double> *x, vector <double> *y, int highestVelocity, int tPeak, int P_Onset) - DispersionEvaluation (double gapQT, double heartAction, int* BazzetState, int* FridericState, int*,HodgesState, int* FraminghamState, double* BazzetValue, double* FridericValue, double* HodgesValue, double* FraminghamValue) - EvaluateQTDisp(double QTT, double QTP) - EvaluateBazzet(double gapQT, double RR) - EvaluateFrideric(double gapQT, double RR) - EvaluateHodges(double gapQT, double heartAction) - EvaluateFramingham(double gapQT, double RR) - CalculateStatistics()

Evaluation
<pre>+ nameOfEvaluation : string + numberOfCorrectQT : int + numberOfTooLowQT : int + numberOfTooHighQT : int + percentOfCorrectQT : double + percentOfTooLowQT : double + percentOfTooHighQT : double + averageQT : double + standardDeviationQT : double + CalculatePercentage() + CalculateAverage(vector <double> *QT) + CalculateStandardDeviation(vector <double> *QT) + StatisticalEvaluation(vector <double> *QT, double avg, double stdDev)</pre>

1.12.1.2 Diagram sekwencji



1.12.2 Badania literaturowe

Dyspersja odcinka QT, oznacza różnicę czasu, między najdłuższym i najkrótszym czasem QT i oddaje ona niejednorodność przestrzenną repolaryzacji. Ten właśnie odcinek jest najdłuższy wśród osób, u których występowały zawały serca spowodowane migotaniem komór sercowych. Długość odstępu QT jest zależna od częstości pracy serca, jednak nie istnieje jeden algorytm, który podawałby wprost tę zależność. Najczęściej stosowanymi algorytmami są algorytmy : Bazzeta:

$$gapQTc = \frac{gapQT}{\sqrt{RR}} \quad (1.40)$$

Friderica:

$$gapQTc = \frac{gapQT}{\sqrt[3]{RR}} \quad (1.41)$$

Hodgesa:

$$gapQTc = gapQT + 1.75 * (heartAction - 60) \quad (1.42)$$

Framighama:

$$gapQTc = gapQT + 0.154 * (1 - RR) \quad (1.43)$$

$$RR = \frac{60}{heartAction} \quad (1.44)$$

Gdzie:

- gapQTc - Skorygowany odstęp QT
- gapQT - odstęp QT
- heartAction - akcja serca w uderzeniach na minutę

Każdy z nich posiada inne zakresy poprawności, jednak algorytmy te są skuteczne jedynie w pewnych wąskich zakresach częstości pracy serca. W przypadku przyspieszonej lub zwolionej pracy serca, określone statystycznie przedziały działania dla tych algorytmów mogą okazać się błędne. Dodatkowo, sytuacje takie jak stosowanie leków (m.in. psychotropowych i antyarytmicznych), zaburzenia elektrolitowe(np. hipokaliemia) czy niedoczynność tarczycy także mogą wpływać na wyniki algorytmów. W przypadku aparatów EKG, analiza odstępu QT często opiera się na ręcznym jego wyznaczaniu przez operatora, co znacznie może wydłużać interpretację badania. Jednak istnieją metody umożliwiające automatyczne wyliczanie odstępu QT, co zostanie opisane w poniższej części.

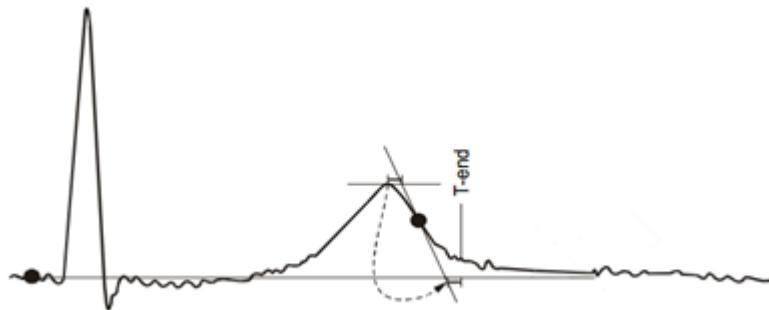
1.12.3 Realizacja projektu

Sama realizacja opierać się będzie głównie na zaproponowanych w pracy prof. Augustyniaka algorytmach wyznaczania końca załamka T. Przyjętym założeniem jest otrzymanie początku odstępu QT w ramach danych wejściowych. Otrzymanie końca załamka T odbywać się będzie na dwa sposoby: sposób pierwszy:

- Odnalezienie stycznej do zstępującego ramienia załamka T, w punkcie o maksymalnej prędkości.
- Odszukanie punktów przecięcia przez styczną linii izoelektrycznej i poziomu maksimum T.

- Odłożenie za punktem przecięcia izolinii przez styczną odcinka od wystąpienia maksimum T do przecięcia stycznej pozwoli na uzyskanie na końcu odcinka na izolinii końca załamka T.

Metoda ta prezentuje się następująco:

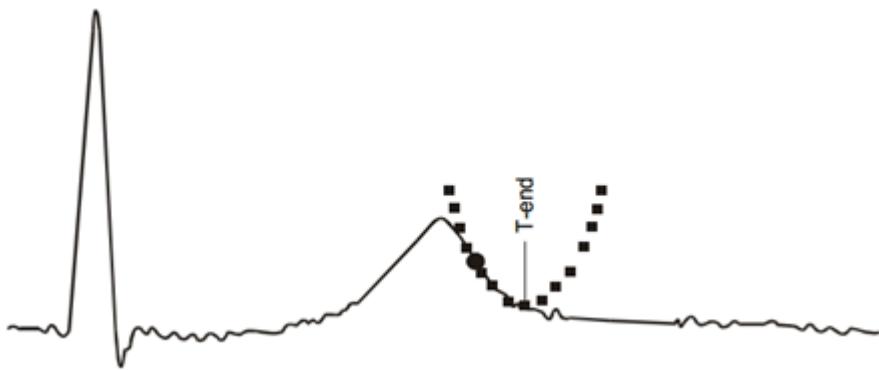


Rysunek 1.70: Metoda stycznych

sposób drugi:

- Odnalezienie punktu o maksymalnej prędkości na zstępującym ramieniu załamka T
- Wyznaczenie najlepiej dopasowanej paraboli do punktów chronologicznie późniejszych, ale poprzedzających globalnie wyznaczony koniec załamka T
- Wyznaczenie wierzchołka takiej paraboli jest tożsame ze znalezieniem końca załamka T.

Metoda ta prezentuje się następująco:



Rysunek 1.71: Metoda parabol

1.12.4 Rezultaty i wnioski

W poniższej tabeli zaprezentowane są wyniki kilku obliczeń szerokości zespołów QT.

Próbka	Bazzet			Frideric		
	OK [%]	Min [%]	Max [%]	OK [%]	Min [%]	Max [%]
100.dat	0,4	1,5	98,1	0,4	1,5	98,1
101.dat	70,1	27,7	1,4	77,8	8,4	13,8
102.dat	0,4	41,1	58,6	0,4	41,0	58,6
105.dat	54,1	24,7	21,2	54,2	29,7	16,2
106.dat	45,3	37,3	17,4	42,1	29,5	28,4
107.dat	0,3	56,7	43,0	0,4	56,6	43,0
111.dat	38,5	59,1	2,39	42,9	53,8	3,2
116.dat	27,7	70,2	2,1	28,6	69,4	1,9

Rysunek 1.72: Tabela zawierająca opisy kilku przeprowadzonych eksperymentów : część 1

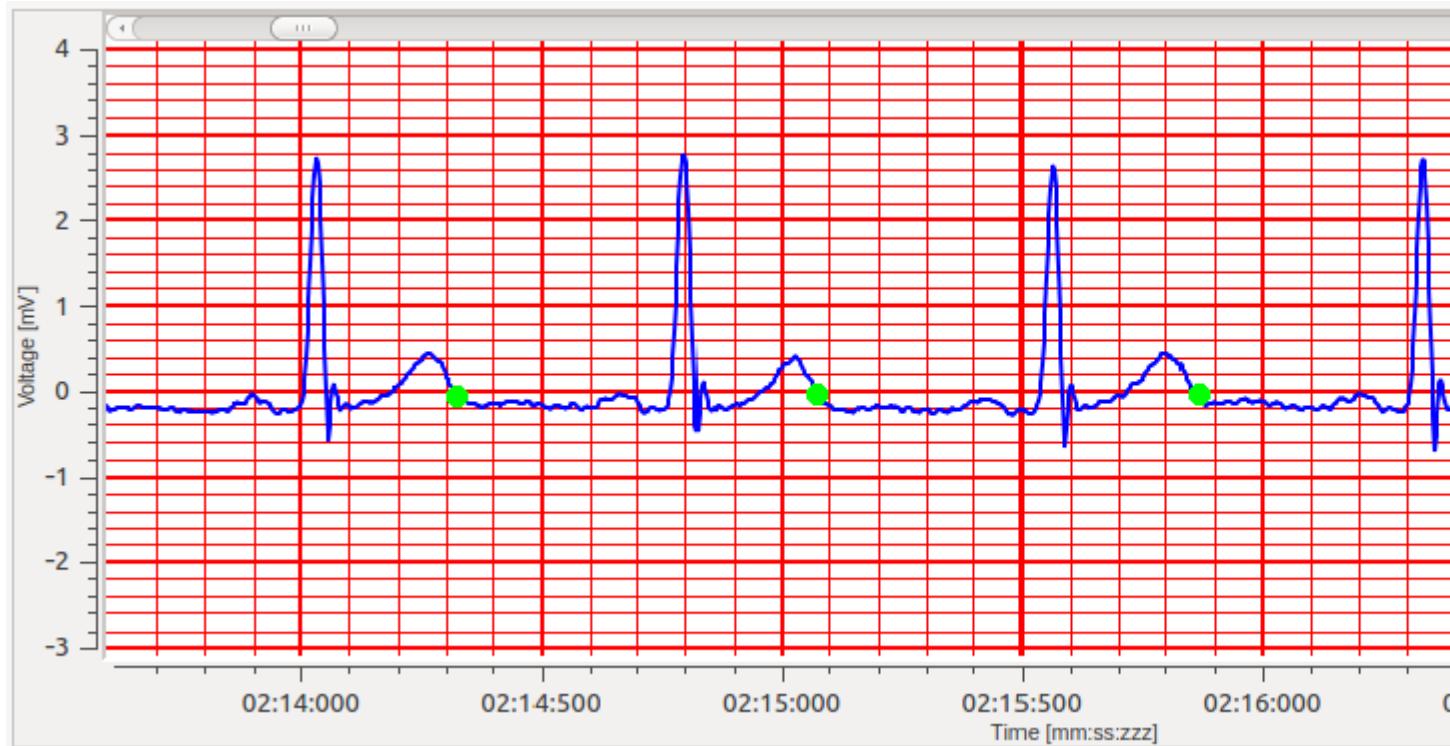
Hodges			Framigham			Średni czas QT [s]	Odchy- lenie standar- dowe [s]
OK [%]	Min [%]	Max [%]	OK [%]	Min [%]	Max [%]		
2,91	1,63	95,5	0,4	1,5	98,1	0,5049	0,0717
79,0	14,4	6,6	76,2	11,6	12,2	0,4021	0,0604
0,7	41,2	58,1	0,4	41,1	58,6	0,3563	0,2075
5,0	86,0	8,9	53,1	32,0	14,8	0,3304	0,2151
45,4	35,0	19,7	41,9	31,7	26,4	0,3749	0,2450
0,2	56,8	43,0	0,3	56,6	43,0	0,0659	3,1851
12,3	87,1	0,6	37,9	59,7	2,39	0,1936	0,8429
1,8	97,4	0,8	20,3	78,4	1,3	0,3114	0,1035

Rysunek 1.73: Tabela zawierająca opisy kilku przeprowadzonych eksperymentów : część 2

Gdzie:

- OK – prawidłowa długość załamka według danej ewaluacji [
- MAX – zbyt wielka długość załamka [
- MIN – zbyt mała długość załamka [

Poniżej też zaprezentowany jest na wykresie jeden zrzut ekranu z sygnału z oznaczonymi na nim końcami załamków T:



Rysunek 1.74: Przykładowe oznaczenie końców załamków T na wykresie EKG

1.13 Moduł HRT

1.13.1 Badania literaturowe

Zagadnienie HRT w literaturze

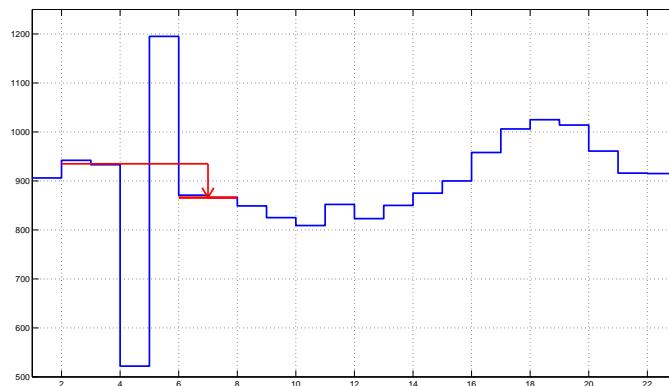
HRT - Turbulencja Rytmu Serca - to dwufazowa odpowiedź węzła zatokowego na przedwcześni skurcz komorowy. Zjawisko turbulencji rytmu serca jest powiązane ze sprawnością odruchu z baroreceptorów, charakteryzuje odpowiedź węzła zatokowego na przedwczesne pobudzenie komorowe (VEB lub VEP), po którym dochodzi do przyspieszenia, a następnie do zwolnienia rytmu zatokowego. Oznacza to, że czas trwania kolejnych cykli (długość odstępów RR) początkowo jest skrócony (następuje kompensacja) i stopniowo się wydłuża. Wystąpienie turbulencji po dodatkowym skurczu komorowym jest objawem prawidłowej funkcji baroreceptorów.



Rysunek 1.75: Zapis EKG wraz z ciśnieniem krwi

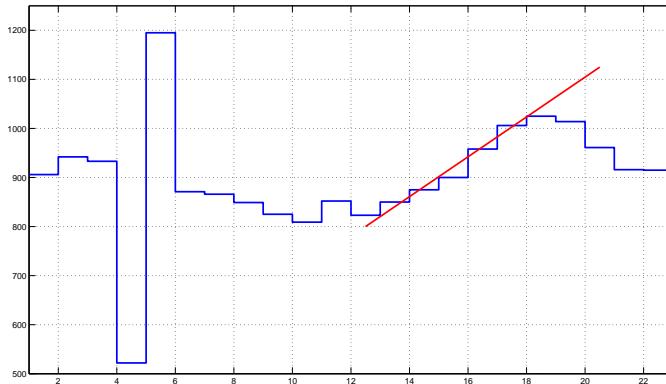
HRT jest ocenione przez dwa liczbowe parametry: Turbulence Onset (początek turbulencji), oraz Turbulence Slope (nachylenie turbulencji). Przeprowadzenie wielu badań zarówno u pacjentów bez dolegliwości sercowych jak i u ludzi chorych pozwoliło na określenie prawidłowych wartości obu parametrów.

Wartości prawidłowe: TO<0%, TS>2.5ms/RR



Rysunek 1.76: Tachogram z graficzną reprezentacją TO

Analiza HRT okazała się przydatna w ocenie ryzyka po zawałe serca. Po przeanalizowaniu danych pacjentów po zawałe serca udało się ustalić, że HRT - niezależnie od innych czynników ryzyka - posiada bardzo dużą, a może nawet największą wartość predykcyjną spośród elektro-



Rysunek 1.77: Tachogram z graficzną reprezentacją TS

trokardiograficznych parametrów wykorzystywanych w celach progностycznych.

Podejście do problemu w literaturze

- Wybór - filtracja danych wejściowych W celu uniknięcia błędów w analizie HRT przyjmuje się pewne kryteria doboru danych poddanych analizie. Bazując na odstępach czasowych interwałów RR klasyfikuje się skurcze komorowe jako przedwczesne. Skrócony RR (przedwczesny skurcz) musi być co najmniej o 20% krótszy od wartości średniej z poprzednich 5 interwałów. Pauza po VEB ma być dłuższa co najmniej o 20% w porównaniu do tej samej wartości średniej. Zalecana jest również filtracja ze względu na długość interwałów po i przed VEB, tak aby wykluczyć zbyt krótkie oraz zbyt długie RR'y. Przyjęto następujące kryteria dotyczące pojedynczego interwału:

- * $RR > 300\text{ms}$,
- * $RR < 2000$,
- * $RR - RR - 1 > 200$

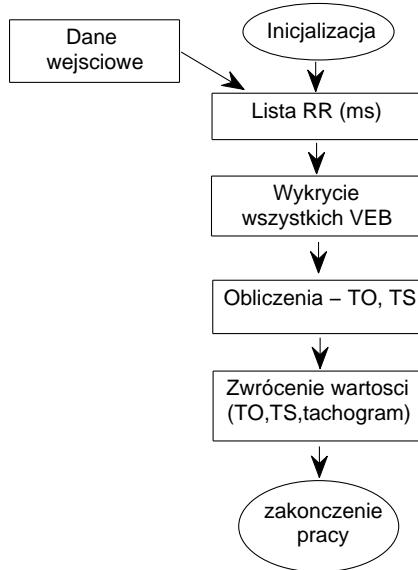
- Wyznaczanie parametrów z wybranych danych Początek turbulencji oblicza się jako różnicę między średnim czasem trwania 2 pierwszych pobudzeń zatokowych następujących po VEB: $(RR_1 + RR_2)/2$, a średnim czasem trwania 2 ostatnich pobudzeń poprzedzających: $(RR_{-1} + RR_{-2})/2$, podzieloną przez średni czas trwania 2 ostatnich pobudzeń poprzedzających. Parametr ten jest wyrażany w procentach na podstawie wzoru, który ma postać:

$$TO = \frac{(RR_1 + RR_2) - (RR_{-1} + RR_{-2})}{(RR_{-1} + RR_{-2})} * 100\% \quad (1.45)$$

Nachylenie turbulencji określa się je jako największe nachylenie prostej regresji wyznaczonej z wykresu czasu trwania wybranych 5 kolejnych odstępów RR jako funkcji czasu spośród 15-20 cykli następujących po VEB. Jednostką TS jest ms/RR.

1.13.2 Koncepcja proponowanego rozwiązania

Po zapoznaniu się z literaturą przystąpiono do rozpisania schematu blokowego algorytmu (opisywany w ten sam sposób w publikacjach).



Rysunek 1.78: Schemat blokowy algorytmu

Koncepcja algorytmu

Dane wejściowe:

- Przefiltrowany sygnał EKG,
- numery próbek pików R,
- częstotliwość próbkowania,
- sklasyfikowane zespoły QRS

Dane wyjściowe:

- Parametry turbulencji rytmu serca (TO, TS)
- Dane do utworzenia wykresu pozwalającego na analizę HRT przez człowieka
- Liczba VEB, które zostały dopuszczone do analizy

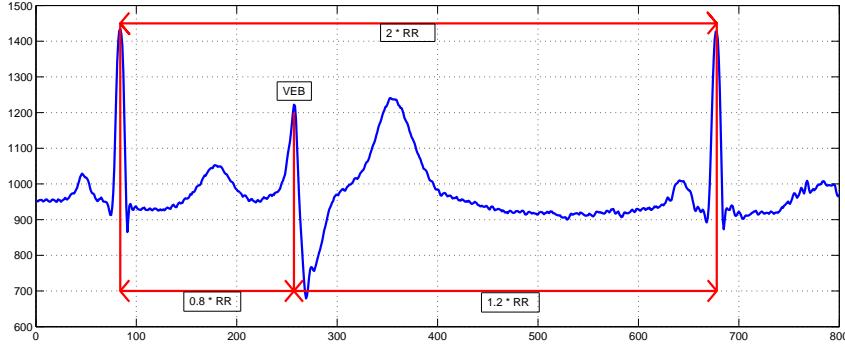
Kroki algorytmu:

- Lista RR
Na podstawie danych z modułu R_PEAKS utworzenie listy z długościami kolejnych interwałów – wyróżnionymi w ms (dla którego wymagana jest częstotliwość próbkowania)
- Wykrycie VEB (przedwczesne skurcze komorowe)
Na podstawie danych z modułu QRS_CLASS oraz przy użyciu wcześniej utworzonej ListyRR wybranie VEB'ów. Dane z modułu QRS_CLASS muszą być dodatkowo sprawdzone, aby analiza HRT dawała poprawne wyniki. Warunki konieczne (przedstawione

również graficznie na rysunku):

$$\frac{(RR - RR_{v1})}{RR} > 0.2 \quad (1.46)$$

$$\frac{(RR_{v2} - RR)}{RR} > 0.2 \quad (1.47)$$



Rysunek 1.79: Wykres sygnału EKG z oznaczonym VEB

- Filtracja

Filtracja interwałów RR po i przed VEB, kryteria:

- * $RR > 300\text{ms}$,
- * $RR < 2000\text{ms}$,
- * $RR - RR_{-1} < 200\text{ms}$

Filtracja interwałów RR po i przed VEB (kryteria podane w literaturze) – jeśli interwał nie spełnia warunków cały ciąg próbek odpowiadający danemu VEB nie będzie brany pod uwagę w następnym kroku algorytmu. Wymagane jest co najmniej 5 poprawnych ciągów próbek, aby analiza była skuteczna (o nie spełnieniu danego warunku użytkownik powinien być powiadomiony).

- Utworzenie tachogramu dla każdego wykrytego VEB. Wymagane jest cztery próbki pochodzące przedwcześnie skurcz komorowy i dwadzieścia jeden po.
- Obliczenia

- * TO – wyliczane dla każdego VEB wg wzoru:

$$TO = \frac{(RR_1 + RR_2) - (RR_{-1} + RR_{-2})}{(RR_{-1} + RR_{-2})} * 100\% \quad (1.48)$$

Następnie wybieramy wartość średnią.

- * TS – wartość maksymalna współczynnika kierunkowego a wyliczona metodą regresji liniowej z uśrednionego tachogramu:

$$a = \frac{n \sum x_i y_i - \sum x_i \sum y_i}{n \sum x_i^2 - (\sum x_i)^2} \quad (1.49)$$

$$b = \frac{1}{n} (\sum y_i - a \sum x_i) \quad (1.50)$$

- Zwrócenie danych
 - * Liczba VEB dopuszczonych do analizy
 - * TO
 - * TS
 - * Uśredniony tachogram

1.13.3 Rezultaty i wnioski

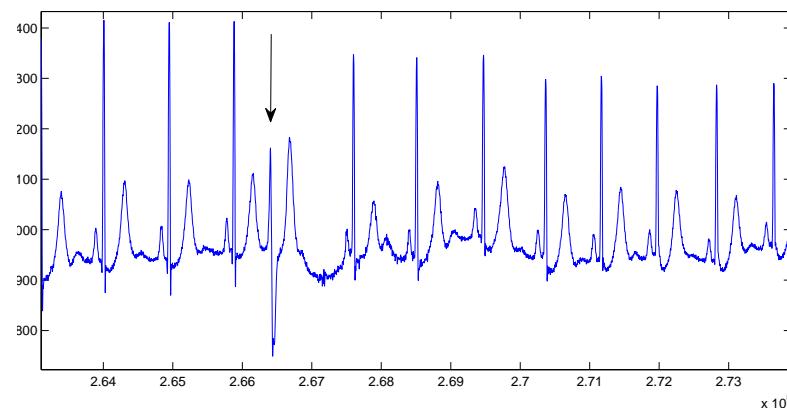
Przedstawiony algorytm został zaimplementowany w języku C++, oraz języku skryptowym oprogramowania Matlab. Skryptem posłużono się w celu weryfikacji i analizy działania modułu.

Działanie algorytmu było testowane na próbkach przepisanych z bazy:

MIT-BIH Arrhythmia Database - 106

Opis sygnału zawierał również informację o miejscu wystąpienia przedwczesnego skurczu komorowego (4min 23s), co było bardzo pomocne przy analizie.

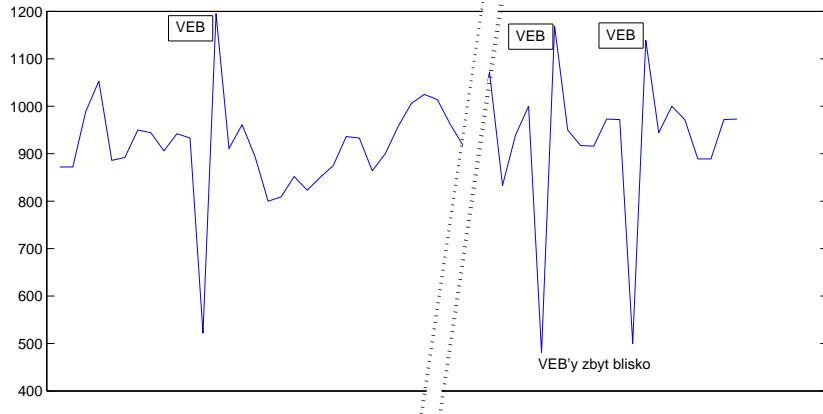
Wycinek sygnału, w którym zaobserwowano przedwczesny skurcz został zaprezentowany graficznie



Rysunek 1.80: Wykres sygnału EKG z oznaczonym VEB [mitdb/106 0:04:20-0:04:30]

Niestety podczas fazy testowej nie dysponowano pozostałymi modułami – R_PEAKS, QRS_CLASS. Przeprowadzenie testu wymagało ręcznego wprowadzenia ciągów próbek pików R (zadanie modułu R_PEAKS). Zbiór testowy został dobrany w taki sposób, aby sprawdzić najważniejsze funkcje algorytmu, tj. odnajdywanie VEB spełniającego podstawowe wymagania czasowe przedstawione w poprzednim punkcie. Odstępy czasowe pików R zostały zachowane, jednak nie przepisano całego ciągu, a jedynie połączono wybrane przedziały w których odnaleziono VEB analizując sygnał EKG. Utworzony ciąg próbek pików R został przedstawiony na tachogramie.

Na tachogramie zaznaczono kolejne skurcze, które powinny być wykryte w pierwszej fazie, ponieważ spełniają wymagania czasowe. Następnie jeśli wykryte skurcze są blisko siebie (drugi i trzeci) powinny zostać odrzucone z analizy, pierwszy VEB pozostanie. W ostatnim etapie filtracji danych sprawdzone zostaną kryteria dotyczące pojedynczych odstępów RR.

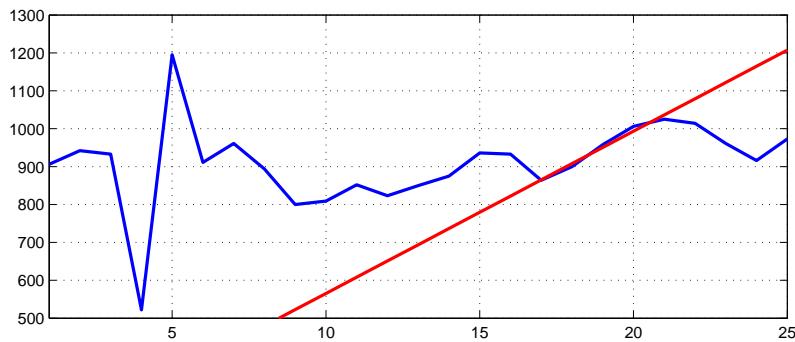


Rysunek 1.81: Tachogram z oznaczonymi VEB [mitdb/106 (0:04:23, 0:7:25, 0:15:21)]

Tablica 1.21: Uzyskane wyniki liczbowe

Całkowita liczba VEB	Liczba VEB po filtracji danych	TO	TS
9	3	0.16%	42.8ms/RR

Jako uzupełnienie wyników liczbowych przedstawiono również tachogram z zaznaczoną prostą definiującą współczynnik TS:



Rysunek 1.82: Wyjściowy tachogram

Analiza HRT to nowa i nie do końca poznana metoda, dlatego też jedyne co można powiedzieć o otrzymanych wynikach to: współczynnik TO jest za wysoki – dla zdrowego człowieka powinien być mniejszy od 0, współczynnik TS jest odpowiedni – większy od 2.5ms/RR. Na uśrednionym tachogramie można zauważać prawidłową pracę korekcyjną po przedwczesnym skurczu komorowym – początkowe przyspieszenie pracy serca, a następnie stopniowe spowalnianie.

W bazie danych nie odnaleziono danych referencyjnych do których można by było porównać otrzymane wyniki.

W efekcie końcowym moduł nie został połączony z QRS_CLASS- wyszukiwanie przedwcześnie skurczy komorowych odbywa się na podstawie danych z utworzonej listy RR.

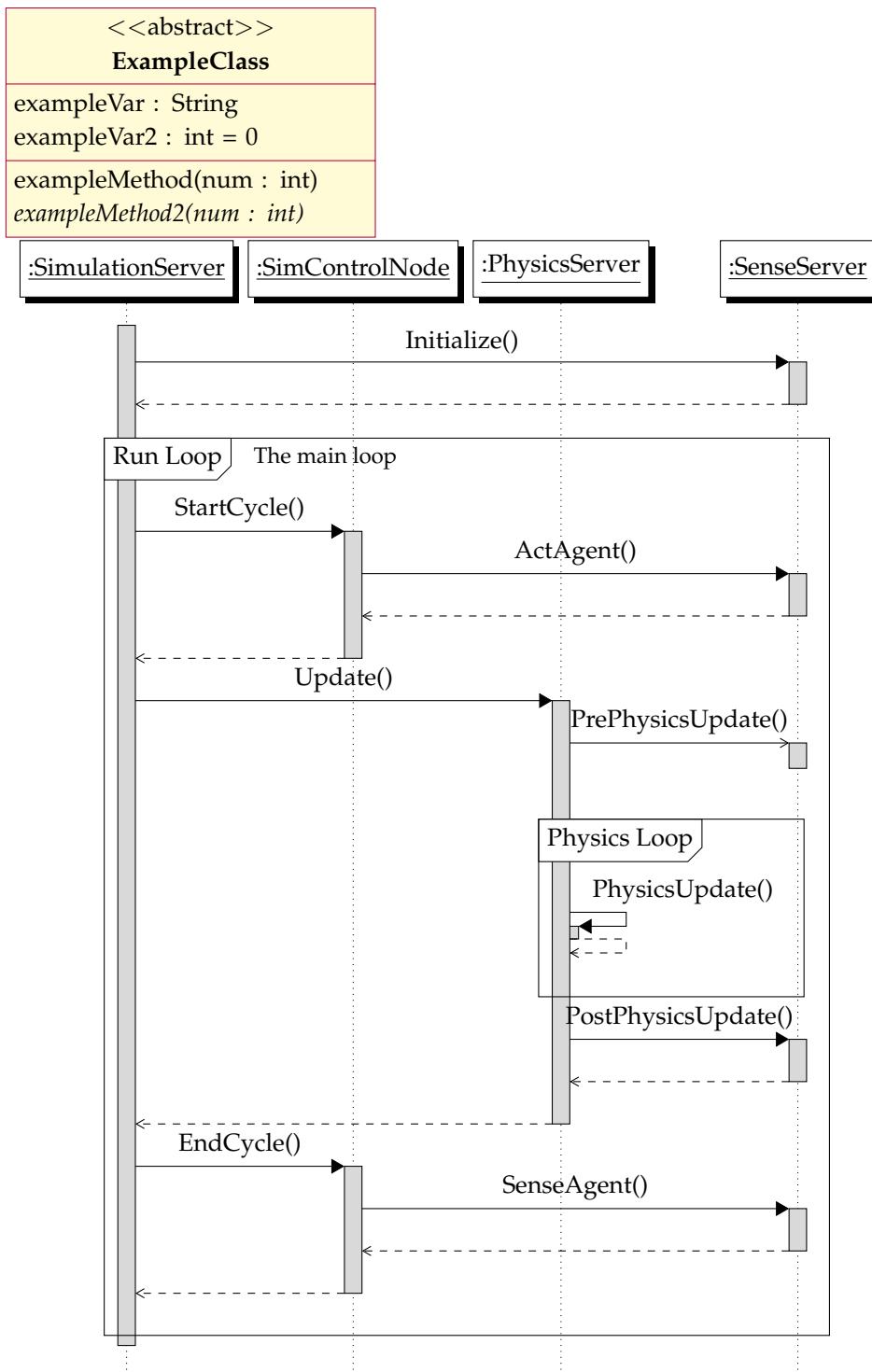
1.13.3.1 Literatura

Spis literatury związanej z modułem HRT

- Georg Schmidt, Raphael Schneider and Petra Barthel - Heart Rate Turbulence, Cardiac Electrophysiology Review 1999
- M. Zajac, J. Drożdż, M. Kurpesa, E. Trzos, T. Rechciński - Turbulencja rytmu serca jako metoda oceny ryzyka nagłego zgonu sercowego
- Heart Rate Turbulence: Standards of Measurement, Physiological Interpretation, and Clinical Use - 2008 by the American College of Cardiology Foundation <<<< HEAD

=====

>>> upstream/dev



Bibliografia

- [1] Medical dictionary. Dostępny w internecie: <http://medical-dictionary.thefreedictionary.com>.
- [2] Piotr Augustyniak. *Przetwarzanie Sygnałów Elektrodiagnostycznych*. 2001.
- [3] G. Hammerly C. Elkan. Learning the k in k-means. *Proceedings of the 17th annual conference on neural information processing systems (NIPS), Vancouver*, pages 281–288, 2003.
- [4] A. Adebisi et al. An exploratory study of k-means and expectation maximization algorithms. *British Journal of Mathematics and Computer Science*, pages 62–71, 2012.
- [5] A. Awal et al. Simplified mathematical model for generating ecg signal and fitting the model using nonlinear least square technique. *Proceedings of the International Conference on Mechanical Engineering, Dhaka*, 2011.
- [6] A. Illanes-Manriquez. et al. An algorithm for robust detection of qrs onset and offset in ecg signals. *Computers in Cardiology 2008*, pages 857–860, 2008.
- [7] A. Martinez et al. A new method for automatic delineation of ecg fiducial points based on the phasor transform. *Engineering in Medicine and Biology Society (EMBC), 2010 Annual International Conference of the IEEE*, pages 4586 – 4589, 2010.
- [8] B.-U. Kohler et al. The principles of software qrs detection. *IEEE Engineering in Medicine and Biology Magazine*, 1:42–57, 2002.
- [9] H.-K. Joeng et al. A new algorithm for p-wave detection in the ecg signal. *Engineering in Medicine and Biology Society, 1989. Images of the Twenty-First Century., Proceedings of the Annual International Conference of the IEEE Engineering in*, 1:42–43, 1989.
- [10] J. Gittinger et al. Unsupervised morphological classification of qrs complexes. *Computers in Cardiology*, pages 683–686, 1989.
- [11] M. Niknazar et al. Detection of characteristic points of ecg using quadratic spline wavelet transfrom. *Signals, Circuits and Systems (SCS), 2009 3rd International Conference on*, pages 1–6, 2009.
- [12] O. Escalona et al. A robust procedure for p-wave detection and segmentation in high resolution 12-lead ecg. *Engineering in Medicine and Biology Society, 1996. Bridging Disciplines for Biomedicine. Proceedings of the 18th Annual International Conference of the IEEE*, 4:1365 – 1366, 1996.
- [13] P. Laguna et al. A database for evaluation of algorithms for measurement of qt and other waveform intervals in the ecg. *Computers in Cardiology 1997*, pages 673–676, 1997.
- [14] Sz. Firlej et al. Projekt ekg-moduł waves, 2013.

- [15] Andrea Zoccola-Sara Mantero George B. Moody, Roger G. Mark. Derivation of respiratory signals from multi-lead ecgs. *Computers in Cardiology 1985, v.12*, 1985.
- [16] Zhijun Xiao Xianglin Yang-Xiaodong Zhang Jinzhong Song, Hong Yan. A robust and efficient algorithm for st-t complex detection in electrocardiograms. *Journal Of Mechanics In Medicine And Biology*, 2010.
- [17] B. Balonek K. Świerk. Raport z modułu obliczeniowego ECG_BASELINE. 2013.
- [18] Murtaza M. Lakdawala. Derivation of the respiratory rate signal from a single lead ecg. 2008.
- [19] Jinying Zhang Hun Li Li Shi, Cenyu Yang. Recognition of st segment of electrocardiogram based on wavelet transform. *Chinese Journal of Scientific Instrument*, 2008.
- [20] MT Johnson RJ Povinelli MA Mneimneh, EE Yaz. An Adaptive Kalman Filter for Removing Baseline Wandering in ECG Signals. *Computers in Cardiology*, 2006.
- [21] Seema Manpreet Kaur, Birmohan Singh. Comparisons of Different Approaches for Removal of Baseline Wander from ECG Signal. *International Conference and workshop on Emerging Trends in Technology (ICWET)*, 2011.
- [22] Laura Mason. Signal processing methods for non-invasive respiration monitoring. 2002.
- [23] MB Shamsollahi O Sayadi. ECG denoising and compression using a modified extended Kalman filter structure. *IEEE Transactions on biomedical engineering*, 55, 2008.
- [24] T. Pięciak. Elektrokardiografia (ekg).
- [25] Arun A. Balakrishnan Prasanth K., Baby Paul. Fetal ECG Extraction Using Adaptive Filters. *International Journal of Advanced Research in Electrical, Electronics and Instrumentation Engineering*, 2013.
- [26] C. Jutten R. Sameni, M.B Shamsollahi. Filtering Electrocardiogram Signals Using the Extended Kalman Filter. *Engineering in Medicine and Biology*, 2005.
- [27] C. Jutten M Babaie-Zadeh R. Sameni, M.B Shamsollahi. Filtering Noisy ECG Signals Using the Extended Kalman Filter Based on a Modified Dynamic ECG Model. *Computers in Cardiology*, 2005.
- [28] Nitish V. Thakor Pere Caminal Raimon Jane, Pablo Laguna. Adaptive Baseline Wander Removal in the ECG: Comparative Analysis With Cubic Spline Technique. *Computers in Cardiology*, 1992.
- [29] T. Jaśkiewicz T. Batko. Moduł obliczeniowy qrs_class. Projekt., 2013.
- [30] Jingsheng Liao Max Q.H Meng Zhao Shen, Chao Hu. An algorithm of st segment classification and detection. *IEEE International Conference on Automation and Logistics*, 2010.