



**KAUNO TECHNOLOGIJOS UNIVERSITETAS
FUNDAMENTALIŲJŲ MOKSLŲ FAKULTETAS
TAIKOMOSIOS MATEMATIKOS KATEDRA**

Jurgita Straukaitė

**KELIŲ ELEKTROKARDIOGRAMŲ
PARAMETRŲ VAIZDAVIMAS FAZINĖSE
PLOKŠTUMOSE**

Magistro darbas

**Vadovai
prof. Z. Navickas**

KAUNAS, 2010



KAUNO TECHNOLOGIJOS UNIVERSITETAS
FUNDAMENTALIŲJŲ MOKSLŲ FAKULTETAS
TAIKOMOSIOS MATEMATIKOS KATEDRA

TVIRTINU
Katedros vedėjas
doc. dr. N. Listopadskis
2010 06 05

KELIŲ ELEKTROKARDIOGRAMŲ
PARAMETRŲ VAIZDAVIMAS FAZINĖSE
PLOKŠTUMOSE

Taikomosios matematikos magistro baigiamasis darbas

Vadovas
Prof Z. Navickas
2010 06 01

Atliko
FMMM-8 gr. stud.
J. Straukaitė
2010 06 01

Recenzentas
dr. K. Berškienė
2010 06 01

KAUNAS, 2010

KVALIFIKACINĖ KOMISIJA

Pirmininkas: Leonas Saulis, profesorius (VGTU)

Sekretorius: Eimutis Valakevičius, docentas (KTU)

Nariai: Algimantas Jonas Aksomaitis, profesorius (KTU)

Vytautas Janilionis, docentas (KTU)

Vidmantas Povilas Pekarskas, profesorius (KTU)

Rimantas Rudzkis, habil. dr., vyriausiasis analitikas (DnB NORD Bankas)

Zenonas Navickas, profesorius (KTU)

Arūnas Barauskas, dr., vice-prezidentas projektams (UAB „Baltic Amadeus“)

Straukaitė J. Presentment of the several electrocardiogram's parameters in the phase planes: Master's work in applied mathematics/ supervisors prof. dr. Z. Navickas; Department of Applied Mathematics, Faculty of Fundamental Sciences, Kaunas University of Technology – Kaunas, 2010. – 70.

SUMMARY

For correct specification human's physiological state, it is very important to evaluate the changes of main human organism systems. In this paper, the parameters that characterize the function of periphery, regulation and supplying systems, JT and RR interval have been studied. Interpolation of discrete data from the physical load obtained by provocative incremental bicycle ergometry stress test was made by cubic spline.

The differences for various groups of person (sportsmen and patients with ischemic heart disease) were investigated. EKG parameters were applied in three methods: EKG parameters changes in speed computing and imaging in phase plane, EKG parameters smoothness level calculation, EKG parameters data grouping method.

The study revealed that the JT interval's changes in speed reliably differences between healthy and ill people. Also JT parameter's smoothness level and JT parameter's discriminants difference between the groups.

TURINYS

LENTELIŲ SĄRAŠAS	7
PAVEIKSLĖLIŲ SĄRAŠAS	8
IŽANGA	9
1 BENDROJI DALIS	10
1.1 FUNKCIJŲ INTERPOLIAVIMO SĄVOKA	10
1.2 LAGRANŽO INTERPOLIACINE IŠRAIŠKA	10
1.3 INTERPOLIAVIMAS TIESINE FUNKCIJA.....	11
1.4 INTERPOLIAVIMAS KUBINIŲ SPLAINŲ	11
1.5 SUGLODINIMAS	13
1.6 MAŽIAUSIŲ KVADRATŲ METODAS	14
1.7 DISKREČIOSIOS FURJÈ TRANSFORMACIJOS APIBRÈŽIMAS.....	15
1.8 BAIGTINIAI SKIRTUMAI	15
1.9 DVIEJŲ LAIKO EILUČIŲ DINAMINIŲ SĄSAJŲ SKAIČIAVIMAS	16
1.10 PAGRINDINIAI PARAMETRAI.....	17
1.11 AMŽIAUS ĮTAKA ŠIRDIES IR KRAUJAGYSLIŲ SISTEMOS REGULIAVIMUI	19
1.12 ŠIRDIES IR KRAUJAGYSLIŲ SISTEMOS FUNKCINIŲ RODMENŲ KAITA, ATLIEKANT FIZINIO KRŪVIO MÈGINIUS.....	20
1.13 IŠEMINÈ ŠIRDIES LIGA	23
1.14 NAUDOTA METODIKA.....	25
2 TIRIAMOJI DALIS	26
2.1 DUOMENYS	26
2.2 TYRIMAS	26
2.3 EKG PARAMETRŲ KITIMO GREIČIO SKAIČIAVIMAS IR VAIZDAVIMAS FAZINÈJE PLOKŠTUMUJE.....	28
2.4 EKG PARAMETRO GLODUMO KOEFICIENTO SKAIČIAVIMAS IR VAIZDAVIMAS 29	
2.5 EKG PARAMETRO DUOMENŲ GRUPAVIMAS	31
2.6 PROGRAMINÈS ĮRANGOS PASIRINKIMO PAGRINDAS IR INSTRUKCIJA VARTOTOJUI	34
DISKUSIJA	37
PADÈKA	38
IŠVADOS	39

LITERATŪRA.....	40
1 PRIEDAS. VIENO ASMENS REZULTATAI.....	42
2 PRIEDAS. NEPARAMETRINIS MANIO-VITNIO-VILKOKSONO TESTAS	45
3 PRIEDAS. PROGRAMOS TEKSTAS	47

LENTELIŲ SĄRAŠAS

1.1 Lentelė.....	16
1.2 lentelė.....	19
2.1 lentelė.....	26
2.2 lentelė.....	27

PAVEIKSLŲ SĄRAŠAS

1. 1 pav. Taškų jungimas laužtėmis.....	11
1. 3 pav. JT intervalo grafinis vaizdas	18
1. 4 pav. Kraujo tėkmės (ml/min) persiskirstymas įvairiuose organuose.....	20
1. 5 pav. Žmogaus organizmo funkcinės būklės įvertinimo modelis krūvio metu	22
1. 6 pav. Integralinio vertinimo modelis ir organizmo sistemų tarpusavio ryšius rodantys žymenys (A. Vainoras, 1996), V - vykdančioji sistema, R - reguliacinė sistema, A – aprūpinančioji sistema	22
2. 1 pav. Turimų reikšmių interpoliavimas kubiniu splineu	28
2. 2 pav. Padidintas grafiko vaizdas	28
2.3. pav. Ligonio fazinių plokštumų ir jų kitimo greičių paveikslai	29
2.4. pav. Glodumo koeficientų paveikslas.....	31
2.5. pav. Sportininko 20-ies porų diskriminantų paveikslas;.....	32
2.6. pav. Ligonio 20-ies porų diskriminantų paveikslas;.....	32
2.7. pav. Sportininko logaritminė išvestinės grafikas.....	33
2.8. pav. Ligonio logaritminė išvestinės grafikas.....	33
2.9. pav. Programos langas	34
2.10. pav. Programos langas	35

IŽANGA

Pasaulio pažanga nepaliaujamai juda į priekį. Dabar žmogaus nestebina buityje atsirandantys nauji technologijos stebuklai, palengvinantys buitį. Tačiau, kad ir kokia gerą materialinę gerovę mums sukuria modernėjantis pasaulis, tai anaipol dar neužtikrina laimingo ir gražaus bei ilgo gyvenimo. Didžiausias mūsų turtas, kaip senoliai mums vis primena, yra sveikata.

Praėjusio amžiaus aštuntojo dešimtmečio pabaigoje bei per visą devintąjį dešimtmetį labai paplito įvairios fizinio aktyvumo formos. Susikūrė ir veikia daug sportinių ar bendrojo fizinio aktyvumo klubų, kuriuos lanko daug žmonių. Plintant šiems judėjimams, kyla vis daugiau problemų, susijusių su fizinio krūvio adekvatumu asmens būklei. Tikėdami, jog fizinis aktyvumas yra vaistas nuo visų ligų, praktiškai nederindami jo sunkumo su sveikatos būkle, žmonės dažnai pajunta, jog sveikatos problemos paaštrėja. Statistiškai pagrįstas tvirtinimas, jog fiziškai neaktyvūs asmenys serga dažniau ar jų amžius yra trumpesnis, nereiškia, jog bet koks fizinis aktyvumas šį sergamumą sumažina. Padėti gali tik sveikatos būklę atitinkantis fizinis aktyvumas, t. y. individualus fizinis krūvis. Todėl tik tinkamai parinktas ir atitinkamai dozuojamas fizinis krūvis gali sumažinti ligos sukeltų simptomų intensyvumą ar neleisti jiems atsirasti. Dėl to mokslininkai ieško būdų, kaip kuo anksčiau pastebėti ligų užuomazgas ir jas pašalinti ligai dar neįsigalėjus žmogaus organizme. Ir vis dažniau galime aptikti darbų, kuriose bandoma ankstyvas širdies kraujagyslių susirgimų problemas aptikti naudojant veloergometrinių pakopomis didėjančio krūvio metodiką.

Šiame darbe tyrėme veloergometrinių mėginių metu užregistruotus elektrokardiogramos parametrus, kurie gali atspindėti organizmo būklę: atskirai reguliacinę, aprūpinančiąją ir vykdančiąją sistemas. Nors šie parametrai ir pakankamai informatyvūs, tačiau dažnai esant nedideliems širdies ir kraujagyslių sutrikimams jokių pokyčių neparodo. Todėl reikalingi nauji modernūs analizės metodai, kurie padėtų atskleisti kokybiškai naują informaciją glūdinčią EKG parametruose ir jų kaitoje. Tyrėme parametrų dinamiką fazinėje plokštumoje panaudodami elektrokardiogramos parametrų duomenis krūvio bei atsigavimo metu. Sukurta taikomoji programa, kuri parašyta „Matlab“ aplinkoje. Ši programa grupuoja duomenis, atlieka reikiamus skaičiavimus ir rezultatus, įvairius grafikus, pateikia vartotojui ekrane.

BENDROJI DALIS

FUNKCIJŲ INTERPOLIAVIMO SĄVOKA

Gamtos moksluose ir technikoje argumentas ir funkcijos reikšmė nustatoma stebėjimais. Stebėtojas tam tikrais laiko momentais išmatuoja ir užrašo argumento x_i , ir funkcijos $y=f(x_i)$ reikšmes. Duomenys surašomi į lentelę, kuria išreiškiama funkcijos priklausomybė nuo argumento atskiruose taškuose. Gautoji lentelė naudojama teoriniam bei praktiniam reiškinių tyrimui.

Apibrėžimas. Funkcijos $f(x)$ duotos reikšmių lentelė, išreiškimas analizine formule vadinamas interpoliavimu, jei reikšmės, gautos pagal formulę taškuose x_i , sutampa su funkcijos reikšmėmis lentelėje.

Apskritai interpoliavimo sąvoka yra platesnė.

Tarkime duota funkcijos $y=f(x)$ reikšmių lentelė (x_i, y_i) ; čia $i=0, 1, \dots, n$, y_i , reikšmės yra tikslios arba paklaidos tokios mažos, kad praktiškai jų galima nepaisyti. Reikia rasti aproksimuojančią funkciją $y=F(x)$, priklausančią funkcijų klasei K ir tenkinančią sąlygas:

$$F(x_i)=y_i, \quad i=0, 1, \dots, n \quad (1.1)$$

Šios sąlygos vadinamos Lagranžo interpoliavimo sąlygomis, o pati funkcija $y=F(x)$ – Lagranžo interpoliacinė funkcija, arba tiesiog interpoliacinė funkcija [10].

LAGRANŽO INTERPOLIACINĖ IŠRAIŠKA

Ieškosime n -tos eilės daugianario $F(x)$ tenkinančio Lagranžo interpoliavimo sąlygas (1.1) formulė. Išnagrinėkime polinomus:

$$F_i(x) = \frac{(x-x_0)\dots(x-x_{i-1})(x-x_{i+1})\dots(x-x_n)}{(x_i-x_0)\dots(x_i-x_{i-1})(x_i-x_{i+1})\dots(x_i-x_n)}, \quad i = \overline{0, n}.$$

$$\text{Jie turi tokią savybę: } F_i(x_j) = \begin{cases} 1, & \text{kai } i = j, \\ 0, & \text{kai } i \neq j; \end{cases} \quad j \in \{0, 1, \dots, n\}.$$

Pasinaudodami polinomis $F_i(x)$, galime iš karto parašyti interpoliacinio polinomo $F(x)$ išraišką:

$$F(x) = \sum_{i=0}^n F_i(x) y_i. \quad (1.2)$$

Šis polinomas interpoliacinis, nes jis yra n -tos eilės polinomas ir tenkina Lagranžo interpoliavimo sąlygas. (1.2) interpoliacinio polinomo išraiška vadinama Lagranžo interpoliacine išraiška, todėl literatūroje simbolis $F(x)$ keičiamas $L_n(x)$ [10].

INTERPOLIAVIMAS TIESINE FUNKCIJA

Sakykime turime taškus x_0, x_1, \dots, x_n ir funkcijos $y=f(x)$ reikšmes $f(x_0), f(x_1), \dots, f(x_n)$. Iš visų interpoliacinių polinomų paprasčiausias yra pirmojo laipsnio interpoliacinis polinomas. Jį apskaičiuojame, naudodamiesi dviem argumento ir funkcijos reikšmėmis.

Užrašykime Lagranžo interpoliacinį polinomą:

$$L_n(x) = \sum_{k=0}^n \frac{(x-x_0)\dots(x-x_{k-1})(x-x_{k+1})\dots(x-x_n)}{(x_k-x_0)\dots(x_k-x_{k-1})(x_k-x_{k+1})\dots(x_k-x_n)} f(x_k)$$

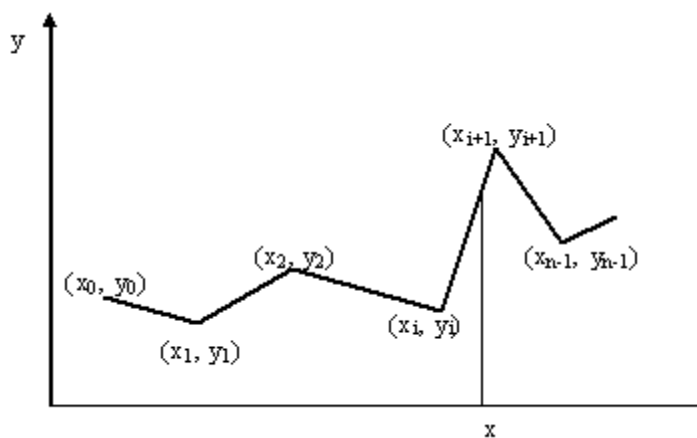
Jei $f(x_i)$ ir $f(x_{i+1})$ yra dvi minėtos funkcijos reikšmės, tai paėmę $n=1$, gauname:

$$L_1(x) = \frac{x-x_{i+1}}{x_i-x_{i+1}} f(x_i) + \frac{x-x_i}{x_{i+1}-x_i} f(x_{i+1}) = f(x_i) + \frac{f(x_{i+1})-f(x_i)}{x_{i+1}-x_i}(x-x_i)$$

Tokiu būdu, pirmojo laipsnio interpoliacinis polinomas funkcijai, kurios reikšmės žinomos $n+1$ taškuose x_0, x_1, \dots, x_n išreiškiamas formule:

$$L_1(x) = f(x_i) + \frac{f(x_{i+1})-f(x_i)}{x_{i+1}-x_i}(x-x_i), \text{ jei } x_i \leq x \leq x_{i+1} \text{ ir } (i = 0, 1, 2, \dots, n-1).$$

Interpoliavimas pirmojo laipsnio polinonais vadinamas tiesiniu interpoliavimu. Tiesinio interpoliavimo formulės (polinomo) grafikas yra laužtė, jungianti taškus $(x_i, f(x_i))$, $(i = 0, 1, \dots, n)$ (1.1 pav.) [10].



1. 1 pav. Taškų jungimas laužtėmis

INTERPOLIAVIMAS KUBINIŲ SPLAINU

Splainas – tai tolydžioji iki p -tosios eilės išvestinės imtinai funkcija, sudaryta iš kurios nors funkcijos dalių. Istoriskai splainai pradėti konstruoti iš n -tosios eilės polinomo dalių, todėl čia, kalbėdami apie splainus, laikysime juos funkcijomis, sudarytomis iš polinomo dalių. Praktikoje dažniausiai naudojami kubiniai splainai.

Uždavinio formulotė. Duota funkcijos $y = f(x)$ reikšmių lentelė (x_i, y_i) , $i = \overline{0, N}$. Reikia rasti kubinį splainą $y = g(x)$, tenkinantį Lagranžo interpoliavimo sąlygą

$$g(x_i) = y_i, \quad i = \overline{0, N}.$$

Kad interpoliacinis kubinis splainas būtų apibrėžtas vienareikšmiškai, pasirenkamos kraštinės sąlygos

$$g''(x_0) = g''(x_N) = 0.$$

Jos dar vadinamos natūraliomis kraštinėmis sąlygomis ir rodo, kad plieninės liniuotės, išraitytos per interpoliavimo taškus, galai paliekami laisvai kaboti.

Kadangi splaino antroji išvestinė yra tolydžioji ir tiesinė kiekviename intervale $[x_{i-1}, x_i]$, ($i = \overline{1, N}$), tai galima rašyti:

$$g''(x) = m_{i-1} \frac{x_i - x}{h_i} + m_i \frac{x - x_{i-1}}{h_i}; \quad (1.3)$$

čia $m_i = g''(x_i)$, $i = \overline{0, N}$, o $h_i = x_i - x_{i-1}$.

(1.3) lygybę suintegravę du kartus, gauname:

$$g(x) = m_{i-1} \frac{(x_i - x)^3}{6h_i} + m_i \frac{(x - x_{i-1})^3}{6h_i} + A_i \frac{x_i - x}{h_i} + B_i \frac{x - x_{i-1}}{h_i}; \quad (1.4)$$

čia A_i, B_i - integravimo konstantos.

A_i, B_i parinkime taip, kad $g(x)$ tenkintų interpoliavimo sąlygą, t.y. kad būtų

$$g(x_{i-1}) = y_{i-1} \text{ ir } g(x_i) = y_i.$$

Iš (1.4) lygybės gauname:

$$A_i = y_{i-1} - \frac{m_{i-1}h_i^2}{6}, \quad B_i = y_{i-1} - \frac{m_i h_i^2}{6}.$$

Vadinasi, kubinio splaino išraiška intervale $[x_{i-1}, x_i]$ yra tokia:

$$g(x) = m_{i-1} \frac{(x_i - x)^3}{6h_i} + m_i \frac{(x - x_{i-1})^3}{6h_i} + \left(y_{i-1} - \frac{m_{i-1}h_i^2}{6} \right) \frac{x_i - x}{h_i} + \left(y_{i-1} - \frac{m_i h_i^2}{6} \right) \frac{x - x_{i-1}}{h_i}. \quad (1.5)$$

(1.5) formulės nežinomi dydžiai m_i yra splaino antrųjų išvestinių reikšmės taškuose x_i . Jas apskaičiuojame remdamiesi sąlyga

$$g'(x_i - 0) = g'(x_i + 0) \quad (i = \overline{1, N-1}) \quad (1.6)$$

ir tardami, kad $m_0 - m_N = 0$, t.y. pasirinkdami natūralias kraštines sąlygas.

Nesunku įsitikinti,

$$g'(x_i - 0) = \frac{h_i}{6} m_{i-1} + \frac{h_i}{3} m_i + \frac{y_i - y_{i-1}}{h_i},$$

$$g'(x_i + 0) = -\frac{h_{i+1}}{3} m_i - \frac{h_{i+1}}{6} m_{i+1} + \frac{y_{i+1} - y_i}{h_{i+1}}.$$

Tada (1.6) lygčių sistemą galima užrašyti matricine išraiška:

$$Am = Hy; \quad (1.7)$$

čia A – simetrinė kvadratinė $(N-1)$ -tosios eilės trijųstrižainė matrica

$$A = \begin{bmatrix} \frac{h_1 + h_2}{3} & \frac{h_2}{6} & 0 & \dots & 0 & 0 \\ \frac{h_2}{6} & \frac{h_2 + h_3}{3} & \frac{h_3}{6} & \dots & 0 & 0 \\ 0 & \frac{h_3}{6} & \frac{h_3 + h_4}{3} & \dots & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & \dots & \frac{h_{N-1}}{6} & \frac{h_{N-1} + h_N}{3} \end{bmatrix},$$

H – stačiakampė (N-1)x(N+1) matrica

$$H = \begin{bmatrix} \frac{1}{h_1} & -\left(\frac{1}{h_1} + \frac{1}{h_2}\right) & \frac{1}{h_2} & 0 & \dots & 0 & 0 & 0 \\ 0 & \frac{1}{h_2} & -\left(\frac{1}{h_2} + \frac{1}{h_3}\right) & \frac{1}{h_3} & \dots & 0 & 0 & 0 \\ \dots & \dots & \dots & \dots & \dots & \dots & \dots & \dots \\ 0 & 0 & 0 & 0 & \dots & \frac{1}{h_{N-1}} & -\left(\frac{1}{h_{N-1}} + \frac{1}{h_N}\right) & \frac{1}{h_N} \end{bmatrix},$$

m, y – matricos stulpeliai

$$m = \begin{bmatrix} m_1 \\ m_2 \\ \dots \\ m_{N-1} \end{bmatrix}, \quad y = \begin{bmatrix} y_0 \\ y_1 \\ \dots \\ y_N \end{bmatrix}.$$

(1.7) sistema yra tiesinių lygčių sistema su trijstrižaine matrica A , kurios pagrindinės įstrižainės elementai didesni už kitų eilutės elementų sumą. Taigi ši sistema yra suderintoji ir turi vienintelį sprendinį [10].

SUGLODINIMAS

Nagrinėdami interpoliavimo uždavinį, funkciją $f(x)$, apibrėžtą reikšmių lentelę, stengėmės pakeisti tokia aproksimuojančiąja funkcija $F(x)$ (n-tojo laipsnio polinomu, splineu ir pan.), kad taškuose x_i ($i = \overline{0, n}$) $F(x_i)$ reikšmės būtų lygios $f(x_i)$ reikšmėms.

Labai dažnai $f(x_i)$ reikšmės yra eksperimento rezultatai ir turi matavimo bei metodo paklaidų. Todėl reikalauti, kad aproksimuojančioji funkcija $F(x)$ tenkintų sąlygą $F(x_i) = f(x_i)$ ($i = \overline{0, n}$), būtų neprotinga. Geriau rasti tokią funkciją $F(x)$, kuri pagal pasirinktą kriterijų geriausiai aproksimuotų $f(x)$. Toks funkcijos $F(x)$ apskaičiavimo metodas vadinamas suglodinimu. Atsižvelgiant į suglodinimo kriterijų, galima gauti įvairias $F(x)$ išraiškas.

Suglodinimo uždavinio formulavimas. Sakykime, funkcija $y = f(x)$ nusakyta reikšmių lentelė (x_i, \tilde{y}_i) ; čia $i = \overline{1, m}$. Simbolis \tilde{y}_i rodo, kad $f(x)$ reikšmės taškuose x_i yra apytikslės. Taip pat žinoma

aprosimuojančiosios funkcijos $F(x)$ analizinė išraiška: $F(x, a_0, \dots, a_n)$; a_i ($i = \overline{0, n}$) – nežinomi parametrai ir $n \ll m$. Reikia rasti tokias parametru a_i reikšmes, su kuriomis $F(x)$ geriausiai aproksimuotų funkciją $f(x)$.

Sprendžiant šį uždavinį, taikomi įvairūs suglodinimo kriterijai, o kartu ir įvairūs suglodinimo metodai [10].

MAŽIAUSIŲ KVADRATŲ METODAS

Mažiausių kvadratų metodas tai dažniausiai taikomas suglodinimo metodas. Jis formuluojamas taip: koeficientus a_k ($k = \overline{0, n}$) reikia apskaičiuoti taip, kad $f(x_i)$ ir $F(x_i)$ skirtumų kvadratų suma būtų pati mažiausia, t.y. reikia minimizuoti

$$z = \sum_{i=1}^m (F(x_i, a_0, \dots, a_n) - \tilde{y}_i)^2. \quad (1.8)$$

(1.8) Formule nusakyta tikslo funkcija turi vienintelį ekstremumą, kuris apskaičiuojamas iš lygčių sistemos

$$\frac{\partial z}{\partial a_k} = 0, \quad k = \overline{0, n}. \quad (1.9)$$

Bendruoju atveju ši sistema yra netiesinė, taigi ją galima spręsti taikant netiesinių lygčių sistemų sprendimo metodus, išnagrinėtus trečiajame skyriuje.

(1.9) sistemos formavimą ir sprendimą galima palengvinti dvejopai:

1) Jei $F(x, a_0, \dots, a_n)$ yra n -tojo laipsnio polinomas, tai (1.9) lygčių sistema yra tiesinė ir jos formavimas bei sprendimas nesudaro sunkumą;

2) Galima taikyti vadinamąjį ištiesinimo metodą, kurio esmė tokia: atitinkamai parinktoje koordinačių sistemoje (X, Y) taškai $(x_i; \tilde{y}_i)$ (čia $i = \overline{1, m}$) apytiksliai tenkina tiesės $Y = kX + b$ lygtį, tada mažiausių kvadratų metodas toje sistemoje realizuojamas labai paprastai.

Pavyzdžiui, tarkime, kad taškai $(x_i; \tilde{y}_i)$ tenkina dėsnį

$$y = ae^{cx}. \quad (1.10)$$

Logaritmuodami abi šios lygybės puses, gauname: $\ln y = \ln a + cx$.

Pažymėkime: $X = x$, $Y = \ln y$, $b = \ln a$. Tada (1.10) lygtį koordinačių sistemoje (X, Y) galėsime užrašyti taip: $Y = cX + b$.

Ištiesinimo metodas gali būti taikomas, norint nustatyti funkcijos $F(x, a_0, \dots, a_n)$ analizinę išraišką, t.y. sužinoti, kokį dėsnį tenkina taškai $(x_i; \tilde{y}_i)$. Šio metodo esmė yra ta, kad turėdami testinį koordinačių sistemų (X, Y) rinkinį, ieškome, kurioje koordinačių sistemoje taškai $(x_i; \tilde{y}_i)$ apytiksliai tenkina tiesės $Y = cX + b$ lygtį [10].

DISKREČIOSIOS FURJĖ TRANSFORMACIJOS APIBRĖŽIMAS

Imkime duomenų vektorių $X = [X(m)]$, kurio elementai $X(m)$, $m = 0, 1, \dots, N-1$, yra realieji arba kompleksiniai skaičiai. Diskrečioji Furjė transformacija vektoriui $[X(m)]$ apibrėžiama taip:

$$\begin{aligned} C_x(k) &= \frac{1}{N} \sum_{m=0}^{N-1} X(m) W^{km} = \\ &= \frac{1}{N} (X(0) + X(1)W + X(2)W^{2k} + \dots + X(N-1)W^{k(N-1)}), \end{aligned} \quad (1.11)$$

čia $W = e^{i\frac{2\pi}{N}}$, $i^2 = -1$, $k = 0, 1, \dots, N-1$.

Lengva parodyti, kad rodiklinės funkcijos W^{km} yra ortogonaliosios, tai yra tenkina sąlygą

$$\sum_{m=0}^{N-1} W^{km} \cdot W^{-lm} = \begin{cases} N, & \text{kai } (k-l) \text{ lygus } 0 \text{ arba} \\ & \text{yra kartotinis skaičiui } N, \\ 0, & \text{kitais atvejais.} \end{cases} \quad (1.12)$$

Daugindami (1.11) lygybės abi puses iš W^{-km} ir sumuodami pagal k (nuo 0 iki $N-1$), gauname

$$\begin{aligned} \sum_{k=0}^{N-1} C_x(k) W^{-km} &= \frac{1}{N} \sum_{k=0}^{N-1} W^{-km} (X(0) + X(1)W^k + \dots + \\ &+ X(m)W^{km} + \dots + X(N-1)W^{k(N-1)}) \end{aligned}$$

Pastaroji lygybė, įvertinus ortogonalumo sąlygą (1.12), leidžia užrašyti atvirkštinę diskrečiąją Furjė transformaciją:

$$X(m) = \sum_{k=0}^{N-1} C_x(k) W^{-km}, \quad m = 0, 1, \dots, N-1. \quad (1.13)$$

Kadangi (1.11) ir (1.13) išraiškos sudaro transformacijų porą, tai duomenų vektoriaus $[X(m)]$ vaizdavimas, panaudojant rodiklines funkcijas (t.y. pervedant vektorių į spektrinę sritį) yra vienintelis [13].

BAIGTINIAI SKIRTUMAI

Jei atstumai tarp interpoliavimo taškų vienodi patogiu naudotis vadinamaisiais baigtiniais skirtumais. Skirtumai

$$\Delta f(x_i) = f(x_{i+1}) - f(x_i) \quad (i = 0, 1, 2, \dots, n-1)$$

vadinami pirmos eilės baigtiniais skirtumais.

Analogiškai, k -tos eilės baigtiniais skirtumais vadinami dydžiai

$$\Delta^k f(x_i) = \Delta^{k-1} f(x_{i+1}) - \Delta^{k-1} f(x_i) \quad (k = 1, 2, \dots) \quad (1.14)$$

Baigtiniai skirtumai surašomi į tokią lentelę:

0.1 Lentelė

Baigtinių skirtumų užrašymas

x_i	$f(x_i)$	$\Delta f(x_i)$	$\Delta^2 f(x_i)$	$\Delta^3 f(x_i)$...
x_0	$f(x_0)$				
x_1	$f(x_1)$	$\Delta f(x_0)$			
x_2	$f(x_2)$	$\Delta f(x_1)$	$\Delta^2 f(x_0)$		
x_4	$f(x_4)$	$\Delta f(x_2)$	$\Delta^2 f(x_1)$	$\Delta^3 f(x_0)$	
...

Dabar išstirsime baigtinių skirtumų savybes.

Lema. k -tos eilės baigtinių skirtumų suma yra lygi skirtumui tarp paskutiniojo ir pirmojo $(k - 1)$ – tos eilės baigtinių skirtumų:

$$\sum_{i=0}^{n-k} \Delta^k f(x_i) = \Delta^{k-1} f(x_{n-k+1}) - \Delta^{k-1} f(x_0).$$

Uždavinio formulavimas. Sakykime, funkcija $y = f(x)$ nusakyta reikšmių lentele (x_i, \tilde{y}_i) ; čia $i = 1, \dots, m$. Symbolis \tilde{y}_i rodo, kad $f(x)$ reikšmės taškuose x_i yra apytikslės. Taip pat žinoma aproksimuojančiosios funkcijos $F(x)$ analizinė išraiška: $F(x, a_0, \dots, a_n)$; čia a_i ($i = 0, \dots, n$) – nežinomi parametrai ir $n \ll m$. Reikia rasti tokias parametru a_i reikšmes, su kuriomis $F(x)$ geriausiai aproksimuotų funkciją $f(x)$. Sprendžiant šį uždavinį, taikomi įvairūs suglodinimo metodai. Vienas jų – mažiausių kvadratų metodas [5].

DVIEJŲ LAIKO EILUČIŲ DINAMINIŲ ŠAŠAJŲ SKAIČIAVIMAS

Sakykite, turime du vienalaikius (synchroninius) signalus $(x_n; n = 0, 1, 2, \dots)$ ir $(y_n; n = 0, 1, 2, \dots)$.

Tada signalai kointegruojami, iš jų sudarant matricų seką $(A_n; n = 0, 1, 2, \dots)$:

$$A_n := \begin{pmatrix} a_n & b_n \\ c_n & d_n \end{pmatrix}; \quad (1.15)$$

čia $a_n := x_n$, $b_n := x_{n+1} - y_{n+1}$, $c_n := y_n$, $d_n := x_{n-1} - y_{n-1}$.

Šių matricų sekos tyrimui naudojamos pagrindinės antros eilės matricų skaitinės charakteristikos ir matricų A_n esminės komponentės:

$$1. \text{ Matricos } A_n \text{ pėdsakas } \text{Tr}A_n := a_n + d_n; \quad (1.16)$$

$$2. \text{ Matricos } A_n \text{ skirtumas pėdsakas } \text{dfr}A_n := a_n - d_n; \quad (1.17)$$

$$3. \text{ Matricos } A_n \text{ kodiagonalinė sandauga } \text{cdp}A_n := b_n \cdot c_n; \quad (1.18)$$

$$4. \text{ Matricos } A_n \text{ esminė komponentė } B_n := \begin{pmatrix} \frac{\text{dfr}A_n}{2} & b_n \\ c_n & -\frac{\text{dfr}A_n}{2} \end{pmatrix}; \quad (1.19)$$

$$5. \text{ Matricos } A_n \text{ diskriminantas } \text{dsk} A_n = (\text{dfr}A_n)^2 + 4\text{cdp} A_n; \quad (1.20)$$

$$6. \text{ Matricos } A_n \text{ determinantas } \det A_n = \frac{1}{4}((\text{Tr}A_n)^2 - \text{dsk} A_n) \quad (1.21)$$

Matricų analizės teorijoje išskiriamos dvi svarbios matricų rūšys. Matricą I vadiname *idempotentu* (pastovios galios matrica), jeigu $I^2 = I$, o matricą N – *nulpotentu* (matrica, netenkančia galios), jeigu $N^2 = \mathbf{0}$, kai $\mathbf{0} := \begin{pmatrix} 0 & 0 \\ 0 & 0 \end{pmatrix}$. Esminė komponentė B_n yra nulpotentas, jei $\text{dsk}A_n = 0$, o

matricos $I_n := \frac{1}{2}E + \frac{1}{\sqrt{\text{dsk} A_n}}B_n$, bei $E - I_n$ - idempotentai, jei tik $\text{dsk}A_n \neq 0$.

Iš pateiktų sąryšių akivaizdu, kad nagrinėjant laiko eilučių dinamines sąsajas svarbūs yra matricų A_n diskriminantai, kurie pasižymi invariantine savybe: jeigu T bet kokia antros eilės matrica, kurios $\det T \neq 0$, tada egzistuoja atvirkštinė matrica T^{-1} ir yra teisingas toks sąryšis: $\text{dsk}(TA_nT^{-1}) = \text{dsk}A_n$, t.y. matricai A_n panašių matricų TA_nT^{-1} diskriminantai yra tokie patys kaip ir matricos A_n .

Remiantis visais nurodytais sąryšiais, galima suformuluoti išvadą - jeigu matricų A_n diskriminantai artėja prie nulio, tai matricos A_n iš idempotentinių matricų virsta nulpotentinėmis, kas rodo, kad duotosios laiko eilutės $(x_n; n = 0,1,2,\dots)$ ir $(y_n; n = 0,1,2,\dots)$ „panašėja“, mažėja jų individualus informatyvumas, o tai reiškia, kad jie aprašo vis labiau susijusias sistemas [6].

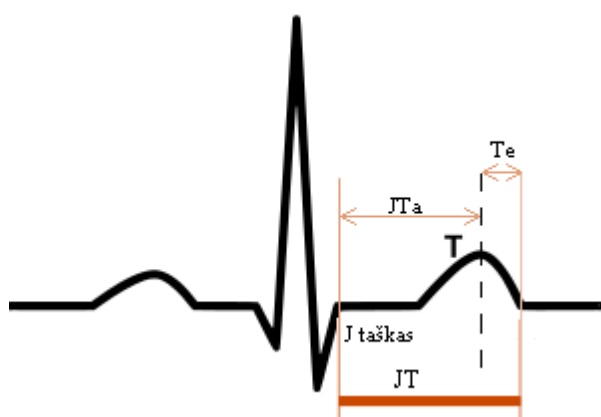
PAGRINDINIAI PARAMETRAI

ŠSD - širdies susitraukimų dažnis. Parametro kitimo ribos 20 - 250 kartų per minutę. Fizinio krūvio metu ŠSD didėja todėl, kad intensyvių širdies darbą, nes būtina patenkinti dirbančių raumenų poreikius. Po fizinio krūvio ŠSD ne iš karto grįžta į pradinį lygį. Atlikus mažo intensyvumo fizinį krūvį, ŠSD greitai mažėja iki lygio, buvusio ramybės būsenoje. Atliekant didelio intensyvumo fizinius pratimus, ŠSD pokyčius galima suskirstyti į dvi fazes: greito eksponentiško mažėjimo ir lėto mažėjimo iki lygio, buvusio ramybės būsenoje. Manoma, kad ilgalaikių treniruočių metu ŠSD mažėja dėl kelių veiksnių. Tai priklauso nuo pratimų intensyvumo, jų apimties, trukmės, pertraukų tarp pratimų bei pratime dalyvaujančių raumenų kiekio. Kol kas mažai išnagrinėti ŠSD pokyčių fiziologiniai mechanizmai, taikant ilgalaikes fizines treniruotes.

JT - intervalas elektrokardiogramoje nuo jungties taško J iki T bangos pabaigos (JT intervalas). Parametro kitimo ribos 0,14 - 0,36 s. Fizinio krūvio metu šis intervalas trumpėja. JT intervalas apibūdina skilvelių repoliarizacijos trukmę. JT intervalas lemia skirtingus elektrofiziologinius

reiškinius ir yra skirstomas į JTa intervalą (nuo taško J iki T viršūnės) bei Te intervalą (nuo T bangos viršūnės iki T pabaigos) (1.1 pav.). JT intervalo pokyčiams įtakos turi reguliacinė nervų sistema. Yra žinoma, kad organizmo metaboliniai pokyčiai yra susiję su repoliarizacijos pokyčiais. Derivacijos, kuriose JT intervalas trumpesnis, rodo, kad tose miokardo zonose vyksta ankstyvesnė repoliarizacija, greitesni metaboliniai pokyčiai, o ten, kur JT intervalas ilgesnis, rodo lėtesnę repoliarizaciją bei lėtesnes metaboliškas reakcijas. Nustatyta, kad JT intervalas sveikiems žmonėms fizinio krūvio metu trumpėja iki 140 ms, tuo tarpu išemine širdies liga sergantiems JT intervalas krūvio metu sutrumpėja žymiai mažiau.

Tarkim, kad skaliarinė laiko eilutė apibrėžta kaip x_1, x_2, \dots, x_N . Tada laiko vėlinimo vektorius gali būti rekonstruotas remiantis tokiu principu:



1. 2 pav. JT intervalo grafinis vaizdas

N – galingumas, matuojamas W.

N_{\max} – tyrimo metu pasiektas maksimalus galingumas, W.

RR – laiko intervalas tarp dviejų širdies susitraukimų. Intervalas apskaičiuojamas tokiu būdu (indeksas i nurodo, kad parametrai skaičiuojami esant tam pačiam galingumui N):

$$RR_i = \frac{60}{\check{SSD}_i} \quad (1.22)$$

JT/RR - parametras, siejantis reguliacinę ir aprūpinančiąją sistemas ir yra santykis (indeksas i reiškia, kad parametrai gaunami esant tam pačiam galingumui N):

$$\frac{JT}{RR_i} = \frac{JT_i}{RR_i} \quad (1.23)$$

Ilgalaikių fizinių treniruočių metu sumažėja \check{SSD} bei sutrumpėja JT intervalas, t.y. širdžiai tenka mažesnis krūvis. Taigi santykis JT/RR mažėja. Vyrų širdys apytikriai trečdaliu yra didesnė už moterų, jų šis santykis mažesnis negu moterų. Kokios fiziologinės ypatybės lemia šias savybes, nėra žinoma.

0.2 lentelė

Fiziologinės kardiologinių parametų ribos naudojamos eilučių parametų normavimui

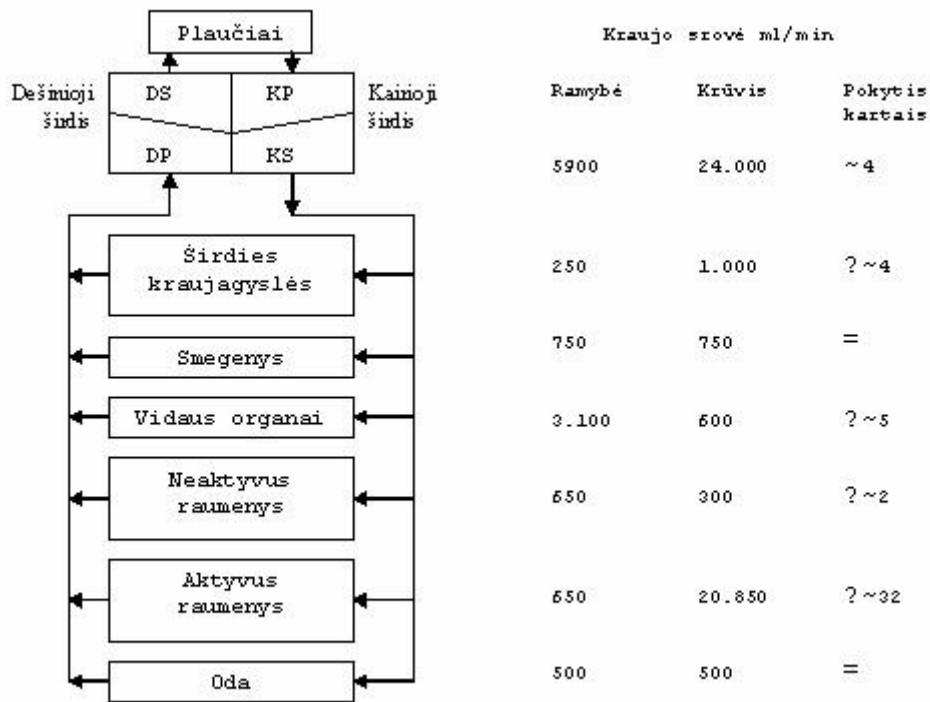
Parametras	Minimali fiziologinė riba	Maksimali fiziologinė riba
RR, ms	140	1500
DJT, ms	50	400

Elektrokardiografinių duomenų analizei fizinio krūvio metu, buvo naudoti EKG duomenys, užregistruoti taikant provokacinio veloergometrino pakopomis didėjančio fizinio krūvio metodiką. (iš pradžių 1 min EKG registruojama ramybėje, vėliau važiuojama specialiu dviračiu, veloergometru, taikant pradinį 50 W krūvį (*N*), jį didinant kas minutę po 25 W moterims ir po 50 W vyrams, tol, kol atsiranda klinikiniai požymiai, ribojantys krūvį), paskui EKG registruojama dar 3 - 5 atsigavimo minutes. Kraujospūdis matuojamas kas minutę. EKG parametrai iš signalo gali būti matuojami dviem protokolais: I – uoju, kai EKG signalo parametrai išskiriami kas minutę, II – uoju, kai parametrai išskiriami kiekvieno kardiociklo metu (žr. prieduose 8.2 skyrelis).

AMŽIAUS ĮTAKA ŠIRDIES IR KRAUJAGYSLIŲ SISTEMOS REGULIAVIMUI

Augant organizmui, pirmuosius 10 - 15 metų, kol dar neišsivystė kraujo srovės organizme perskirstymo mechanizmas, pagrindinė svarba, didinant širdies darbingumą fizinio aktyvumo metu, tenka širdies susitraukimų dažniui. Tuo galime įsitikinti nagrinėdami fiziologinio maksimalaus širdies dažnio priklausomybę nuo amžiaus: kuo žmogus jaunesnis, tuo maksimalus širdies susitraukimų dažnis didesnis. Visiškai susiformavus organizmui, padidėjus širdies susitraukimo jėgai, didėja išstumiamo į aortą kraujo kiekis - sistolinis tūris. Todėl subrendus organizmui, pagrindinį širdies ir kraujagyslių sistemos darbingumo didėjimą apibūdina suminio vertinimo kitimas. Vyresniems žmonėms, atsiradus pokyčiams kraujagyslėse, mažėja diastolinė funkcija (kaip kompensacija didėja arterinio kraujo spaudimas). Jau vien šie pokyčiai lemia, jog fizinis aktyvumas skirtingais amžiaus etapais aktyvina skirtingus fiziologinius adaptacinius mechanizmus, t.y. skirtingus jų parametrus. Taigi amžius, lytis, įvairios patologijos lemia individualaus fizinio aktyvumo būtinumą.

Panagrinėkime bendruosius dėsningumus, subrendusiam organizmui reaguojant į fizinį aktyvumą (1.4 pav).



1. 3 pav. Kraujo tėkmės (ml/min) persiskirstymas įvairiuose organuose ramybės ir fizinio krūvio metu

Pastebime, kad krūvio metu vyksta pokyčiai visame organizme, ne tik aktyviuose raumenyse. Visas organizmas prisitaiko prie kintančio krūvio. Skirtingose organų sistemose šie pokyčiai priešingų kryptių. labiausia stengiamasi padidinti kraujotaką per aktyvią raumenų grupę bei tą darbą lemiančių organų sistemą (širdį, plaučius). Per organus, kurie labai svarbūs skatinant atlikti krūvį, stengiamasi išlaikyti stabilią kraujotaką (pvz., smegenis, kurios lemia visos sistemos reguliaciją, ir per odą, kuri reguliuoja šilumos apykaitą tarp organizmo ir aplinkos). Kitose organų sistemose tekančio kraujo kiekis labai sumažėja. Visi šie pokyčiai ir sudaro hemodinamikos perskirstymo organizmą, kurio pagrindinė paskirtis atlikti organizmo adaptaciją. Kuo didesnius pokyčius (aišku, neperžengdami tam tikrų fiziologinių pokyčių ribų) mes galime sukelti organizme, tuo prie didesnio krūvio organizmas gali prisitaikyti, t.y. organizmas bus veiklesnis, pajėgesnis.

ŠIRDIES IR KRAUJAGYSLIŲ SISTEMOS FUNKCINIŲ RODMENŲ KAITA, ATLIEKANT FIZINIO KRŪVIO MĖGINIUS

Organizmo adaptavimasis ilgalaikėms fizinėms treniruotėms bei adaptavimosi proceso kontrolė yra aktualus fiziologijos ir sporto medicinos uždavinys. Šio uždavinio sprendimas sietinas su organizmo funkcinės būklės įvertinimu.

Organizmo širdies ir kraujagyslių sistemos funkcinės būklės įvertinimui atliekami fizinio krūvio mėginiai. Fiziologijos, klinikinės medicinos bei sporto mokslo žurnaluose galima rasti daug dabar taikomų žmogaus organizmo funkcinio pajėgumo bei adaptavimosi pokyčių vertinimo protokolų.

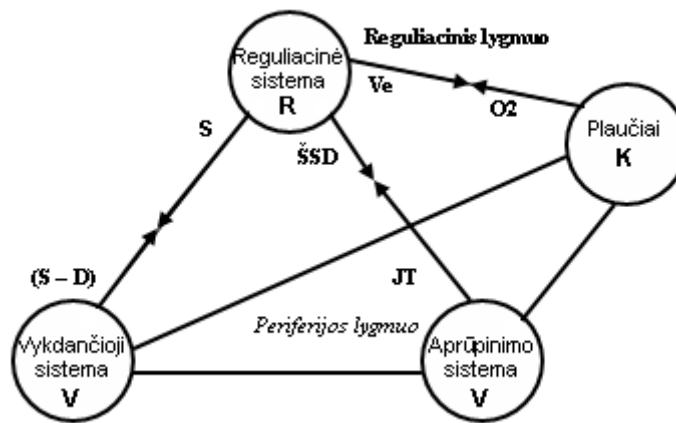
Iš gausybės vertinimo kriterijų galima išskirti tris grupes rodiklių, nusakančių organizmo darbingumą ar kitų parametru pokyčius bei nurodančių teigiamą ilgalaikių fizinių treniruočių poveikį sportuojančio žmogaus organizmui. Pirmajai - maksimalių galimybių rodiklių grupei priskiriami fizinio darbingumo ir funkcinių organizmo sistemų maksimalaus aktyvumo laipsnio įvertinimo parametrai (maksimalus išvystytas galingumas, maksimali fizinio krūvio trukmė, maksimalus organizmo funkcijų aktyvumo laipsnis). Kokybiniais priskiriami rodikliai, nurodantys organizmo funkcijų aktyvumo laipsnio pokyčius, atliekant tam tikro intensyvumo fizinius krūvius; kiekybiniais rodikliams – santykiniai rodikliai, parodantys biologinę atliekamo fizinio krūvio vertę. Dabar klinikinėje praktikoje naudojami organizmo funkcinę būklę įvertinantys parametrai ir indeksai ne visai įvertina sisteminių atsaką į fizinį krūvį. Vieni jų atspindi bei rodo kvėpavimo sistemos funkciją, kiti - širdies ir kraujagyslių sistemos atsaką, tretieji apibendrina organizmo pajėgumą bei reakciją į fizinį krūvį, neįvertindami atskirų sistemų veiklos.

Atliekant pratimą ilgiau nei keletą sekundžių, labai svarbu, kad į skeleto raumenis būtų tiekama adekvatus krūviui deguonies ir energetinių medžiagų kiekis. Judėjimo ir atramos sistema fizinio krūvio metu adekvačiai aprūpinama tada, kai suaktyvėja aprūpinimo sistemų veikla, o tai apsprendžia reguliuojančios sistemos. Atliekant šį tyrimą organizmo reakcijai į fizinį krūvį įvertinti išskirtos anksčiau minėtos sistemos: vykdančioji sistema (dirbantis raumenynas), reguliacinė, apimanti centrinę nervų sistemą, autonominių bei humoralinių valdymą ir aprūpinančioji – širdies ir kraujagyslių sistema. Lietuvoje sukurtas integralios organizmo reakcijos į fizinį krūvį modelis teikia naujas analizės galimybes. Naudojantis šiuo modeliu galima įvertinti minėtų sistemų veiklą tiriamajam atliekant fizinio krūvio mėginį. Fizinio krūvio metu vyksta kraujotakos perskirstymas, kas leidžia organizmui adekvačiau atlikti fizinį krūvį, t.y. padėti pagrindinėms organizmo sistemoms adaptuotis fiziniam krūviui. Juo didesni pokyčiai, neviršijant tam tikrų fiziologinių ribų, atsiranda organizme minėtų trijų sistemų atžvilgiu, tuo didesni fizinį krūvį organizmas gali atlikti. Dėl to organizmo pagrindinės sistemos geriau adaptuojasi bei išvysto didesnę pajėgumą.

Organizmui išvysčius tam tikrą pajėgumą, jame vyksta kompleksinė reakcija į fizinį krūvį, kurią apsprendžia minėti elementai. Remiantis žmogaus fiziologija galima išskirti šiuos pagrindinius elementus, apsprendžiančius pagrindinius pokyčius organizme krūvio metu:

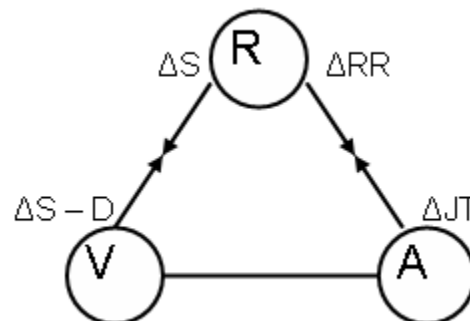
- reguliacinė sistema (R) - smegenys visuose reguliaciniuose lygmenyse;
- (V) vykdančioji sistema - fizinio krūvio metu dirbantis raumenys;
- širdies ir kraujagyslių - aprūpinančioji sistema (A), užtikrinanti adekvačią kraujotaką;
- kvėpavimo sistema (K), kuri, priklausomai nuo pratekančio per plaučius kraujo kiekio, absorbuoja deguonį bei išskiria anglies dvideginį.

Šias, keturias sistemas apibendrinantis modelis schematiškai pavaizduotas (1.5 pav.).



1. 4 pav. Žmogaus organizmo funkcinės būklės įvertinimo modelis krūvio metu

Santykius tarp šių elementų ir jų funkcijų, galima nusakyti daugeliu parametų. Modelyje nurodyti ir nežalojančiu būdu nustatomi, elementų funkciją atspindintys dydžiai. Šį modelį iš dalies galima supaprastinti jungiant kvėpavimo ir širdies bei kraujagyslių sistemas į vieną aprūpinančią (A) sistemą. Per dešiniąją širdį, plaučius ir kairiąją širdį kraujas teka nuosekliai, nėra nei nuotėkio (praradimo), nei pritekėjimo (papildymo), todėl hemodinamiką šioje organizmo dalyje lemia širdies funkcija ir kvėpavimo funkcija (t. y. absorbuoto deguonies kiekis) nuo jos visiškai priklauso. Taigi galima modelį supaprastinti (1.6 pav.).



1. 5 pav. Integralinio vertinimo modelis ir organizmo sistemų tarpusavio ryšius rodantys žymenys (A. Vainoras, 1996), V - vykdančioji sistema, R - reguliacinė sistema, A – aprūpinančioji sistema

Išskiriami trys funkciniai modelio elementai: 1) vykdančioji sistema (V), fizinio aktyvumo metu veikianti raumenų grupė; 2) reguliacinė sistema (R), apimanti CNS, autonominio bei humoralinio valdymo elementus; 3) širdies ir kraujagyslių - aprūpinančioji sistema (A), atsakinga už centrinę hemodinamiką. Santykį tarp šių elementų galima nusakyti daugeliu parametų, tačiau šiam tyrimui buvo pasirinkti paprasčiausi ir lengviausiai nustatomi. Veikiančią raumenų sistemą vertiname pasiektu galingumu N, reguliacinę sistemą - intervalu RR (laikas tarp dviejų širdies susitraukimų), o širdies veiklos suintensyvėjimą - JT intervalu (intervalas elektrokardiogramoje nuo jungties J iki T bangos

pabaigos). Organizmo adaptacija fiziniam krūviui vertinta remiantis pasirinktų parametų pokyčiais. Normalizuotų pokyčių suma vertinta kaip nuotolis Euklido erdvėje ir sudaro suminį vertinimą (Sv). Jo dydis susijęs su tiriamojo funkcinė būkle: sportininkams jis sudaro 70–100 proc., sveikiems tiriamiesiems 60–70 proc, o turintiems patologiją, šis dydis mažėja. Atliekant šį vertinimą, reikšmingas ne tik pats dydis, bet ir jo kitimas bei jį sudarančių parametų tarpusavio santykis. Jo pokyčiai rodo organizmo reguliacinių procesų pokyčius, kurie dar nesukelia pataloginių reiškinių, bet rodo, jog yra jų tikimybė. Šis modelis pagal parametų pokyčius bei jų santykį padeda įvertinti organizmo atsaką į įvairius poveikius [14, 15].

IŠEMINĖ ŠIRDIES LIGA

Išeminė širdies liga – tai miokardo disfunkcija dėl reliatyvaus ar visiško arterinio kraujo pritekėjimo į miokardą sutrikimo, pasireiškianti ūminiais ir lėtiniais išeminiais sindromais.

Širdies ir kraujagyslių ligos yra viena svarbiausių ne tik medicinos, bet ir socialinių problemų. Širdies ir kraujagyslių ligos sudaro apie pusę visų mirčių Lietuvoje, 1/3 invalidumo bei nulemia 15–20 proc. apsilankymų sveikatos priežiūros įstaigose. 1999 metais Lietuvoje nuo širdies ir kraujagyslių ligų mirė 21 903 žmonės, arba 592,0/100 000 gyventojų.

Išeminė širdies liga yra labiausiai paplitusi širdies ir kraujagyslių sistemos liga. Ji yra viena dažniausia civilizuotų šalių gyventojų mirties priežastis. Užkirsti kelią išeminei širdies ligai sunku, nes jos pradžia beveik nepastebima. Pirmieji ligos požymiai pasireiškia gana vėlai, kada liga yra įsigalėjusi, širdis išsekusi. Todėl gydytojui dažnokai tenka pradėti gydyti toli užleistą ligą. Pastarųjų dešimtmečių mokslininkų duomenimis laiku pradėjus gydyti sergantįjį, išeminę širdies ligą, galima sustabdyti. Labai svarbu, kad ligonis žinotų, kaip dirba jo širdis, kas trukdo jai plakti, kaip galima apsaugoti, kaip reikia gyventi, kad ilgai galėtų būti aktyvus ir darbingas.

Širdis yra žmogaus kumščio dydžio ir sveria apie 300 - 400 gramų. Sergančiojo žmogaus širdis gali būti ir didesnė. Širdis - tai raumenims siurblys. Vieną kartą susitraukdama širdis perpumpuoja apie 70 -80 mililitrų kraujo, o per parą, susitraukdama iki 100 000 kartų širdis perpumpuoja apie 10 tonų kraujo.

Išemija - tai dėl kokių nors priežasčių pablogėjusi vainikinių širdies kraujagyslių kraujotaka ir dėl to sutrikęs širdies raumens aprūpinimas krauju ir maisto medžiagomis. Dažniausia jos priežastis – širdies vainikinių kraujagyslių aterosklerozė, kai susidariusios plokštelės užkemša kraujagyslių spindį. Tačiau širdies raumens išemiją gali sukelti ir širdies vainikinių kraujagyslių trombai, spazmai, plaučių, kraujo ir kitos ligos. Skiriamos trys pagrindinės išeminės širdies ligos formos:

- krūtinės angina,
- širdies infarktas,
- staigi mirtis.

Nepakankamai krauju aprūpinimas širdies raumuo pakinta, todėl dažnai pradeda skaudėti širdies plote. Tai vadinama krūtinės angina. Kartais sutrinka širdies ritmas, pasireiškia širdies nepakankamumo požymiai: trūksta oro, dūstama, tinsta kojos. Iš pradžių šie reiškiniai pasireiškia fizinio krūvio metu, kai, širdžiai pasidaro sunkiau dirbti, o vėliau, kai dar labiau susiaurėja kraujagyslės, ir ramybės būsenoje. Negaudamos kraujo, širdies raumens ląstelės žūva, prasideda infarktas. Sveikstant, per du mėnesius, infarkto vietoje susidaro randas iš jungiamojo audinio. Dėl rando širdies raumenyje gali susilpnėti širdies susitraukimai, sutrikti jos ritmas.

Staigi mirtis ištinka tada, kai širdies išemija prasideda ūmiai, ir širdis nespėja prisitaikyti prie pakitusių sąlygų, tada pradeda virpėti širdies skilveliai arba širdis sustoja.

Nustatyta, kad išeminę širdies ligą sukelia ne vienas, o daugelis rizikos veiksnių. Svarbiausi išeminės širdies ligos rizikos faktoriai yra:

- netinkama mityba,
- rūkymas,
- nepakankamas fizinis aktyvumas,
- per didelė kūno masė,
- padidėjęs arterinis kraujospūdis,
- žmogaus psichinė būseną ir patiriami stresai,
- žalingi aplinkos faktoriai, tokie kaip oro užterštumas, įvairūs chemikalai ir t.t.,
- genetinis faktorius.

Ypač nepalankiai veikia kelių rizikos faktorių derinys. Jei vienu metu veikia 2 ar 3 rizikos faktoriai, tikimybė susirgti išemine širdies liga yra dešimtis kartų didesnė. Nepašalinus rizikos faktorių ir gydant tik vaistais, neužkirsime kelio išeminei širdies ligai.

Nustatyta, kad kelių rizikos veiksnių derinys didina išeminės širdies ligos riziką, o susirgus – mirštamumą nuo jos.

Būtent remiantis krūvio mėginio rezultatais sprendžiama dėl diagnozės ir kaip toliau tyrinėti. Jei pacientas pakelia didelį krūvį, neatsiranda EKG išeminių pakitimų, normaliai kyla kraujospūdis ir pulsas, tada dažniausiai siūloma gydytis vaistais (silpnais) ar ieškoti kitos skausmų krūtinėje priežasties ir, gal būt, po kiek laiko krūvio mėginį reiktų pakartoti. Jei pacientas pakelia tik vidutinį ar mažą fizinį krūvį, bet EKG neatsiranda išemijos požymių, skausmo krūtinėje krūvio metu, o būna tik silpnumas, kojų nuovargis, oro stoka, galvos svaigimas, per greitai pakyla kraujospūdis ar pulsas – rekomenduojama gydytis vaistais ir po kiek laiko (savaičių ar mėnesių) tyrimą pakartoti. Jei pacientas pakelia tik mažą fizinį krūvį ar atsiranda skausmas krūtinėje ir/ar elektrokardiogramos pakitimai, tada svarstoma apie širdies zondavimo tikslingumą ir dažniausiai pacientui siūloma šį labai svarbų tyrimą atlikti [13, 14, 15, 17].

NAUDOTA METODIKA

Medikai jau nuo seno žino, kad žmogaus organizmas patirdamas krūvį ir atsipalaiduodamas veikia skirtingai. Yra žinoma, kad patiriant krūvį ir jį palaipsniui didinant veikia vieni organizmo atraktoriai, o atsipalaiduojant organizmą ima veikti kiti. Mokslininkai pritaiko tam tikrus algoritmus ir iš gautų rezultatų mato, kad tikrai organizme vykstantys procesai šiais dviem momentais yra skirtingi.

Iki šiol jie neturėjo galimybės įsitikinti tuo vizualiai. Todėl šiame darbe mes nagrinėjome elektrokardiogramas, kurios parodo ne tik širdies darbą, bet ir atspindi viso kūno būseną. Nagrinėjome EKG parametrus, jiems pritaikėme matematinius metodus ir atvaizdavome grafiškai. Programą ir vartotojo sąsają sukūrėme „Matlab“ aplinkoje. sukurtais metodais gautus rezultatus patikrinome neparimetrinių Manio-Vtnio-Vilkoksono testu, „SPSS“ (statistinės analizės ir duomenų apdorojimo programine įranga) paketu.

TIRIAMOJI DALIS

DUOMENYS

Nagrinėjome 20 diagnostinių tyrimų, gautų naudojant veloergometrinio pakopomis didėjančio provokacinio fizinio krūvio metodiką [18]. Pagal tiriamojo amžių, gyvenimo būdą bei sveikatos būklę jie buvo suskirstyti į dvi grupes – profesionalių sportininkų (sp) (7 asmenys) ir įvairias širdies ir kraujagyslių sistemos patologijas turinčių asmenų grupę – ligonių (is) (13 asmenų).

Veloergometriniui mėginiui atlikti naudota Kauno medicinos universitete Kardiologijos institute sukurta elektrokardiogramos (EKG) registracijos ir analizės sistema „Kaunas – Krūvis W01“. Sinchroniškai registruota 12-kos standartinių derivacijų EKG. Taikyta veloergometrinio pakopomis didėjančio provokacinio fizinio krūvio metodika. Mano tyrimui atlikti, buvo atrinkti reikalingi parametrai *RR* ir *JT*, kurie buvo fiksuojami kiekvieno kardiociklo metu. Atrinkti duomenys saugomi byloje „Duomenys“. Vienam tyrimui skirtas vienas tekstinis dokumentas.

Veloergometrijos tyrimai atlikti, taikant pradinį 50 W krūvį (N) ir kas minutę jį didinant 25 W. Maksimalus, praktiškai pasiekiamas galingumas vyrams buvo 300 W, o moterims 175 W. Krūvis buvo didinamas kas minutę ir tęsiamas iki submaksimalaus širdies susitraukimo dažnio arba kol atsiranda klinikiniai požymiai, ribojantys krūvį. Taip pat nagrinėti parametru kitimai per 5 atsigavimo minutes, priklausomai nuo to, kaip greit jie atsistatydavo.

Darbe nagrinėjome parametru kitimą laike. Nepriklausomai nuo to kiek laiko truko tyrimas ir kokį maksimalų galingumą tiriamasis išvystė. Duomenų atskaitų skaičius svyruoja nuo 1500 iki 3500 stebėjimų kiekvienam tiriamajam.

Pradinių duomenų pavyzdys yra 2.1 lentelėje:

0.1 lentelė

Pradinių duomenų pavyzdys

Laikas, ms	RR, ms	JT, ms
1260	1040	322
2268	1012	322
3276	1004	322
4280	1002	318
5248	974	322
...

TYRIMAS

Kiekvieno tyrimo metu gauti duomenys – parametru *RR* ir *JT* laikinės sekos: $RR(t), JT(t)$.

Kadangi medicinos literatūroje nurodoma, jog parametro *JT* kitimas fizinio krūvio metu stipriai skiriasi sveikiems ir išemine liga sergantiems asmenims, tai savo darbe toliau daugiausiai dėmesio ir skirsiu šio parametro analizei. Pagal išeminės ligos poveikį širdies veiklai galima tikėtis, kad sveikų

tiriamųjų JT parametro reikšmė didinant krūvį mažės greičiau, o išvestinė įgis moduliu didesnes reikšmes.

Nagrinėtų EKG signalų parametų reikšmių aibės yra labai skirtingos, todėl tam, kad šie skirtumai neįtakotų gaunamų rezultatų ir juos būtų galima palyginti tarpusavyje, pradiniai duomenys buvo normuojamai į intervalą $[0; 1]$:

$$x_i' = \frac{x_i - x_{\min}}{x_{\max} - x_{\min}} ;$$

čia i – kardiociklo numeris; x_i' – sunormuota parametro reikšmė taške i ; x_i - pradinė parametro reikšmė taške i ; x_{\min} , x_{\max} - fiziologinės parametų minimali ir maksimali reikšmės (pateiktos 02 lentelėje).

0.2 lentelė

**Fiziologinės kardiologinių parametų ribos naudojamos
laiko eilučių parametų normavimui**

Parametras	Minimali fiziologinė riba	Maksimali fiziologinė riba
RR, ms	140	1500
DJT, ms	50	400

Tam, kad nustatytume parametų reikšmių priklausomybę nuo gaunamo krūvio ar atsigavimo laiko, gautas pradines duomenų reikšmes interpoliuojame kubiniu interpoliaciniu splineu skaičiuojame tarpines nustatytos priklausomybės reikšmes. Interpoliavimo kubiniu splineu pavyzdys 2.1 pav. viršutinis grafikas (mėlynas).

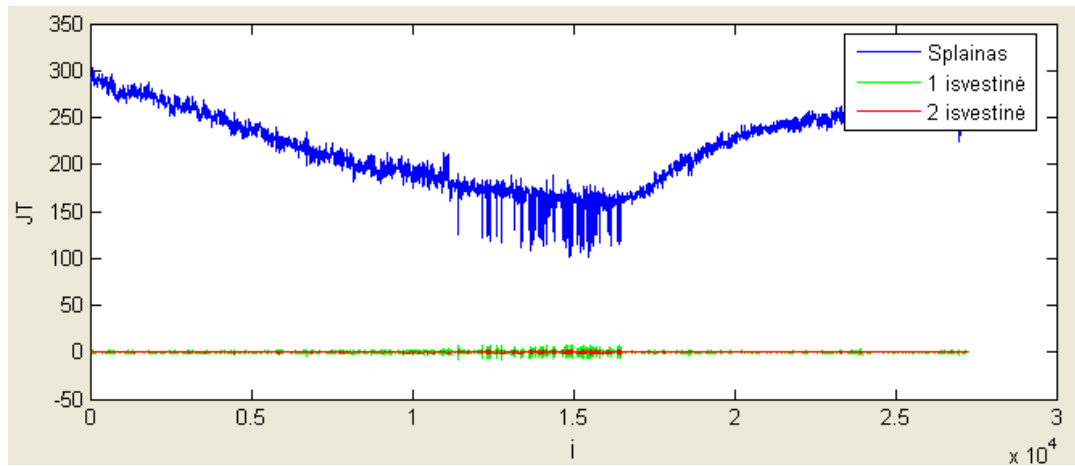
Turėdami parametro kitimo laike funkcinę išraišką, ieškome šios funkcijos išvestinės. Kadangi atstumai tarp interpoliavimo taškų vienodi, naudojame baigtinius skirtumus.

$$JT'(t) = \frac{JT(t + \Delta t) - JT(t)}{\Delta t} = \frac{dJT}{dt} .$$

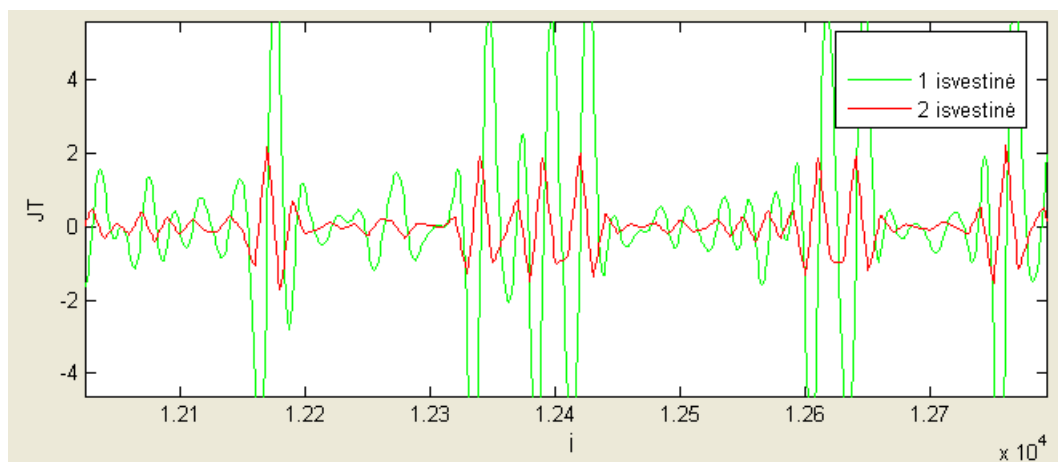
Taigi gauname funkcijos kitimo greitį. Analogiškai randame ir funkcijos antrąją išvestinę – funkcijos kitimo pagreitį.

$$JT''(t) = \frac{d^2 JT}{dt^2} .$$

Grafiko pavyzdys, kai stebime JT parametro dinamiką laike, pateiktas 2.1 pav. (2.2 pav. – padidintas grafiko vaizdas), žalia linija - gauta pirmoji išvestinė, o raudona – antroji.



2. 1 pav. Turimų reikšmių interpoliavimas kubiniu splainu



2. 2 pav. Padidintas išvestinių grafiko vaizdas

Toliau lyginsime abiejų grupių duomenis.

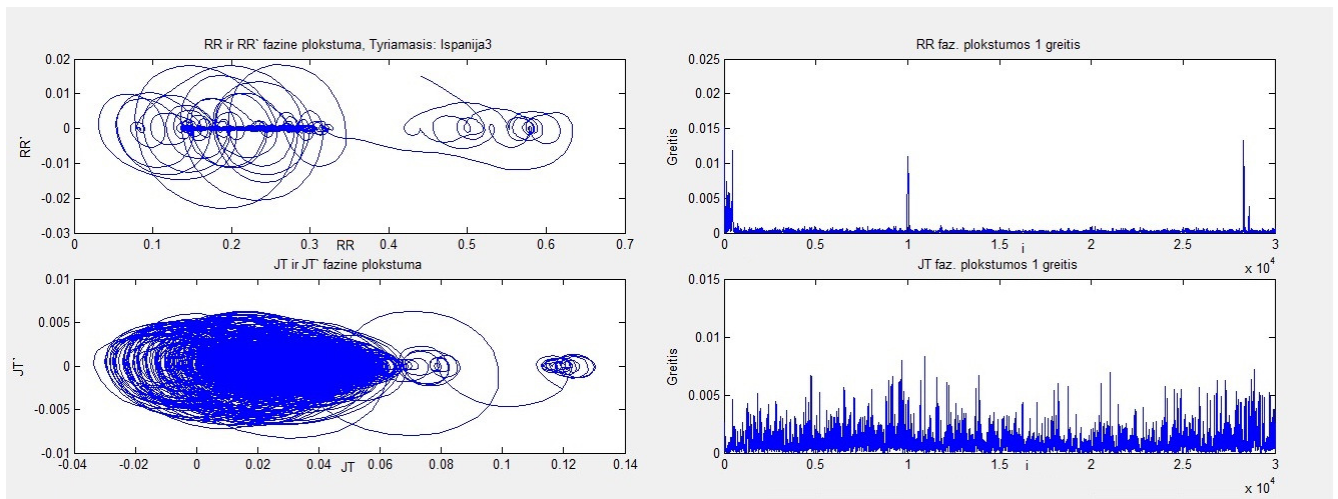
EKG PARAMETRŲ KITIMO GREIČIO SKAIČIAVIMAS IR VAIZDAVIMAS FAZINĖJE PLOKŠTUMOJE

Turimoms kiekvieno asmens EKG parametro JT fazinėms plokštumoms apskaičiavome fazinius greičius pagal šias formules:

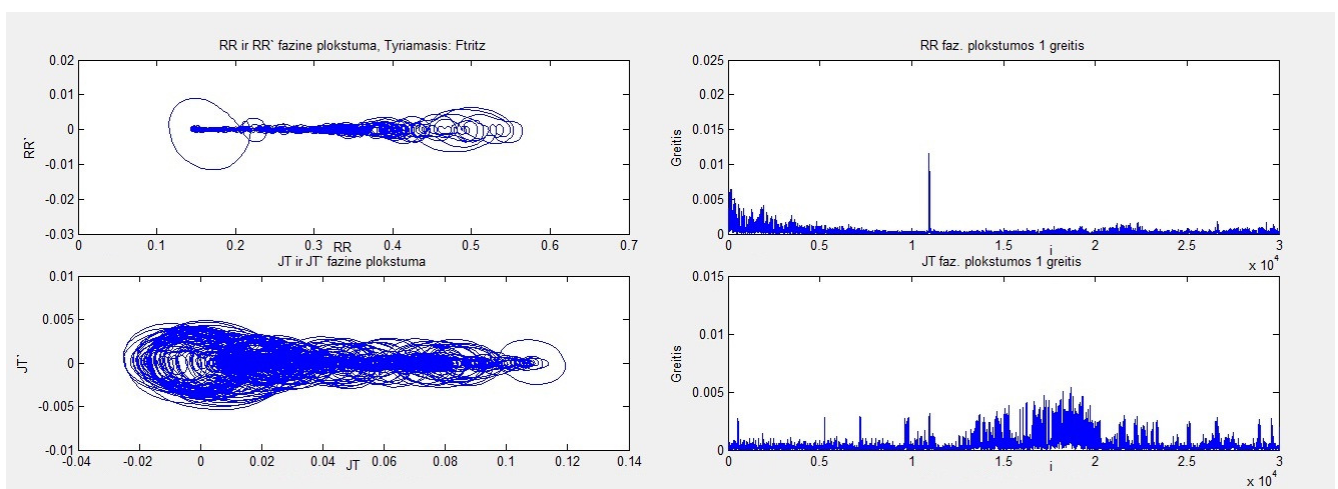
$$\sigma_1 = \sqrt{(f'(x))^2 + (f''(x))^2},$$

$$\sigma_2 = \sqrt{(f''(x))^2 + (f'''(x))^2}.$$

Dviejų asmenų fazinių plokštumų RR su RR' , JT su JT' paveikslai ir jų kitimo greičiai pateikti 2.3. pav., 2.4. pav.



2.3. pav. Sportininko fazinių plokštumų ir jų kitimo greičių paveikslai



2.3. pav. Ligonio fazinių plokštumų ir jų kitimo greičių paveikslai

Matome, kad JT parametro fazinės plokštumos JT su JT' kitimo greičiai skiriasi tarp sportininkų ir asmenų su tam tikromis patologijomis grupių. Tačiau tarp skirtingų grupių RR parametro nepastebėjome. Šiai hipotezei pagrįsti pritaikėme statistinius metodus.

Nepriklausomų imčių požymiams (RR ir JT parametro fazinės plokštumos kitimo greičiams) palyginti buvo taikytas neparametrinis Manio-Vitnio-Vilkoksono testas (Z_M), reikšmingumo lygmuo ($p = 0,05$). JT parametro fazinės plokštumos kitimo greičiais sportininkams ir asmenims su klinikiniais nusiskundimais patikimai išsiskyrė, nes $Z_M = -1,983$, $p = 0,047 < 0,05$. Tuo tarpu RR parametro fazinės plokštumos kitimo greičiais šioms dviem grupėms neišsiskyrė: $Z_M = -1,228$, $p = 0,219 > 0,05$ (žr. 2 Priedas).

EKG PARAMETRO GLODUMO KOEFICIENTO SKAIČIAVIMAS IR VAIZDAVIMAS

Šio metodo esmė apskaičiuoti glodumo koeficientą.

Pirmiausia turimus JT parametro duomenis $X = \begin{bmatrix} x_1 \\ \dots \\ x_m \end{bmatrix}$, $m = 1, 2, \dots, N$ transformavome

naudodamiesi Furjė transformaciją:

$$y_k = c_x(k) = \frac{1}{N} \sum_{m=0}^{N-1} x_m e^{-i \frac{2\pi}{N} km},$$

$$Y = \Phi \cdot X = \begin{bmatrix} c_x(1) \\ \dots \\ c_x(N) \end{bmatrix}.$$

Gavome $Y = \begin{bmatrix} y_1 \\ \dots \\ y_m \end{bmatrix}$, $m = 1, 2, \dots, N$. Toliau skaičiavome dydžių absoliutines reikšmes $|y_k|$ ir jas

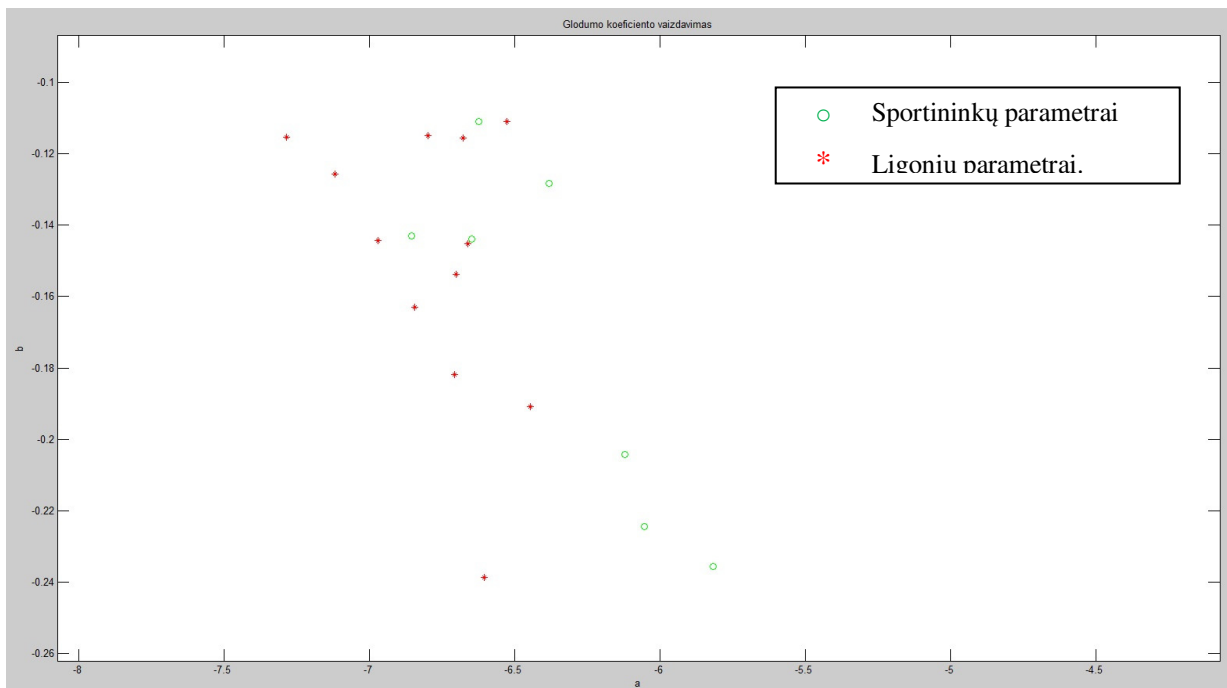
logaritmavome $\ln|y_k|$. Atlikome suglodinimą turimiems duomenims pasinaudodami mažiausių kvadratų metodu (MKM). Skaičiavome suglodintos kreivės $y_k = a \ln k + b$, $k \in N$ a ir b parametrus.

MKM pirmosios eilės polinomui lygčių sistema atrodo taip:

$$\begin{cases} na + b \sum_{i=1}^n z_i = \sum_{i=1}^n y_i \\ a \sum_{i=1}^n z_i + b \sum_{i=1}^n z_i^2 = \sum_{i=1}^n z_i y_i \end{cases} \Rightarrow \begin{cases} b = \frac{n \sum_{i=1}^n z_i y_i - \sum_{i=1}^n z_i \sum_{i=1}^n y_i}{n \sum_{i=1}^n z_i^2 - \left(\sum_{i=1}^n z_i \right)^2} \\ a = \frac{1}{n} \left(\sum_{i=1}^n y_i - b \sum_{i=1}^n z_i \right) \end{cases} \quad (2.1)$$

čia $z_k = \ln k$, $k \in N$.

Iš (2.1) sistemos apskaičiavome nežinomuosius parametrus a ir b . Juos atvaizdavome grafike:



2.4. pav. Glodumo koeficientų paveikslas

Kaip matome, asmenų su klinikiniais nusiskundimais parametrai grupuojasi grafiko kairėje, sportininkų didesnė dešinėje.

Ir šio metodo rezultatams tarp skirtingų grupių asmenų buvo taikytas neparametrinis Manio-Vitnio-Vilkoksono testas (Z_M) su reikšmingumo lygmeniu $p = 0,05$ nepriklausomų imčių požymiams (koeficientams a ir b) palyginti. koeficientas a sportininkams ir asmenims su klinikiniais nusiskundimais patikimai išsiskyrė, nes $Z_M = -2,100$, $p = 0,036 < 0,05$. Tuo tarpu koeficientas b šioms dviem grupėms neišsiskyrė: $Z_M = -0,277$, $p = 0,782 > 0,05$ (žr. 2 Priedas).

EKG PARAMETRO DUOMENŲ GRUPAVIMAS

Pirmiausiai atlikome duomenų grupavimą. Tegul, duota baigtinė laiko eilutė x_1, x_2, \dots, x_n . Po to tą eilutę sugretinome pačią su savimi.

$$\begin{array}{cccccccc}
 & x_1 & x_2 & \dots & x_{n-k-1} & x_{n-k} & \dots & x_{n-1} & x_n \\
 x_1 & \dots & x_{k-1} & x_k & x_{k+1} & \dots & x_{n-1} & x_n &
 \end{array}$$

Sudarome poras:

$$\begin{pmatrix} x_1 & x_2 & x_3 & \dots & x_{n-k-1} & x_{n-k} \\ x_k & x_{k+1} & x_{k+2} & \dots & x_{n-1} & x_n \end{pmatrix} = \begin{pmatrix} x_1 & x_2 & x_3 & \dots & x_{n-1} & x_n \\ y_1 & y_2 & y_3 & \dots & y_{n-1} & y_n \end{pmatrix},$$

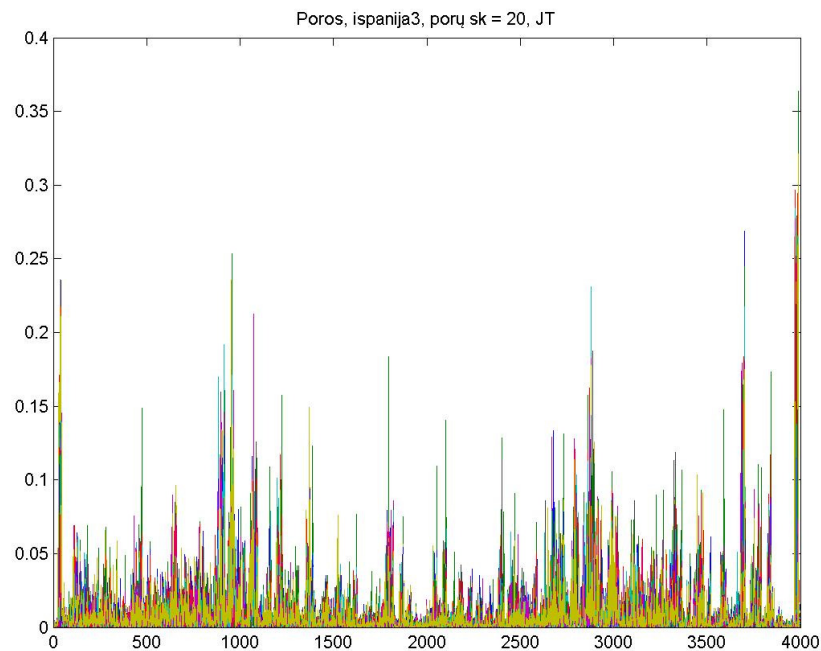
kai k yra fiksuotas. Iš šių porų sudarome tokias matricas:

$$\begin{bmatrix} x_{k-1} & x_k - y_k \\ x_{k-2} - y_{k-2} & y_{k-1} \end{bmatrix}, k = 3, 4, \dots, n.$$

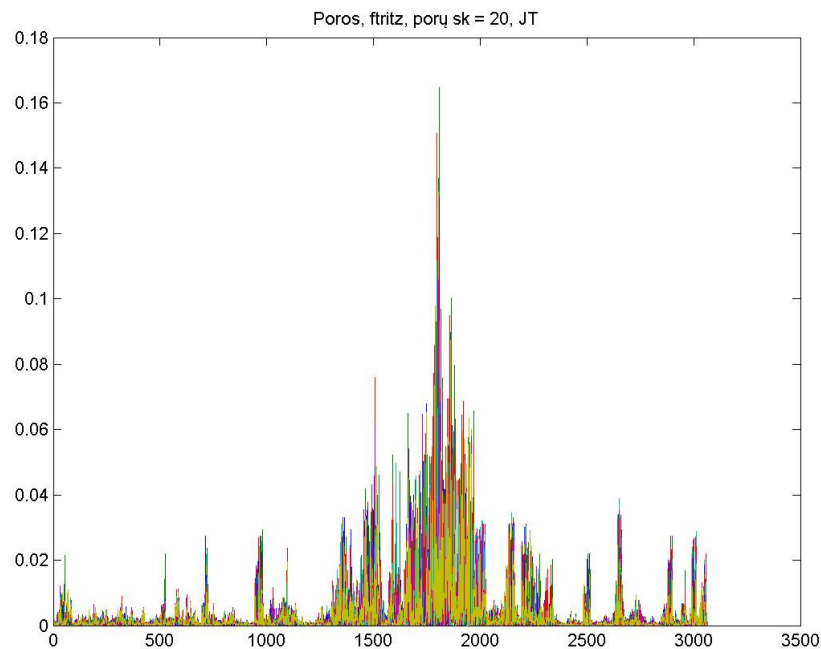
Po to apskaičiuojame šių matricų diskriminantus pagal formulę

$$dsk \begin{bmatrix} x_i & x_{i+1} - y_{i+1} \\ x_{i-1} - y_{i-1} & y_i \end{bmatrix} = (x_i - y_i)^2 + 4(x_{i+1} - y_{i+1})(x_{i-1} - y_{i-1}).$$

Galiausiai, gautus matricių diskriminantus atvaizdavome grafikuose.



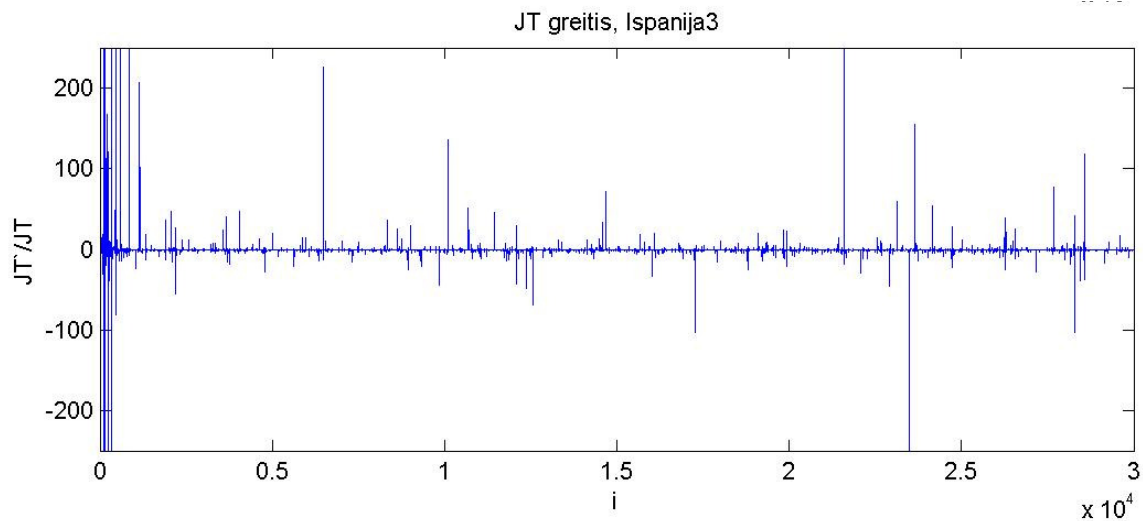
**2.5. pav. Sportininko 20-ies porų diskriminantų paveikslas;
ant Ox ašies i , ant Oy diskriminanto reikšmė**



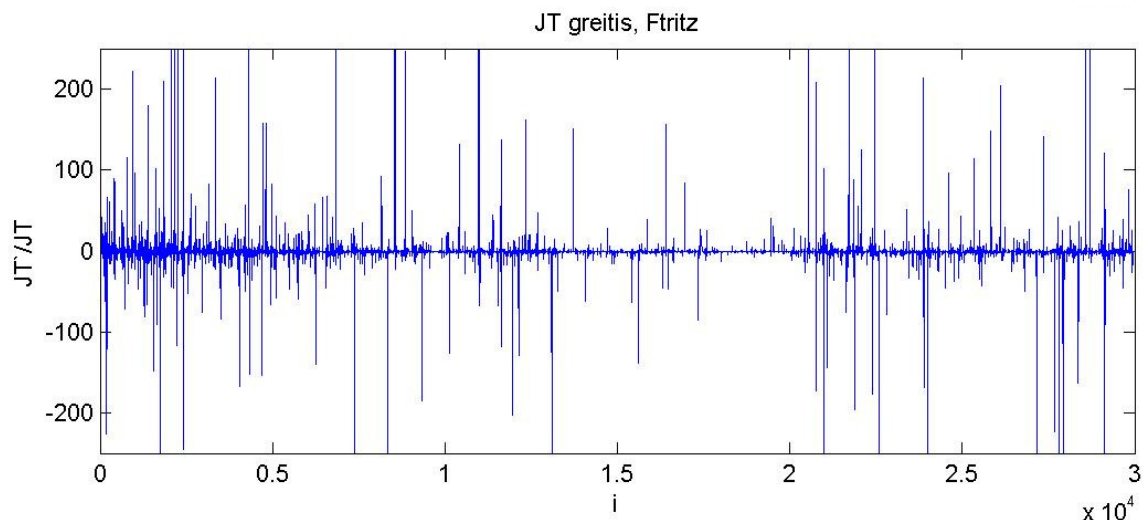
**2.6. pav. Ligonio 20-ies porų diskriminantų paveikslas;
ant Ox ašies i , ant Oy diskriminanto reikšmė**

(2.5) ir (2.6) paveiksluose pavaizduoti dvidešimties duomenų rinkinių (x_i, x_{i+1}) , (x_i, x_{i+2}) , ..., (x_i, x_{i+k}) diskriminantai, sudaryti iš vieno asmens EKG JT parametro duomenų stulpelio.

Pastebėjome, kad (2.5) ir (2.6) paveikslai panašūs į JT parametro logaritminės išvestinės grafikus, tačiau amplitudės kitimas yra priešingas (žr. 2.7. pav., 2.8. pav.).



2.7. pav. Sportininko logaritminė išvestinės grafikas



2.8. pav. Ligonio logaritminė išvestinės grafikas

Pastebėjome, kad diskriminantų paveiksle (2.5 pav., 2.6. pav.) sportininkų diskriminantų amplitudės yra didesnės, nei asmenų su klinikiniais nusiskundimais, o logaritminių išvestinių grafikuose (2.7 pav., 2.8. pav.) – atvirkščiai. Pritaikėme Manio-Vitnio-Vilkoksono testą (Z_M) su reikšmingumo lygmeniu $p = 0,05$ nepriklausomų imčių požymių (porų diskriminantų vidurkio ir logaritminės išvestinės modulio vidurkio) palyginimui. Gavome tokius rezultatus:

- Porų diskriminantų metodo: $Z_M = -2,220$, $p = 0,026 < 0,05$;
- Logaritminės išvestinės metodo: $Z_M = -3,130$, $p = 0,002 < 0,05$.

Manio-Vitnio-Vilkoksono testo rezultatai pateikti 2 priede.

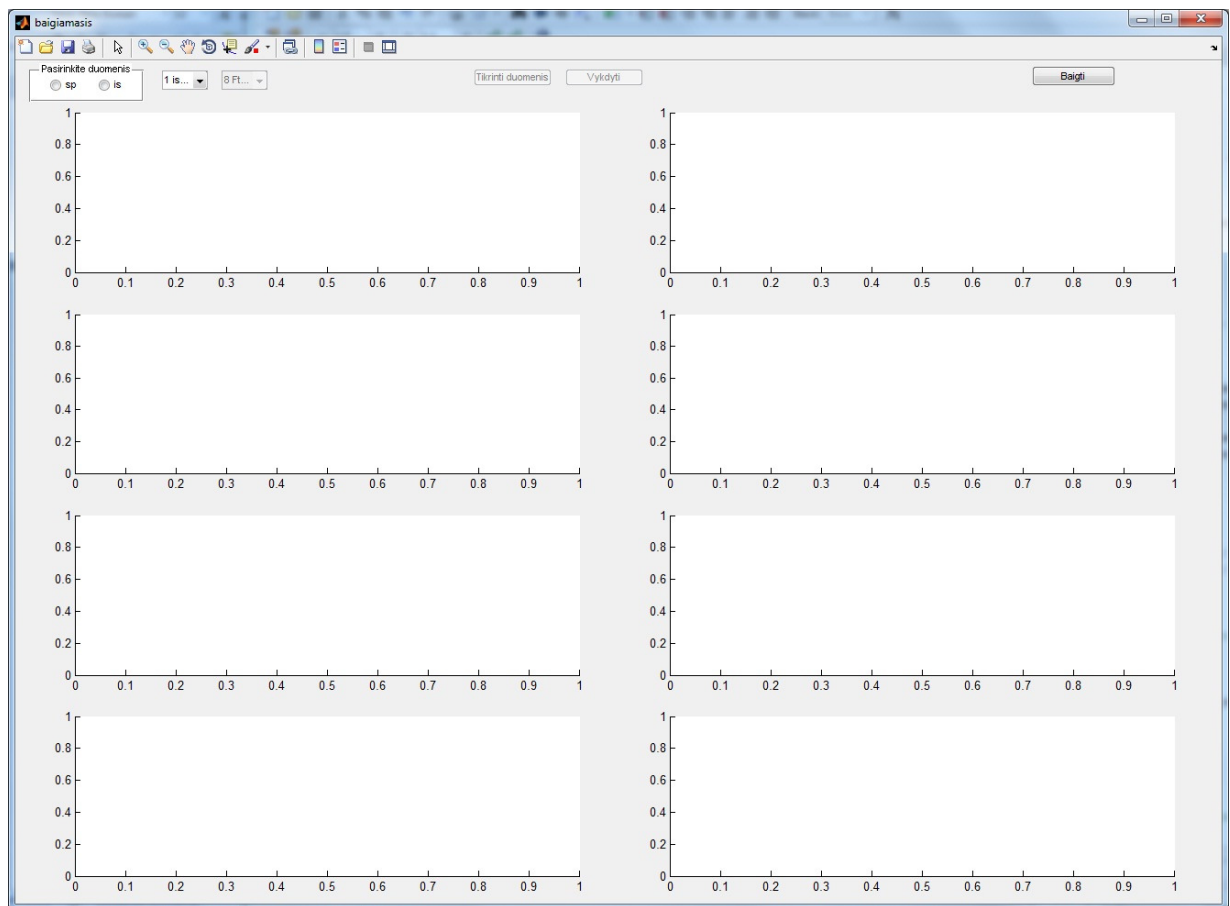
Abiejų metodų rezultatai patikimai skiriasi. Tai reiškia sportininkų EKG parametrų RR ir JT logaritminių išvestinių reikšmės reikšmingai mažesnės, nei sergančių asmenų, tuo tarpu porų

diskriminantų vidurkio reikšmės reikšmingai didesnės, nei sergančiųjų. Tai galėtų būti vienas kriterijų, kaip atskirti sveikus ir sergančius asmenis, kai pradiniai EKG duomenys jokių skirtumų neparodo.

PROGRAMINĖS ĮRANGOS PASIRINKIMO PAGRINDAS IR INSTRUKCIJA VARTOTOJUI

Šio darbo viena iš užduočių buvo sukurti programinę įrangą, kurios tikslas tirti elektrokardiogramų parametrus, panaudojant matematinius metodus, bei atvaizduoti juos grafiškai. Programinė įranga sukurta „Matlab“ aplinkoje. Šią įrangą pasirinkome nes ji turi pritaikytą matematinių paketų rinkinį, skirtą įvairioms matematinėms problemoms spręsti. Lengvai galėjome atlikti skaičiavimus, nesunkiai apdoroti duomenis matematiniais metodais ir gauti rezultatus bei juos pateikti įvairiuose grafikuose.

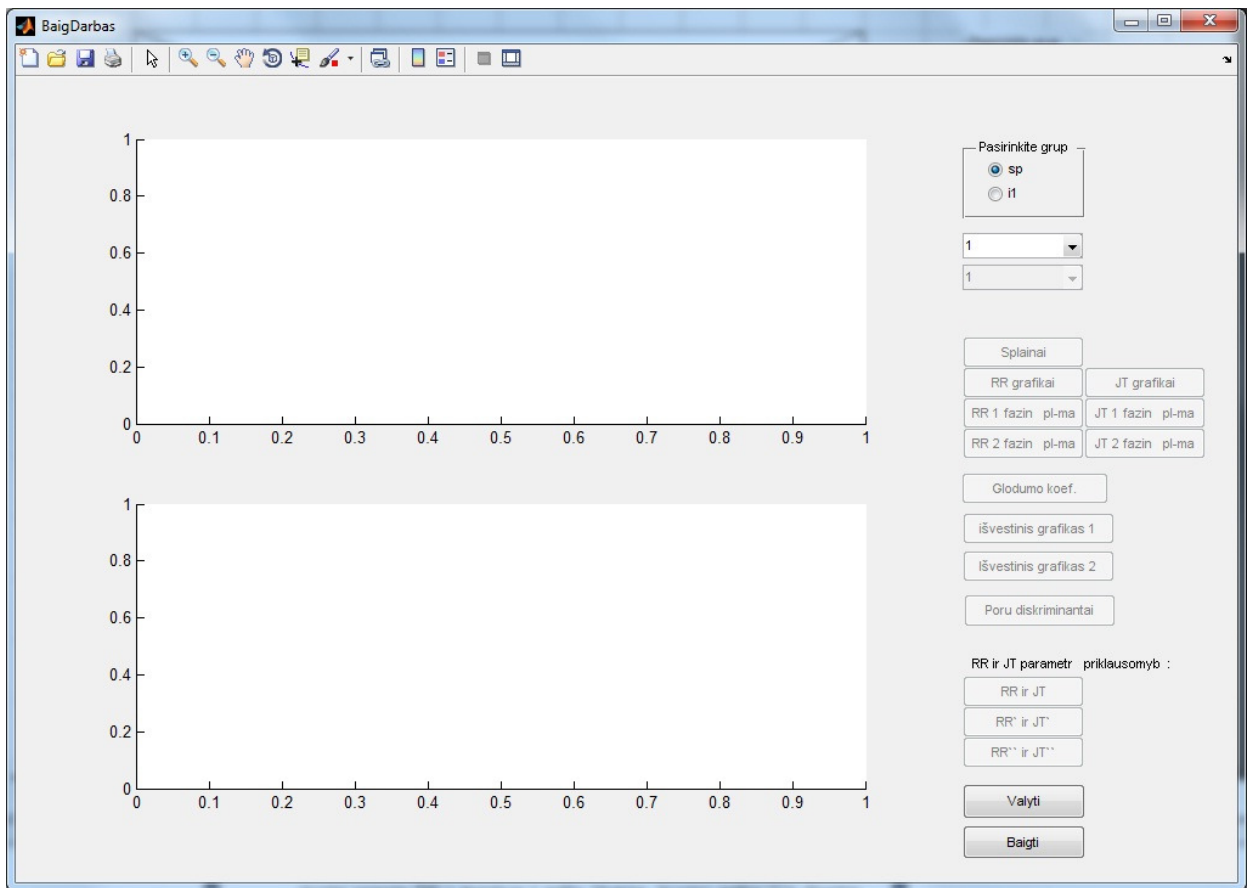
Sukūrėme vartotojo sąsajas, kurias patogiu naudoti. Programos langai pateikti 2.9 ir 2.10 paveikslėliuose.



2.9. pav. Programos langas

Tam, kad vartotojas galėtų sėkmingai pradėti dirbti su programa, mes padarėme taip, kad jis iš pradžių galėtų pasirinkti tik, kurios grupės asmens duomenis nori nagrinėti. Pasirinkęs tiriamąjį turi paspausti mygtuką „Tikrinti duomenis“. Šio mygtuko paspaudimu yra patikrinami ir normuojami duomenys. Mygtukas „Vykdyti“ pateikia „RR ir RR'“, „RR' ir RR'“, „JT ir JT'“, „JT' ir JT'“ fazinių plokštumų grafikus atitinkamose ašyse bei paskaičiuoja atitinkamų fazinių kreivių kitimo

greičio grafikus ir juos taip pat atvaizduoja atitinkamose ašyse. Mygtuku „Baigti“ vartotojas gali bet kuriuo metu baigti programos darbą.



2.10. pav. Programos langas

Vartotojas dirbdamas su šia programa taip pat gali pradėti darbą pradėdamas tik norimos grupės pasirinkimu. Pasirinkęs tiriamąjį turi paspausti mygtuką „Splainai“. Šis mygtukas duomenis interpoliuoja kubiniu splainu bei suskaičiuoja pirmąsias ir antrąsias išvestines. Paspaudus šį mygtuką, vėliau vartotojas gali spausti visus kitus mygtukus, atitinkamai toki, kokį grafiką jis nori pamatyti kompiuterio ekrane. Mygtukai „RR grafikai“ ir „JT grafikai“ pateikia abiejų parametrų grafikus atitinkamose ašyse. Mygtukai „RR 1 fazinė pl-ma“ ir „JT 1 fazinė pl-ma“ pateikia fazinių plokštumų RR ir RR išvestinės bei JT ir JT išvestinės kreives atitinkamuose ašyse. Mygtukai „RR 2 fazinė pl-ma“ ir „JT 2 fazinė pl-ma“ pateikia RR ir JT parametrų pirmųjų ir antrųjų išvestinių grafikus fazinėse plokštumose. Mygtukas „Glodumo koef.“ paskaičiuoja išvestinius glodumo koeficientus ir juos atvaizduoja grafike. Mygtukai „Išvestinis grafikas 1“ ir „Išvestinis grafikas 2“ atvaizduoja logaritminius EKG parametrų grafikus. Mygtukas „Poru diskriminantai“ pavaizduoja EKG parametrų diskriminantų grafikus atitinkamose ašyse. Galiausiai mygtukai „RR ir JT“, „RR` ir JT`“ ir „RR`` ir JT``“ atvaizduoja atitinkamas parametrų priklausomybes. Mygtuku „Baigti“ vartotojas gali bet kuriuo metu baigti programos darbą.

Gydytojai, norėdami detaliau ir giliau nagrinėti elektrokardiogramas, jų parametrų kitimus, galėtų pasinaudoti mūsų sukurta programa.

DISKUSIJA

Mūsų darbo tikslas buvo sukurti programinę realizaciją, kuri detaliau negu įprasta nagrinėtų EKG parametrus. Kadangi analizinių EKG parametrų tyrinėjimo metodu, kurie atskleistų esminius skirtumus tarp sveikų ir asmenų su tam tikrais širdies ir kraujagyslių nusiskundimais, nėra [19-29], sukūrėme programą, kuri gali išskirti sveikus asmenis iš nesveikų asmenų grupės.

Tyrėme EKG *JT* ir *RR* parametrus. Pradiniuose EKG parametrų užregistruotų veloergometrinių mėginių metu atskaitymuose jokių skirtumų tarp skirtingų grupių atstovų pastebėti negalima. Tyrimo eigoje sukūrėme tris metodus :

1. EKG parametrų kitimo greičio skaičiavimas ir vaizdavimas fazinėje plokštumoje;
2. EKG parametro glodumo koeficiento skaičiavimas ir vaizdavimas;
3. EKG parametro duomenų grupavimo metodas,

kurie gali išskirti visus asmenis į ligonių ir sveikų asmenų grupes.

Pirmajame metode pastebėjome EKG *JT* parametro fazinių plokštumų kitimo greičių skirtumus tarp ligonių ir sveikų asmenų grupių, antrajame EKG *JT* parametro glodumo koeficiento a , o trečiajame EKG *JT* parametro sudarytų porų diskriminantų vidurkių skirtumus. Norėdami įsitikinti gautų hipotezių teisingumu pritaikėme šių metodų rezultatams neparametrinį Manio-Vitnio-Vilkoksono testą (Z_M) su reikšmingumo lygmeniu $p = 0,05$. Gavome, kad gautos hipotezės yra teisingos, gautos p reikšmės buvo mažesnės už $0,05$. Tai yra EKG *JT* parametro fazinių plokštumų kitimo greičiai, EKG *JT* parametro glodumo koeficientai a ir EKG *JT* parametro sudarytų porų diskriminantų vidurkiai reikšmingai skiriasi tarp sveikų asmenų ir asmenų su tam tikrais kardiologiniais nusiskundimais grupių.

Patobulinus mūsų metodus, sukūrus reikiamą įrangą, tai galėtų būti reali priemonė gydytojams ligoninėse, poliklinikose ar ambulatorijose pastebėti kardiologines problemas dar ankstyvose jų stadijose, kurių realiai nėra įmanoma aptikti.

Atliktas darbas – tai tik dar vienas žingsnis, siekiant giliau nagrinėti žmogaus organizme vykstančius procesus. Gautus rezultatus reikėtų tikrinti su kitais duomenimis. Tai padėtų lengviau juos interpretuoti.

PADĖKA

Dėkoju Kauno Medicinos Universiteto profesoriui Alfonsui Vainorui už iškeltus kardiologinius uždavinius bei pagalbą juos sprendžiant. Taip pat dėkoju Kauno Technologijos Universiteto profesoriui Zenonui Navickui už visokeriopą pagalbą sprendžiant iškeltus uždavinius, patarimus bei matematinės žinias.

IŠVADOS

EKG parametrų kitimo greičio skaičiavimo ir vaizdavimo fazinėje plokštumoje metode patikimai išsiskyrė EKG JT parametro fazinių kreivių kitimo greičiai skirtingose grupėse.

EKG parametro glodumo koeficiento skaičiavimo ir vaizdavimo metode patikimai išsiskyrė EKG JT parametro glodumo koeficientai a sportininkų ir asmenų su tam tikrais nusiskundimais grupėse.

EKG parametro duomenų grupavimo metode patikimai išsiskyrė EKG JT parametro sugrupuotų porų diskriminantų vidurkiai duotose skirtingose grupėse.

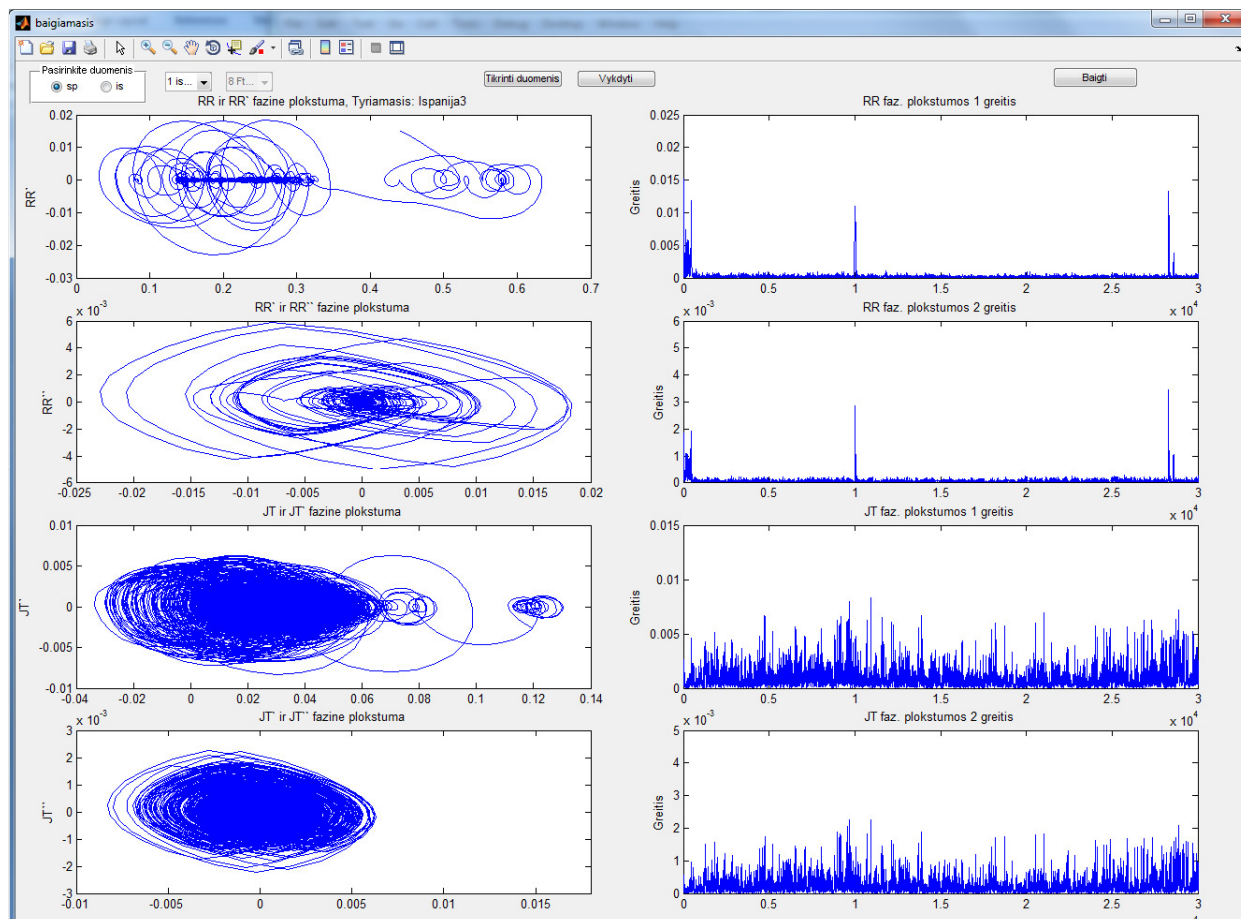
LITERATŪRA

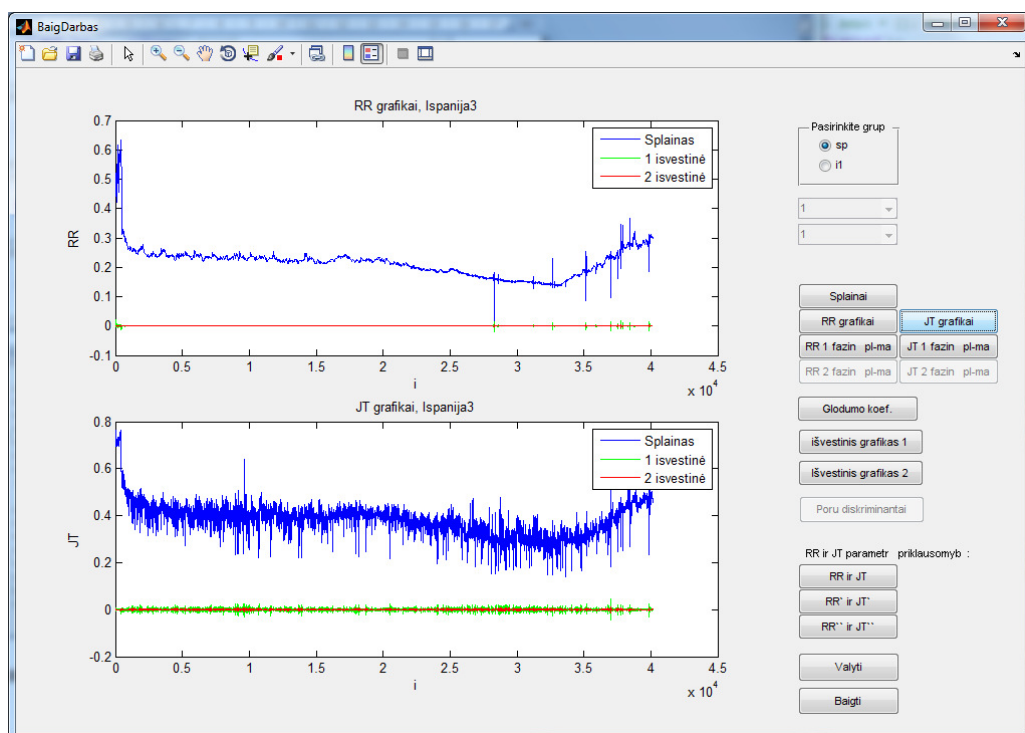
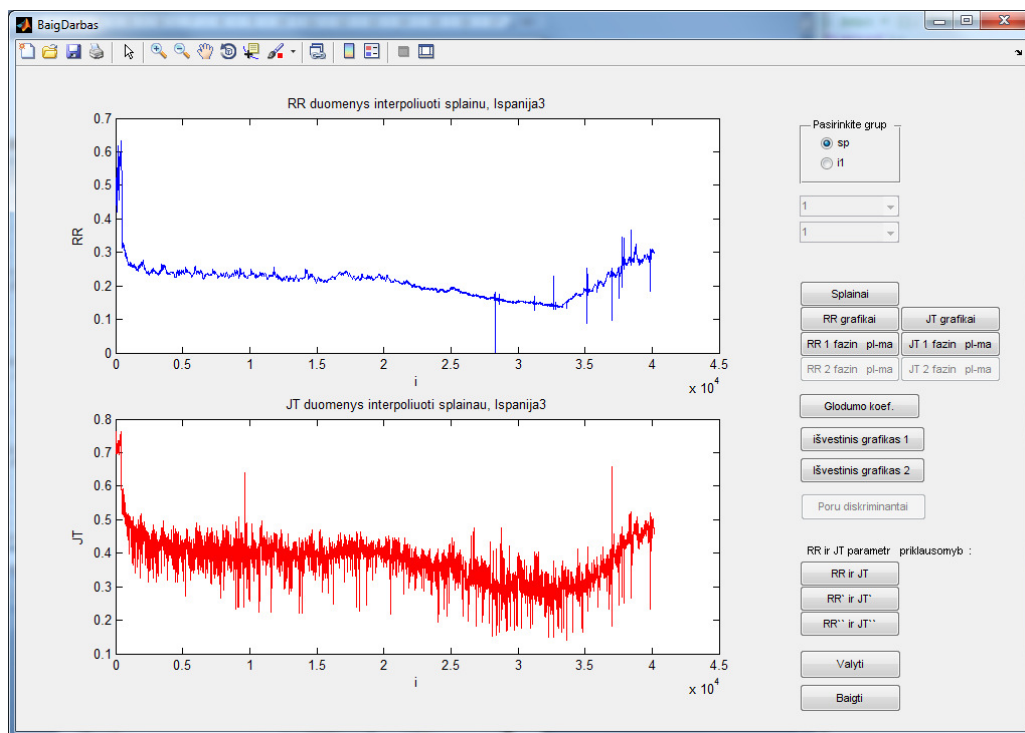
1. Ašeriškytė D. Maksimalaus išsvystomo galingumo prognozavimas fizinio krūvio metu / Ašeriškytė D., Matusevičiūtė K., Vainoras A., Navickas Z. // Biomedicininė inžinerija: tarptautinės konferencijos pranešimų medžiaga [Kauno technologijos universitetas, 2003 m. spalio 23,24 d.] Kaunas, 2003. p. 54 - 57.
2. Babarskienė R. Širdies ligos: universiteto vadovėlis // Išeminė širdies liga Kaunas, 2001. 352 p.
3. Berškienė K. Sistolinio kraujo spaudimo ir elektrokardiogramos ryšys / Berškienė K., Ašeriškytė D., Vainoras A., Navickas Z., Sedekerskienė V., Vitartaitė A.// Biomedicininė inžinerija: tarptautinės konferencijos pranešimų medžiaga [Kauno technologijos universitetas, 2004 m. spalio 28,29 d.] Kaunas, 2004. p. 180 - 184.
4. Berškienė K. Is it possible to evaluate the systolic arterial blood pressure by electrocardiographic parameters? / Berškienė K., Ašeriškytė D., Vainoras A., Navickas Z., Sedekerskienė V., Vitartaitė A. // IFMBE Proceedings: Medicon and Health Telematics 2004 „Health in the Information Society" Naples 2004, 308-8 no 522 [4p.].
5. Kvedaras, Bronius; Sapagovas, Mifodijus. Skaičiavimo metodai: vadovėlis respublikos aukštųjų mokyklų matematikos specialybei. Vilnius, 1974. 516 p.
6. Navickas, Zenonas, Bikulčienė, Liepa. Antros eilės matricų informatyvieji dėstiniai // Matematika ir matematinis modeliavimas / Kauno technologijos universitetas. Kaunas: Technologija. ISSN 1822-2757. T. 4 (2008), p. 26-33.
7. Matusevičiūtė K. Kelių sąveikaujančių žmogaus fiziologinių sistemų analizė ir modeliavimas / Matusevičiūtė K., Ašeriškytė D., Vainoras A., Navickas Z. // Biomedicininė inžinerija: tarptautinės konferencijos pranešimų medžiaga [Kauno technologijos universitetas, 2003 m. spalio 23,24 d.] Kaunas, 2003. p. 50 - 53.
8. Matusevičiūtė K. Sistolinio kraujo spaudimo prognozavimas pagal EKG parametrus / Matusevičiūtė K., Ašeriškytė D., Vainoras A., Navickas Z. // Taikomoji matematika: V studentų konferencijos pranešimų medžiaga [Kauno technologijos universitetas, 2004 m. gegužės 8 d.] Kaunas, 2004. p. 43 - 45.
9. Pekarskas, Vidmantas. Diferencialinis ir integralinis skaičiavimas: vadovėlis. Kaunas, 1997-2003 ISBN 9986-13-417-X.
10. Plukas, Kostas. Skaitiniai metodai ir algoritmai: vadovėlis aukštųjų mokyklų studentams. Kaunas, 2001. 549 p. ISBN 9955-03-061-5.
11. Šilanskienė A. Žmogaus organizmo funkcinės būklės kitimo ilgalaikių treniruočių metu vertinimas: daktaro disertacijos santrauka: biomedicinos mokslai: medicina (07B) / Aistė

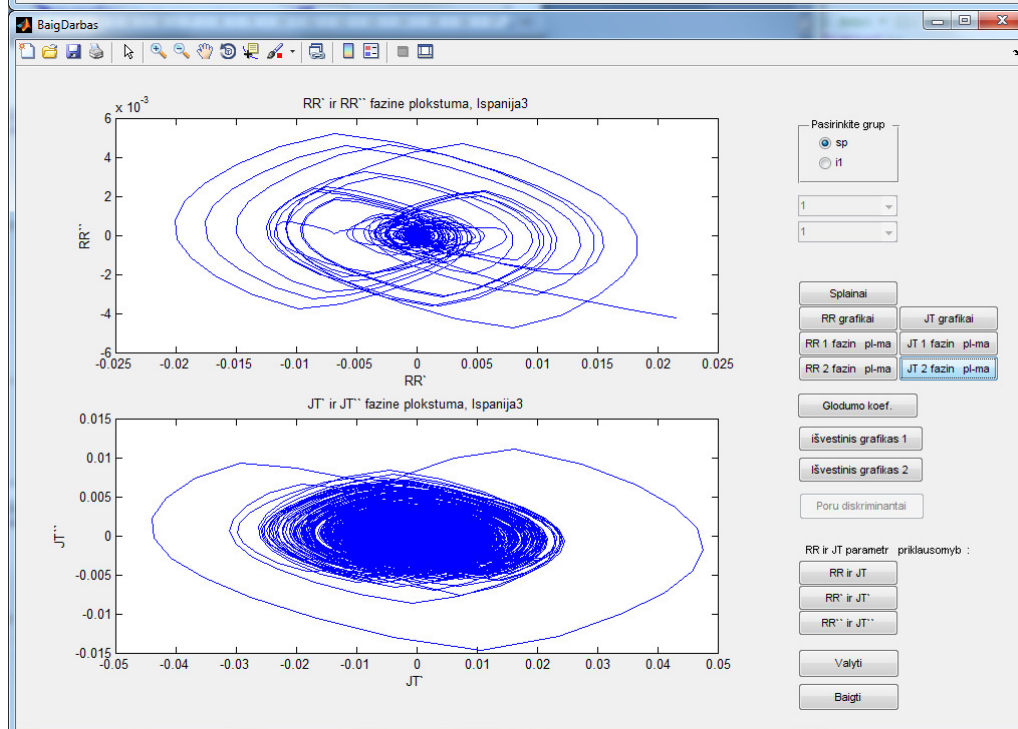
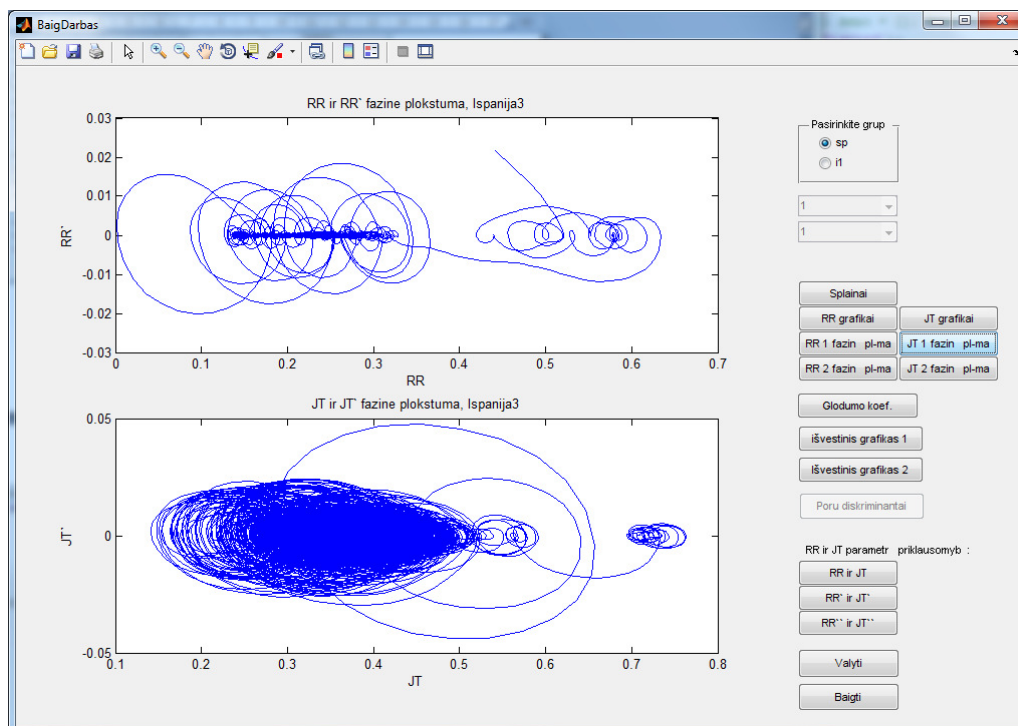
- Šilanskienė; Kauno medicinos universitetas. Kaunas, 2003. 32 p.
12. Valantinas, Jonas. Fraktalinė geometrija: vadovėlis. Kaunas, 1999. 186 psl. ISBN 9986-12 705-5.
 13. Valantinas, Jonas. Diskrečiosios transformacijos: vadovėlis. Technologija Kaunas, 2008. 88p. ISBN 978-9955-25-559-8.
 14. Išeminė širdies liga. Paruošė Vaitiekūnas V. Prieiga per internetą: <<http://kardiologas-grinius.w3.lt/AKA.htm>>.
 15. Išeminės širdies ligos diagnozė. Paruošė V. Grinius. Prieiga per internetą: <http://www.kvsc.lt/04_sveikata/c_gyv_sveikata/ischemine.htm>.
 16. Matematiniai metodai. Prieiga per internetą: <<http://mathworld.wolfram.com/>>.
 17. <<http://www.medicine.lt/straipsnis.asp?StraipsnioID=4654>>.
 18. Vainoras, A. ir kt. Veloergometrija ir sisteminių vertinimų galimybė. Lithuanian Journal of Cardiology, 1999 vol. 6, no 3, p. 760-763.
 19. Jaruševičius, G. ir kt. Changes of ventricular repolarization in acute myocardial infarction following coronary angioplasty. Computers in Cardiology, Institute of Electrical and Electronics Engineers. 2006, vol. 33, p. 453-455.
 20. Stankevičienė, A.; Petrauskienė, B. Tradiciniai išeminio miokardo gydymo būdai ir įrodymais pagrįsta medicina. Gydymo menas, Kaunas, Nr. 5(128), 2006, p. 7-9.
 21. Berškienė, K. ir kt. Interrelations of Electrocardiogram parameters. Elektronika ir elektrotechnika, 2009, 7(95), p. 95-98.
 22. Berškienė, K. ir kt. Netiesinių dinaminų fiziologinių sistemų sąveikos tyrimas. Biomedicininė inžinerija: tarptautinės konferencijos pranešimų medžiaga, spalio 23-24 d, 2008, Kaunas. Kaunas: Technologija, 2008, p. 38-41.
 23. Žumbakytė, R. Krepšininkų, futbolininkų ir nesportuojančiųjų atsigavimo proceso ypatybės po submaksimalaus krūvio mėginio. Ugdymas. Kūno kultūra. Sportas, 2006, No. 3(62), p. 59-64.
 24. Vainoras, A. et. al. Analysis of electric cardiac signals - methods and application results. Elektronika ir elektrotechnika, 2008, 5(85), p. 81-84.
 25. Gargasas, L. ir kt. Fiziologinių procesų stebėsenos ir širdies nepakankamumo diagnostikos kompiuterizuotų sistemų kūrimas. Elektronika ir elektrotechnika, 2005, 4(60), p.53- 57.
 26. Korsakas, S. et al. Personal Monitor for Evaluation of Human Functional State. Elektronika ir elektrotechnika, 2007, 6(78), p.61-66.
 27. Vainoras, A. Functional model of human organism reaction to load-evaluation of sportsman training effect. Education. Physical training. Sport, 2002, Nr. 3(44), p. 88-93.
 28. Marozas, V.; Jegelevičius, D. Biomedicinių signalų apdorojimas: vadovėlis. Kaunas, 2008. 195 p. ISBN 9786098007039.

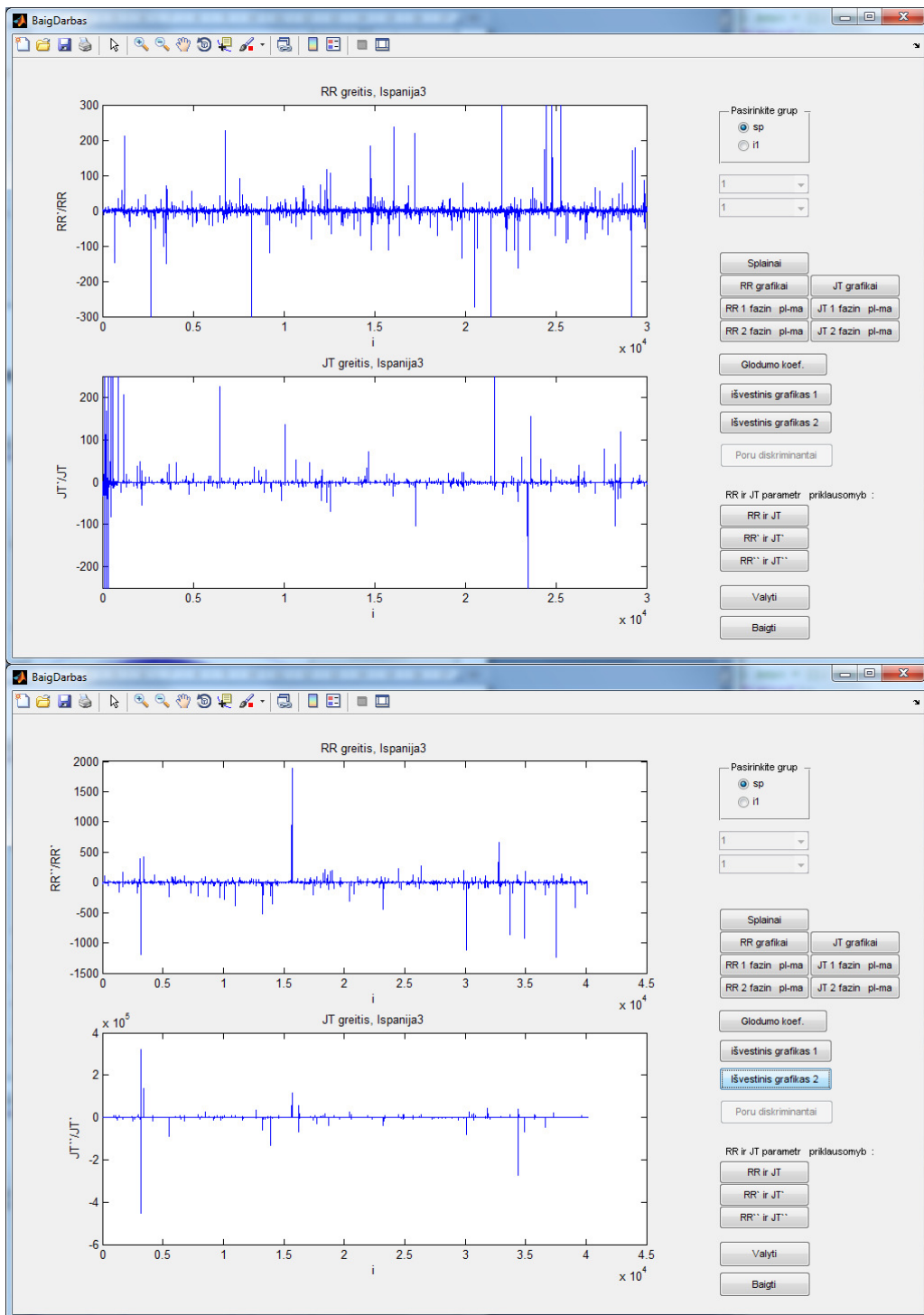
29. Acharya, R. et. al. Classification of cardiac abnormalities using heart rate signals. Medical and Biological Engineering and Computing. May, 2004; 42(3), p. 288-293.

1 PRIEDAS. VIENO ASMENS REZULTATAI









2 PRIEDAS. NEPARAMETRINIS MANIO-VITNIO-VILKOKSONO TESTAS

Test Statistics^b

	RR vid faz greitis	JT vid faz greitis
Mann-Whitney U	30,000	20,500
Wilcoxon W	58,000	111,500
Z	-1,228	-1,983
p	,219	,047

Exact Sig. [2*(1-tailed Sig.)]	,241 ^a	,046 ^a
--------------------------------	-------------------	-------------------

	glodumo koef a	glodumo koef b	20 poru vidurkis	log isvestines modulio vidurkis
Mann-Whitney U	19,000	42,000	17,500	6,000
Wilcoxon W	110,000	70,000	108,500	34,000
Z	-2,100	-,277	-2,220	-3,130
p	,036	,782	,026	,002
Exact Sig. [2*(1-tailed Sig.)]	,037 ^a	,817 ^a	,024 ^a	,001 ^a

a. Not corrected for ties.

b. Grouping Variable: Grupe

```
USE ALL.
COMPUTE filter_$=(Grupe=1).
VARIABLE LABEL filter_$ 'Grupe=1
(FILTER)'.
VALUE LABELS filter_$ 0 'Not Selected' 1 'Selected'.
FORMAT filter_$ (f1.0).
FILTER BY filter_$.
EXECUTE.
DESCRIPTIVES VARIABLES=RRvidfazgreitis JTvidfazgreitis glodumokoefa glodumokoefb
@20poruvidurkis logisvestinesmoduliovidurkis
/STATISTICS=MEAN SEMEAN.
```

Descriptive Statistics

	N	Mean	
		Statistic	Std. Error
RR vid faz greitis	7	,00050468	,000067160
JT vid faz greitis	7	,00087864	,000098088
glodumo koef a	7	-6,35660316	,141652072
glodumo koef b	7	-,17006443	,018947900
20 poru vidurkis	7	,00732857	,000767095
log isvestines modulio vidurkis	7	1,99964286	,141596642
Valid N (listwise)	7		

```
USE ALL.
COMPUTE filter_$=(Grupe=0).
VARIABLE LABEL filter_$ 'Grupe=0
(FILTER)'.
VALUE LABELS filter_$ 0 'Not Selected' 1 'Selected'.
FORMAT filter_$ (f1.0).
FILTER BY filter_$.
EXECUTE.
DESCRIPTIVES VARIABLES=RRvidfazgreitis JTvidfazgreitis glodumokoefa glodumokoefb
@20poruvidurkis logisvestinesmoduliovidurkis
/STATISTICS=MEAN SEMEAN.
```

Descriptive Statistics

	N	Mean	
		Statistic	Std. Error
RR vid faz greitis	13	,00065448	,000083260

JT vid faz greitis	13	,00065520	,000064256
glodumo koef a	13	-6,72846586	,081453567
glodumo koef b	13	-,15889522	,013609251
20 poru vidurkis	13	,00457692	,000728289
log isvestines modulio vidurkis	13	3,84546923	,597231817
Valid N (listwise)	13		

PRIEDAS. PROGRAMOS TEKSTAS

Baigiamasi.m

```
function varargout = baigiamasis(varargin)
% BAIGIAMASIS M-file for baigiamasis.fig
%   BAIGIAMASIS, by itself, creates a new BAIGIAMASIS or raises the existing
%   singleton*.
%
%   H = BAIGIAMASIS returns the handle to a new BAIGIAMASIS or the handle to
%   the existing singleton*.
%
%   BAIGIAMASIS('CALLBACK', hObject, eventData, handles,...) calls the local
%   function named CALLBACK in BAIGIAMASIS.M with the given input arguments.
%
%   BAIGIAMASIS('Property','Value',...) creates a new BAIGIAMASIS or raises the
%   existing singleton*. Starting from the left, property value pairs are
%   applied to the GUI before baigiamasis_OpeningFunction gets called. An
%   unrecognized property name or invalid value makes property application
%   stop. All inputs are passed to baigiamasis_OpeningFcn via varargin.
%
%   *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
%   instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Copyright 2002-2003 The MathWorks, Inc.

% Edit the above text to modify the response to help baigiamasis

% Last Modified by GUIDE v2.5 20-Mar-2009 00:40:43

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @baigiamasis_OpeningFcn, ...
                  'gui_OutputFcn',  @baigiamasis_OutputFcn, ...
                  'gui_LayoutFcn',   [] , ...
                  'gui_Callback',    []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargin
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
```

```

% --- Executes just before baigiamasis is made visible.
function baigiamasis_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to baigiamasis (see VARARGIN)

% Choose default command line output for baigiamasis
handles.output = hObject;

clc;
disp('Baigiamasis darbas');

set(handles.sp_popupmenu, 'Enable', 'on');
set(handles.il_popupmenu, 'Enable', 'off');
% set(handles.i2_popupmenu, 'Enable', 'off');
set(hObject, 'toolbar', 'figure');
cla(handles.axes1, 'reset'); % cleans the axes
cla(handles.axes2, 'reset'); % cleans the axes
cla(handles.axes3, 'reset'); % cleans the axes
cla(handles.axes4, 'reset'); % cleans the axes
cla(handles.axes5, 'reset'); % cleans the axes
cla(handles.axes6, 'reset'); % cleans the axes
cla(handles.axes7, 'reset'); % cleans the axes
cla(handles.axes8, 'reset'); % cleans the axes
set(handles.duomenys_pushbutton, 'Enable', 'off');
set(handles.vykdyti_pushbutton, 'Enable', 'off');

set(handles.duomenys_uipanel, 'SelectionChangeFcn', @duomenys_uipanel_SelectionChangeFcn); %??????

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes baigiamasis wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = baigiamasis_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in Duomenys_pushbutton2.
function duomenys_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to Duomenys_pushbutton2 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

Duomenys = [];
Duomenys = xlsread(handles.failas);
[handles.n handles.m] = size(Duomenys);

%%
for i = 1:handles.n

```



```

    if i < 4 % pirmiesiems trims
        for j = 1:3
            itasisduom = Duomenys(i,j);
            if (isnan(itasisduom))
                Duomenys(i,j) = Duomenys(i+1,j);
            end
        end
    else
        for j = 1:3
            itasisduom = Duomenys(i,j);
            if (isnan(itasisduom))
                Duomenys(i,j) = mean([Duomenys(i-3,j) Duomenys(i-2,j) Duomenys(i-
1,j)]);
            end
        end
    end
end;

%% RR
handles.RR = [];
handles.RR = Duomenys(:,2);

%% JT
handles.JT = [];
handles.JT = Duomenys(:,3);

%% Laikas
handles.T = [];
handles.T = Duomenys(:,1);

%% max ir min salygu patikrinimas, piku istaisymas RR parametrui
for i = 4:handles.n
    if ((handles.RR(i) < 140) || (handles.RR(i) > 1500)) || ((handles.RR(i) <
(handles.RR(i-1)*0.5)) || (handles.RR(i) > (handles.RR(i-1)*1.5)))
        handles.RR(i) = mean([handles.RR(i-3) handles.RR(i-2) handles.RR(i-1)]);
    else handles.RR(i) = handles.RR(i);
    end;
end;

%% max ir min salygu patikrinimas, piku istaisymas JT parametrui
for i = 4:handles.n
    if ((handles.JT(i) < 50) || (handles.JT(i) > 400)) || ((handles.JT(i) <
(handles.JT(i-1)*0.5)) || (handles.JT(i) > (handles.JT(i-1)*1.5)))
        handles.JT(i) = mean([handles.JT(i-3) handles.JT(i-2) handles.JT(i-1)]);
    else handles.JT(i) = handles.JT(i);
    end;
end;

%% normuojam RR
for i=1:handles.n
    handles.RR(i)=(handles.RR(i)-140)/(1500-140);
end;

%% normuojam JT
for i=1:handles.n
    handles.JT(i)=(handles.JT(i)-140)/(1500-140);
end;

%% ----splainai-----
cla(handles.axes1, 'reset'); % cleans the axes
cla(handles.axes2, 'reset'); % cleans the axes

```

```

delta = .1;
xx = 1:delta:handles.n-1;
x = 1:handles.n;

clear handles.rr_splainas handles.jt_splainas;

handles.rr_splainas = spline(x,handles.RR,xx);
handles.jt_splainas = spline(x,handles.JT,xx);

axes(handles.axes1);
plot(handles.rr_splainas, '-');
title(['RR duomenys interpoliuoti splainu, Tyriamasis: ', handles.asmuo]);
xlabel('i');
ylabel('RR');

axes(handles.axes2);
plot(handles.jt_splainas, 'r-');
title(['JT duomenys interpoliuoti splainau, Tyriamasis: ', handles.asmuo]);
xlabel('i');
ylabel('JT');

%% ----RR`-----
handles.rr_isvestine = [];
handles.rr_isvestine = diff(handles.rr_splainas,1,2);

%% ----JT`-----
handles.jt_isvestine = [];
handles.jt_isvestine = diff(handles.jt_splainas,1,2);

%% ----RR``-----
handles.rr_2_isvestine = [];
handles.rr_2_isvestine = diff(handles.rr_splainas,2,2);

%% ----JT``-----
handles.jt_2_isvestine = [];
handles.jt_2_isvestine = diff(handles.jt_splainas,2,2);

%% ----RR``-----
handles.rr_3_isvestine = [];
handles.rr_3_isvestine = diff(handles.rr_splainas,3,2);

%% ----JT``-----
handles.jt_3_isvestine = [];
handles.jt_3_isvestine = diff(handles.jt_splainas,3,2);

set(handles.vykdyti_pushbutton, 'Enable', 'on');
guidata(hObject, handles);

% --- Executes on button press in baigti_pushbutton.
function baigti_pushbutton_Callback(hObject, eventdata, handles)
%% hObject    handle to baigti_pushbutton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close all;

% --- Executes on button press in vykdyti_pushbutton.
function vykdyti_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to vykdyti_pushbutton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

```

```

%% ----RR fazine pl-ma 1-----
cla(handles.axes1, 'reset'); % cleans the axes
axes(handles.axes1);

handles.rr_splainas1 = [];
handles.rr_dydis = length(handles.rr_splainas);
for i = 1:handles.rr_dydis-1
    rr_splainas_tmp = handles.rr_splainas(i);
    handles.rr_splainas1 = [handles.rr_splainas1 rr_splainas_tmp];
end

plot(handles.rr_splainas1, handles.rr_isvestine, '-');
axis([0 0.7 -0.03 .02 ]);
title(['RR ir RR` fazine plokstuma, Tyriamasis: ', handles.asmuo]);
% xlabel('RR');
ylabel('RR`');

%% ----RR fazine pl-ma 2-----
cla(handles.axes3, 'reset'); % cleans the axes
axes(handles.axes3);

handles.rr_isvestinel1 = [];
for i = 1:handles.rr_dydis-2
    rr_isvestine_tmp = handles.rr_isvestine(i);
    handles.rr_isvestinel1 = [handles.rr_isvestinel1 rr_isvestine_tmp];
end

plot(handles.rr_isvestinel1, handles.rr_2_isvestine, '-');
axis([-0.025 .02 -0.006 .006 ]);
title('RR` ir RR`` fazine plokstuma');
% xlabel('RR`');
ylabel('RR``');

%% ----JT fazine pl-ma 1-----
cla(handles.axes5, 'reset'); % cleans the axes
axes(handles.axes5);

handles.jt_splainas1 = [];
handles.jt_dydis = length(handles.jt_splainas);
for i = 1:handles.jt_dydis-1
    jt_splainas_tmp = handles.jt_splainas(i);
    handles.jt_splainas1 = [handles.jt_splainas1 jt_splainas_tmp];
end

plot(handles.jt_splainas1, handles.jt_isvestine, '-');
axis([-0.04 0.14 -0.01 .01 ]);
title('JT ir JT` fazine plokstuma');
% xlabel('JT');
ylabel('JT`');

%% ----JT fazine pl-ma 2-----
cla(handles.axes7, 'reset'); % cleans the axes
axes(handles.axes7);

handles.jt_isvestinel1 = [];
for i=1:handles.jt_dydis-2
    jt_isvestine_tmp = handles.jt_isvestine(i);
    handles.jt_isvestinel1 = [handles.jt_isvestinel1 jt_isvestine_tmp];
end

```

```

plot(handles.jt_isvestine1, handles.jt_2_isvestine, '-');
axis([-0.010 .018 -0.003 .003 ]);
title ('JT` ir JT`` fazine plokstuma');
% xlabel('JT`');
ylabel('JT``');

%% ----RR fazines pl-mos 1 greitis-----
cla(handles.axes2, 'reset'); % cleans the axes
axes(handles.axes2);

handles.rr_lgreitis = [];
[q w] = size (handles.rr_2_isvestine);
xx = 1:w-1;
for i=1:w-1
    rr_lgreitis_tmp =
power((power(handles.rr_isvestine(i),2)+power(handles.rr_2_isvestine(i),2)),0.5);
    handles.rr_lgreitis = [handles.rr_lgreitis rr_lgreitis_tmp];
end

plot( xx, handles.rr_lgreitis, '-');
axis([0 30000 0 0.025 ]);
title ('RR faz. plokstumos 1 greitis');
% xlabel('i');
ylabel('Greitis');

%% ----RR fazines pl-mos 2 greitis-----
cla(handles.axes4, 'reset'); % cleans the axes
axes(handles.axes4);
handles.rr_2greitis = [];

[q w] = size (handles.rr_3_isvestine);
xx = 1:w-1;
for i=1:w-1
    rr_2greitis_tmp =
power((power(handles.rr_2_isvestine(i),2)+power(handles.rr_3_isvestine(i),2)),0.5)
;
    handles.rr_2greitis = [handles.rr_2greitis rr_2greitis_tmp];
    i = i+1000;
end

plot( xx, handles.rr_2greitis, '-');
axis([0 30000 0 0.006 ]);
title ('RR faz. plokstumos 2 greitis');
% xlabel('i');
ylabel('Greitis');

%% ----JT fazines pl-mos 1 greitis-----
cla(handles.axes6, 'reset'); % cleans the axes
axes(handles.axes6);

handles.jt_lgreitis = [];
[q w] = size (handles.jt_2_isvestine);
xx = 1:w-1;
for i=1:w-1
    jt_lgreitis_tmp =
power((power(handles.jt_isvestine(i),2)+power(handles.jt_2_isvestine(i),2)),0.5);
    handles.jt_lgreitis = [handles.jt_lgreitis jt_lgreitis_tmp];
end

plot( xx, handles.jt_lgreitis, '-');
axis([0 30000 0 0.015 ]);

```

```

title ('JT faz. plokstumos 1 greitis');
% xlabel('i');
ylabel('Greitis');

%% ----JT fazines pl-mos 2 greitis-----
cla(handles.axes8, 'reset'); % cleans the axes
axes(handles.axes8);
handles.jt_2greitis = [];

[q w] = size (handles.jt_3_isvestine);
xx = 1:w-1;
for i=1:w-1
    jt_2greitis_tmp =
power((power(handles.jt_2_isvestine(i),2)+power(handles.jt_3_isvestine(i),2)),0.5)
;
    handles.jt_2greitis = [handles.jt_2greitis jt_2greitis_tmp];
    i = i+1000;
end

plot( xx, handles.jt_2greitis, '-');
axis([0 30000 0 0.005 ]);
title ('JT faz. plokstumos 2 greitis');
% xlabel('i');
ylabel('Greitis');

%% RR fazines pl-mos 1 greitis vidutinis greitis
%handles.RR1vidutinisgreitis = 0;
RR1vidutinisgreitis = mean(handles.rr_1greitis);
disp(handles.asmuo);
disp(['RR 1 vidutinis greitis = ',RR1vidutinisgreitis]);

disp(RR1vidutinisgreitis);
%% RR fazines pl-mos 2 greitis vidutinis greitis
%handles.RR2vidutinisgreitis = 0;
RR2vidutinisgreitis = mean(handles.rr_2greitis);
disp(['RR 2 vidutinis greitis = ',RR2vidutinisgreitis]);
disp(RR2vidutinisgreitis);
%% JT fazines pl-mos 1 greitis vidutinis greitis
%handles.JT1vidutinisgreitis = 0;
JT1vidutinisgreitis = mean(handles.jt_1greitis);
disp(['JT 1 vidutinis greitis = ',JT1vidutinisgreitis]);
disp(JT1vidutinisgreitis);
%% JT fazines pl-mos 2 greitis vidutinis greitis
%handles.JT2vidutinisgreitis = 0;
JT2vidutinisgreitis = mean(handles.jt_2greitis);
disp(['JT 2 vidutinis greitis = ',JT2vidutinisgreitis]);
disp(JT2vidutinisgreitis);

saveas(gca, ['Pav_',handles.asmuo, '.jpg']); %int2str(deltaa),
guidata(hObject, handles);

% --- Executes on selection change in sp_popupmenu.
function sp_popupmenu_Callback(hObject, eventdata, handles)
%% hObject    handle to sp_popupmenu (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns sp_popupmenu contents as cell
array
%          contents{get(hObject,'Value')} returns selected item from sp_popupmenu

choice = get(handles.sp_popupmenu, 'Value');

```

```

switch choice
    case 1
        handles.failas = ('ispanija3.xls'); handles.asmuo = 'Ispanija3';
    case 2
        handles.failas = ('ispanija4.xls');handles.asmuo = 'Ispanija4';
    case 3
        handles.failas = ('ispanija5.xls');handles.asmuo = 'Ispanija5';
    case 4
        handles.failas = ('ispanija6.xls');handles.asmuo = 'Ispanija6';
    case 5
        handles.failas = ('ispanija7.xls');handles.asmuo = 'Ispanija7';
    case 6
        handles.failas = ('ispanija8.xls');handles.asmuo = 'Ispanija8';
    case 7
        handles.failas = ('ispanija9.xls');handles.asmuo = 'Ispanija9';
end;

set(handles.duomenys_pushbutton, 'Enable', 'on');
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function sp_popupmenu_CreateFcn(hObject, eventdata, handles)
%% hObject    handle to sp_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in sp_radiobutton.
function sp_radiobutton_Callback(hObject, eventdata, handles)
% hObject    handle to sp_radiobutton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of sp_radiobutton
% --- Executes on selection change in il_popupmenu.
function il_popupmenu_Callback(hObject, eventdata, handles)
%% hObject    handle to il_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns il_popupmenu contents as cell
array
%         contents{get(hObject,'Value')} returns selected item from il_popupmenu

choice = get(handles.il_popupmenu, 'Value');
switch choice
    case 1
        handles.failas = ('Volker.xls');handles.asmuo = 'Volker';
    case 2
        handles.failas = ('christian.xls');handles.asmuo = 'Christian';
    case 3
        handles.failas = ('ruth.xls');handles.asmuo = 'Ruth';
    case 4
        handles.failas = ('patrick.xls');handles.asmuo = 'Patrick';
    case 5
        handles.failas = ('wolfgang.xls');handles.asmuo = 'Wolfgang';

```

```

case 6
    handles.failas = ('mario.xls');handles.asmuo = 'Mario';
case 7
    handles.failas = ('steffen.xls');handles.asmuo = 'Steffen';
case 8
    handles.failas = ('ftritz.xls');handles.asmuo = 'Ftritz';
case 9
    handles.failas = ('alexander.xls');handles.asmuo = 'Alexander';
case 10
    handles.failas = ('Hoffmann.xls');handles.asmuo = 'Hoffmann';
case 11
    handles.failas = ('steffen.xls');handles.asmuo = 'Steffen';
case 12
    handles.failas = ('thomas.xls');handles.asmuo = 'Thomas';
case 13
    handles.failas = ('bernd.xls');handles.asmuo = 'Bernd';
case 14
    handles.failas = ('ressman.xls');handles.asmuo = 'Ressman';

end;

set(handles.duomenys_pushbutton, 'Enable', 'on');
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function i1_popupmenu_CreateFcn(hObject, eventdata, handles)
%% hObject    handle to i1_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on button press in i1_radiobutton.
function i1_radiobutton_Callback(hObject, eventdata, handles)
% hObject    handle to i1_radiobutton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of i1_radiobutton

% % --- Executes on selection change in i2_popupmenu.
% function i2_popupmenu_Callback(hObject, eventdata, handles)
% %% hObject    handle to i2_popupmenu (see GCBO)
% % eventdata  reserved - to be defined in a future version of MATLAB
% % handles    structure with handles and user data (see GUIDATA)
% %
% % Hints: contents = get(hObject,'String') returns i2_popupmenu contents as cell
array
% %         contents{get(hObject,'Value')} returns selected item from i2_popupmenu
% %
% %
% set(handles.duomenys_pushbutton, 'Enable', 'on');
% guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function i2_popupmenu_CreateFcn(hObject, eventdata, handles)

```

```

%% hObject    handle to i2_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     empty - handles not created until after all CreateFcns called

```

```

% Hint: popupmenu controls usually have a white background on Windows.
%       See ISPC and COMPUTER.

```

```

if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

```

```

% --- Executes on button press in i2_radiobutton.

```

```

function i2_radiobutton_Callback(hObject, eventdata, handles)
% hObject    handle to i2_radiobutton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles     structure with handles and user data (see GUIDATA)

```

```

% Hint: get(hObject,'Value') returns toggle state of i2_radiobutton

```

```

% -----
function duomenys_uipanel_SelectionChangeFcn(hObject, eventdata)
%% retrieve GUI data, i.e. the handles structure
handles = guidata(hObject);

```

```

switch get(eventdata.NewValue,'Tag') % Get Tag of selected object

```

```

    case 'sp_radiobutton'

```

```

        %execute this code when sp_radiobutton is selected

```

```

        set(handles.sp_popupmenu, 'Enable', 'on');

```

```

        set(handles.il_popupmenu, 'Enable', 'off');

```

```

%         set(handles.i2_popupmenu, 'Enable', 'off');

```

```

    case 'il_radiobutton'

```

```

        %execute this code when il_radiobutton is selected

```

```

        set(handles.sp_popupmenu, 'Enable', 'off');

```

```

        set(handles.il_popupmenu, 'Enable', 'on');

```

```

%         set(handles.i2_popupmenu, 'Enable', 'off');

```

```

    case 'i2_radiobutton'

```

```

        %execute this code when i2_radiobutton is selected

```

```

        set(handles.sp_popupmenu, 'Enable', 'off');

```

```

        set(handles.il_popupmenu, 'Enable', 'off');

```

```

%         set(handles.i2_popupmenu, 'Enable', 'on');

```

```

end;

```

```

set(handles.duomenys_pushbutton, 'Enable', 'off');

```

```

set(handles.vykdyti_pushbutton, 'Enable', 'off');

```

```

guidata(hObject, handles);

```

BaigDarbas.m

```

function varargout = BaigDarbas(varargin)

```

```

% BAIGDARBAS M-file for BaigDarbas.fig

```

```

% BAIGDARBAS, by itself, creates a new BAIGDARBAS or raises the existing
% singleton*.

```

```

%

```

```

% H = BAIGDARBAS returns the handle to a new BAIGDARBAS or the handle to
% the existing singleton*.

```

```

%

```

```

% BAIGDARBAS('CALLBACK',hObject,eventData,handles,...) calls the local
% function named CALLBACK in BAIGDARBAS.M with the given input arguments.

```

```

%

```

```

% BAIGDARBAS('Property','Value',...) creates a new BAIGDARBAS or raises the

```



```

%     existing singleton*. Starting from the left, property value pairs are
%     applied to the GUI before BaigDarbas_OpeningFunction gets called. An
%     unrecognized property name or invalid value makes property application
%     stop. All inputs are passed to BaigDarbas_OpeningFcn via varargin.
%
%     *See GUI Options on GUIDE's Tools menu. Choose "GUI allows only one
%     instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES
% Edit the above text to modify the response to help BaigDarbas
% Last Modified by GUIDE v2.5 19-May-2010 01:42:36

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                  'gui_Singleton',  gui_Singleton, ...
                  'gui_OpeningFcn', @BaigDarbas_OpeningFcn, ...
                  'gui_OutputFcn',  @BaigDarbas_OutputFcn, ...
                  'gui_LayoutFcn',  [] , ...
                  'gui_Callback',   []);
if nargin & isstr(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT

% --- Executes just before BaigDarbas is made visible.
function BaigDarbas_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to BaigDarbas (see VARARGIN)

% Choose default command line output for BaigDarbas
handles.output = hObject;
set(handles.sp_popupmenu, 'Enable', 'on');
set(handles.il_popupmenu, 'Enable', 'off');
% set(handles.i2_popupmenu, 'Enable', 'off');

set(handles.splainai_pushbutton, 'Enable', 'off');
set(handles.RRgrafikai_pushbutton, 'Enable', 'off');
set(handles.JTgrafikai_pushbutton, 'Enable', 'off');
set(handles.RR1fazPlokstuma_pushbutton, 'Enable', 'off');
set(handles.JT1fazPlokstuma_pushbutton, 'Enable', 'off');
set(handles.RR2fazPlokstuma_pushbutton, 'Enable', 'off');
set(handles.JT2fazPlokstuma_pushbutton, 'Enable', 'off');
set(handles.SSD_pushbutton, 'Enable', 'off');
set(handles.isvestinis1_pushbutton, 'Enable', 'off');
set(handles.isvestinis2_pushbutton, 'Enable', 'off');
set(handles.RR_JT_pushbutton, 'Enable', 'off');
set(handles.RR_JT_2_pushbutton, 'Enable', 'off');
set(handles.RR_JT_3_pushbutton, 'Enable', 'off');

set(hObject, 'toolbar', 'figure');
cla(handles.axes1, 'reset'); % cleans the axes
cla(handles.axes2, 'reset'); % cleans the axes
clc;

```

```

set(handles.pasirinktiGrupe_uipanel, 'SelectionChangeFcn', @pasirinktiGrupe_uipanel_
SelectionChangeFcn);

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes BaigDarbas wait for user response (see UIRESUME)
% uiwait(handles.figure1);

% --- Outputs from this function are returned to the command line.
function varargout = BaigDarbas_OutputFcn(hObject, eventdata, handles)
% varargout cell array for returning output args (see VARARGOUT);
% hObject handle to figure
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

% Get default command line output from handles structure
varargout{1} = handles.output;

% --- Executes on button press in splainai_pushbutton.
function splainai_pushbutton_Callback(hObject, eventdata, handles)
% hObject handle to splainai_pushbutton (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles structure with handles and user data (see GUIDATA)

Duomenys = [];
Duomenys = xlsread(handles.failas);
[handles.n handles.m] = size(Duomenys);

% tusciu laukeliu uzpildymas
for i = 1:2
    for j = 1:3
        itasisduom = Duomenys(i,j);
        if (isnan(itasisduom))
            Duomenys(i,1) = (Duomenys(i+1,1)+Duomenys(i-1,1))/2;
        end
    end
end

end;

% tusciu laukeliu uzpildymas
for i = 4:handles.n
    for j = 1:3
        itasisduom = Duomenys(i,j);
        if (isnan(itasisduom))
            Duomenys(i,j) = mean([Duomenys(i-3,j) Duomenys(i-2,j) Duomenys(i-
1,j)]);
        end
    end
end;

% max ir min salygu patikrinimas
for i = 5:handles.n
    % RR max, min patikrinimas
    itasisduom = Duomenys(i,2);
    if (itasisduom < 140) && (itasisduom > 1500)
        Duomenys(i,2) = mean([Duomenys(i-3,2) Duomenys(i-2,2) Duomenys(i-1,2)]);
    end
    % JT max, min patikrinimas
    itasisduom = Duomenys(i,3);
    if (itasisduom < 50) || (itasisduom > 400)

```

```

        Duomenys(i,3) = mean([Duomenys(i-3,3) Duomenys(i-2,3) Duomenys(i-1,3)]);
    end
end;

% piku, klaidu istaisymas
for i = 4:handles.n-1
    for j = 1:3
        itasisduom = Duomenys(i,j);
        if (itasisduom*0.5 > Duomenys(i+1,j)) || (itasisduom*1.5 <
Duomenys(i+1,j))
            Duomenys(i,j) = mean([Duomenys(i-3,j) Duomenys(i-2,j) Duomenys(i-
1,j)]);
        end
    end
end;

% Duomenų normavimas
for i=1:handles.n
    Duomenys(i,2)=(Duomenys(i,2)-140)/(1500-140);
    Duomenys(i,3)=(Duomenys(i,3)- 50)/( 400- 50);
end;

handles.RR = [];
x = 1:handles.n;
for i=1:handles.n
    tmp = Duomenys(i,2);
    handles.RR = [handles.RR tmp];
end
% JT
handles.JT = [];
x = 1:handles.n;
for j=1:handles.n
    tmp = Duomenys(j,3);
    handles.JT = [handles.JT tmp];
end
% Laikas
handles.T = [];
x = 1:handles.n;
for j=1:handles.n
    tmp = Duomenys(j,1);
    handles.T = [handles.T tmp];
end
%---splainai-----
cla(handles.axes1, 'reset'); % cleans the axes
cla(handles.axes2, 'reset'); % cleans the axes

axes(handles.axes1);
delta = .1;
xx = 2:delta:handles.n-1; % nuo 2 nes pirmas elementas 0
x = 1:handles.n;

handles.rr_splainas = [];
handles.jt_splainas = [];

handles.rr_splainas = spline(x,handles.RR,xx);
handles.jt_splainas = spline(x,handles.JT,xx);

plot(handles.rr_splainas, '-');
title(['RR duomenys interpoliuoti splainu, ', handles.asmuo]);
xlabel('i');
ylabel('RR');

```

```

axes(handles.axes2);
plot(handles.jt_splainas, 'r-');
title(['JT duomenys interpoliuoti splainau, ', handles.asmuo]);
xlabel('i');
ylabel('JT');

%----RR`-----
handles.rr_isvestine = [];
handles.rr_isvestine = diff(handles.rr_splainas,1,2);

%----JT`-----
handles.jt_isvestine = [];
handles.jt_isvestine = diff(handles.jt_splainas,1,2);

%----RR``-----
handles.rr_2_isvestine = [];
handles.rr_2_isvestine = diff(handles.rr_splainas,2,2);

%----JT``-----
handles.jt_2_isvestine = [];
handles.jt_2_isvestine = diff(handles.jt_splainas,2,2);

%----RR``-----
handles.rr_3_isvestine = [];
handles.rr_3_isvestine = diff(handles.rr_splainas,3,2);

%----JT``-----
handles.jt_3_isvestine = [];
handles.jt_3_isvestine = diff(handles.jt_splainas,3,2);

set(handles.RRgrafikai_pushbutton, 'Enable', 'on');
set(handles.JTgrafikai_pushbutton, 'Enable', 'on');
set(handles.RR1fazPlokstuma_pushbutton, 'Enable', 'on');
set(handles.JT1fazPlokstuma_pushbutton, 'Enable', 'on');
set(handles.SSD_pushbutton, 'Enable', 'on');
set(handles.RR_JT_pushbutton, 'Enable', 'on');
set(handles.RR_JT_2_pushbutton, 'Enable', 'on');
set(handles.RR_JT_3_pushbutton, 'Enable', 'on');
set(handles.isvestinis1_pushbutton, 'Enable', 'on');
set(handles.isvestinis2_pushbutton, 'Enable', 'on');
set(handles.sp_popupmenu, 'Enable', 'off');
set(handles.il_popupmenu, 'Enable', 'off');
%set(handles.i2_popupmenu, 'Enable', 'off');

guidata(hObject, handles);

% --- Executes on button press in RRgrafikai_pushbutton.
function RRgrafikai_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to RRgrafikai_pushbutton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
cla(handles.axes1, 'reset'); % cleans the axes
axes(handles.axes1);
plot(handles.rr_splainas, 'b-');
hold on;
plot(handles.rr_isvestine, 'g-');
hold on;
plot(handles.rr_2_isvestine, 'r-');
title(['RR grafikai, ', handles.asmuo]);
xlabel('i');
ylabel('RR');
legend('Splainas','1 isvestinė','2 isvestinė');

```

```

guidata(hObject, handles);

% --- Executes on button press in JTgrafikai_pushbutton.
function JTgrafikai_pushbutton_Callback(hObject, eventdata, handles)
% hObject      handle to JTgrafikai_pushbutton (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
cla(handles.axes2, 'reset'); % cleans the axes
axes(handles.axes2);
plot(handles.jt_splainas, 'b-');
hold on;
plot(handles.jt_isvestine, 'g-');
hold on;
plot(handles.jt_2_isvestine, 'r-');
title(['JT grafikai, ', handles.asmuo]);
xlabel('i');
ylabel('JT');
legend('Splainas', '1 isvestiné', '2 isvestiné');

guidata(hObject, handles);

% --- Executes on button press in RR1fazPlokstuma_pushbutton.
function RR1fazPlokstuma_pushbutton_Callback(hObject, eventdata, handles)
% hObject      handle to RR1fazPlokstuma_pushbutton (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
cla(handles.axes1, 'reset'); % cleans the axes
axes(handles.axes1);

handles.rr_splainas1 = [];
handles.rr_dydis = length(handles.rr_splainas);
for i=1:handles.rr_dydis-1
    rr_splainas_tmp = handles.rr_splainas(i);
    handles.rr_splainas1 = [handles.rr_splainas1 rr_splainas_tmp];
end

plot(handles.rr_splainas1, handles.rr_isvestine, '-');
title(['RR ir RR` fazine plokstuma, ', handles.asmuo]);
xlabel('RR');
ylabel('RR`');

set(handles.RR2fazPlokstuma_pushbutton, 'Enable', 'on');
guidata(hObject, handles);

% --- Executes on button press in JT1fazPlokstuma_pushbutton.
function JT1fazPlokstuma_pushbutton_Callback(hObject, eventdata, handles)
% hObject      handle to JT1fazPlokstuma_pushbutton (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
cla(handles.axes2, 'reset'); % cleans the axes
axes(handles.axes2);

handles.jt_splainas1 = [];
handles.jt_dydis = length(handles.jt_splainas);
for i=1:handles.jt_dydis-1
    jt_splainas_tmp = handles.jt_splainas(i);
    handles.jt_splainas1 = [handles.jt_splainas1 jt_splainas_tmp];
end

plot(handles.jt_splainas1, handles.jt_isvestine, '-');
title(['JT ir JT` fazine plokstuma, ', handles.asmuo]);

```

```

xlabel('JT');
ylabel('JT`');

set(handles.JT2fazPlokstuma_pushbutton, 'Enable', 'on');

guidata(hObject, handles);

% --- Executes on button press in RR2fazPlokstuma_pushbutton.
function RR2fazPlokstuma_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to RR2fazPlokstuma_pushbutton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
cla(handles.axes1, 'reset'); % cleans the axes
axes(handles.axes1);

handles.rr_isvestinel = [];
for i=1:handles.rr_dydis-2
    rr_isvestine_tmp = handles.rr_isvestine(i);
    handles.rr_isvestinel = [handles.rr_isvestinel rr_isvestine_tmp];
end

plot(handles.rr_isvestinel, handles.rr_2_isvestine, '-');
title(['RR` ir RR`` fazine plokstuma, ', handles.asmuo]);
xlabel('RR`');
ylabel('RR``');

guidata(hObject, handles);

% --- Executes on button press in JT2fazPlokstuma_pushbutton.
function JT2fazPlokstuma_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to JT2fazPlokstuma_pushbutton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
cla(handles.axes2, 'reset'); % cleans the axes
axes(handles.axes2);

handles.jt_isvestinel = [];
for i=1:handles.jt_dydis-2
    jt_isvestine_tmp = handles.jt_isvestine(i);
    handles.jt_isvestinel = [handles.jt_isvestinel jt_isvestine_tmp];
end

plot(handles.jt_isvestinel, handles.jt_2_isvestine, '-');
title(['JT` ir JT`` fazine plokstuma, ', handles.asmuo]);
xlabel('JT`');
ylabel('JT``');

guidata(hObject, handles);

% --- Executes on button press in SSD_pushbutton.
function SSD_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to SSD_pushbutton (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
cla(handles.axes1, 'reset'); % cleans the axes
axes(handles.axes1);
%set(handles.axes1, 'Position','width', 200);

handles.SSD= [];
xx = 2:handles.n;
for i=2:handles.n
    SSDtmp = 60/handles.RR(i);
    handles.SSD = [handles.SSD SSDtmp];

```

```

end;

plot( xx, handles.SSD, '-');
title(['ŠSD grafikas, ', handles.asmuo]);
xlabel('i');
ylabel('ŠSD');
cla(handles.axes2, 'reset'); % cleans the axes
set (handles.axes2, 'Visible', 'of');
guidata(hObject, handles);

% --- Executes on button press in valyti_pushbutton.
function valyti_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to valyti_pushbutton (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
cla(handles.axes1, 'reset'); % cleans the axes
cla(handles.axes2, 'reset'); % cleans the axes
guidata(hObject, handles);

% --- Executes on button press in baigti_pushbutton.
function baigti_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to baigti_pushbutton (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
close all;

% --- Executes on button press in RR_JT_pushbutton.
function RR_JT_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to RR_JT_pushbutton (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
cla(handles.axes1, 'reset'); % cleans the axes
cla(handles.axes2, 'reset'); % cleans the axes
set (handles.axes2, 'Visible', 'of');
axes(handles.axes1);

plot (handles.jt_splainas,handles.rr_splainas, '-');
title(['JT ir RR grafikas, ', handles.asmuo]);
xlabel('JT');
ylabel('RR');
guidata(hObject, handles);

% --- Executes on button press in RR_JT_2_pushbutton.
function RR_JT_2_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to RR_JT_2_pushbutton (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
cla(handles.axes1, 'reset'); % cleans the axes
cla(handles.axes2, 'reset'); % cleans the axes
set (handles.axes2, 'Visible', 'of');
axes(handles.axes1);

plot (handles.jt_isvestine,handles.rr_isvestine, '-');
title(['JT` ir RR` grafikas, ', handles.asmuo]);
xlabel('JT`');
ylabel('RR`');
guidata(hObject, handles);

% --- Executes on button press in RR_JT_3_pushbutton.
function RR_JT_3_pushbutton_Callback(hObject, eventdata, handles)
% hObject    handle to RR_JT_3_pushbutton (see GCBO)
% eventdata reserved - to be defined in a future version of MATLAB

```

```

% handles      structure with handles and user data (see GUIDATA)
cla(handles.axes1, 'reset'); % cleans the axes
cla(handles.axes2, 'reset'); % cleans the axes
set (handles.axes2, 'Visible', 'off');
axes(handles.axes1);

plot (handles.jt_2_isvestine,handles.rr_2_isvestine, '-');
title(['JT` ir RR` grafikas, ', handles.asmuo]);
xlabel('JT`');
ylabel('RR`');
guidata(hObject, handles);

% --- Executes on button press in isvestinis1_pushbutton.
function isvestinis1_pushbutton_Callback(hObject, eventdata, handles)
% hObject      handle to isvestinis1_pushbutton (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
cla(handles.axes1, 'reset'); % cleans the axes
cla(handles.axes2, 'reset'); % cleans the axes
%---atraktorius greitis RR ir RR` fazineje pl-moje-----
axes(handles.axes1);
greitis1 = [];
[q w] = size (handles.rr_isvestine);
xx = 1:w-1;
for i = 1:w-1
    greitis_tmp = (handles.rr_isvestine(i+1)-
handles.rr_isvestine(i))/(handles.rr_splainas(i+1)-handles.rr_splainas(i));
    greitis1 = [greitis1 greitis_tmp];

end
plot(xx, greitis1, '-');
title (['RR greitis, ', handles.asmuo]);
xlabel('i');
ylabel('RR`/RR');
axis([0 30000 -300 300 ]);
%---atraktorius greitis JT ir JT` fazineje pl-moje-----
axes(handles.axes2);
greitis2 = [];
[q w] = size (handles.jt_isvestine);
xx = 1:w-1;
for i = 1:w-1
    greitis_tmp = (handles.rr_isvestine(i+1)-
handles.jt_isvestine(i))/(handles.jt_splainas(i+1)-handles.jt_splainas(i));
    greitis2 = [greitis2 greitis_tmp];

end
plot(xx, greitis2, '-');
title (['JT greitis, ', handles.asmuo]);
xlabel('i');
ylabel('JT`/JT');
axis([0 30000 -250 250]);
saveas(gcf, ['Isvestinis_JT_', handles.asmuo, '.jpg']); %int2str(deltaa),

disp(handles.asmuo);
vidurkis = mean(abs(greitis2));
disp('vidurkis');
disp(vidurkis);

guidata(hObject, handles);

% --- Executes on button press in isvestinis2_pushbutton.

```



```

function isvestinis2_pushbutton_Callback(hObject, eventdata, handles)
% hObject      handle to isvestinis2_pushbutton (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)
cla(handles.axes1, 'reset'); % cleans the axes
cla(handles.axes2, 'reset'); % cleans the axes
%---atraktorius greitis RR` ir RR`` fazineje pl-moje-----
axes(handles.axes1);
greitis1 = [];
[q w] = size (handles.rr_2_isvestine);
xx = 1:w-1;
for i = 1:w-1
    greitis_tmp = (handles.rr_2_isvestine(i+1)-
handles.rr_2_isvestine(i))/(handles.rr_isvestine(i+1)-handles.rr_isvestine(i));
    greitis1 = [greitis1 greitis_tmp];

end
plot(xx, greitis1, '-');
title (['RR greitis, ', handles.asmuo]);
xlabel('i');
ylabel('RR`/RR`');
%---atraktorius greitis JT` ir JT`` fazineje pl-moje-----
axes(handles.axes2);
greitis2 = [];
[q w] = size (handles.jt_2_isvestine);
xx = 1:w-1;
for i = 1:w-1
    greitis_tmp = (handles.jt_2_isvestine(i+1)-
handles.jt_2_isvestine(i))/(handles.jt_isvestine(i+1)-handles.jt_isvestine(i));
    greitis2 = [greitis2 greitis_tmp];

end
plot(xx, greitis2, '-');
title (['JT greitis, ', handles.asmuo]);
xlabel('i');
ylabel('JT`/JT`');

guidata(hObject, handles);

% --- Executes on selection change in sp_popupmenu.
function sp_popupmenu_Callback(hObject, eventdata, handles)
% hObject      handle to sp_popupmenu (see GCBO)
% eventdata    reserved - to be defined in a future version of MATLAB
% handles      structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns sp_popupmenu contents as cell
array
%      contents{get(hObject,'Value')} returns selected item from sp_popupmenu
choice = get(handles.sp_popupmenu, 'Value');
switch choice
    case 1
        handles.failas = ('ispanija3.xls'); handles.asmuo = 'Ispanija3';
    case 2
        handles.failas = ('ispanija4.xls'); handles.asmuo = 'Ispanija4';
    case 3
        handles.failas = ('ispanija5.xls'); handles.asmuo = 'Ispanija5';
    case 4
        handles.failas = ('ispanija6.xls'); handles.asmuo = 'Ispanija6';
    case 5
        handles.failas = ('ispanija7.xls'); handles.asmuo = 'Ispanija7';
    case 6
        handles.failas = ('ispanija8.xls'); handles.asmuo = 'Ispanija8';
    case 7

```

```

        handles.failas = ('ispanija9.xls');handles.asmuo = 'Ispanija9';
end;

set(handles.splainai_pushbutton, 'Enable', 'on');
guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function sp_popupmenu_CreateFcn(hObject, eventdata, handles)
% hObject    handle to sp_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% --- Executes on selection change in il_popupmenu.
function il_popupmenu_Callback(hObject, eventdata, handles)
% hObject    handle to il_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = get(hObject,'String') returns il_popupmenu contents as cell
array
%         contents{get(hObject,'Value')} returns selected item from il_popupmenu
choice = get(handles.il_popupmenu, 'Value');
switch choice
    case 1
        handles.failas = ('Volker.xls');handles.asmuo = 'Volker';
    case 2
        handles.failas = ('christian.xls');handles.asmuo = 'Christian';
    case 3
        handles.failas = ('ruth.xls');handles.asmuo = 'Ruth';
    case 4
        handles.failas = ('patrick.xls');handles.asmuo = 'Patrick';
    case 5
        handles.failas = ('wolfgang.xls');handles.asmuo = 'Wolfgang';
    case 6
        handles.failas = ('mario.xls');handles.asmuo = 'Mario';
    case 7
        handles.failas = ('steffen.xls');handles.asmuo = 'Steffen';
    case 8
        handles.failas = ('ftritz.xls');handles.asmuo = 'Ftritz';
    case 9
        handles.failas = ('alexander.xls');handles.asmuo = 'Alexander';
    case 10
        handles.failas = ('Hoffmann.xls');handles.asmuo = 'Hoffmann';
    case 11
        handles.failas = ('steffen.xls');handles.asmuo = 'Steffen';
    case 12
        handles.failas = ('thomas.xls');handles.asmuo = 'Thomas';
    case 13
        handles.failas = ('bernd.xls');handles.asmuo = 'Bernd';
    case 14
        handles.failas = ('ressman.xls');handles.asmuo = 'Ressman';

end;

set(handles.splainai_pushbutton, 'Enable', 'on');

```

```

guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function i1_popupmenu_CreateFcn(hObject, eventdata, handles)
% hObject    handle to i1_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

% % --- Executes on selection change in i2_popupmenu.
% function i2_popupmenu_Callback(hObject, eventdata, handles)
% % hObject    handle to i2_popupmenu (see GCBO)
% % eventdata  reserved - to be defined in a future version of MATLAB
% % handles    structure with handles and user data (see GUIDATA)
% %
% % Hints: contents = get(hObject,'String') returns i2_popupmenu contents as cell
array
% %         contents{get(hObject,'Value')} returns selected item from i2_popupmenu
% choice = get(handles.i2_popupmenu,'Value');
% switch choice
%     case 1
%         handles.failas = ('thomas.xls');handles.asmuo = 'Thomas';
%     case 2
%         handles.failas = ('bernd.xls');handles.asmuo = 'Bernd';
%     case 3
%         handles.failas = ('ressman.xls');handles.asmuo = 'Ressman';
%     case 4
%         handles.failas = ('galinaitis.xls');handles.asmuo = 'Galinaitis';
%
% end;
%
% set (handles.splainai_pushbutton, 'Enable', 'on');
%
% guidata(hObject, handles);

% --- Executes during object creation, after setting all properties.
function i2_popupmenu_CreateFcn(hObject, eventdata, handles)
% hObject    handle to i2_popupmenu (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: popupmenu controls usually have a white background on Windows.
%         See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'),
get(0,'defaultUiControlBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

%-----
function pasirinktiGrupe_uipanel_SelectionChangeFcn(hObject, eventdata)

%retrieve GUI data, i.e. the handles structure
handles = guidata(hObject);

switch get(eventdata.NewValue,'Tag')    % Get Tag of selected object
    case 'sp_radiobutton'

```

```

%execute this code when sp_radiobutton is selected
set(handles.sp_popupmenu, 'Enable', 'on');
set(handles.il_popupmenu, 'Enable', 'off');
%
    set(handles.i2_popupmenu, 'Enable', 'off');

case 'il_radiobutton'
%execute this code when il_radiobutton is selected
set(handles.sp_popupmenu, 'Enable', 'off');
set(handles.il_popupmenu, 'Enable', 'on');
%
    set(handles.i2_popupmenu, 'Enable', 'off');

%
case 'i2_radiobutton'
%
%execute this code when i2_radiobutton is selected
%
    set(handles.sp_popupmenu, 'Enable', 'off');
%
    set(handles.il_popupmenu, 'Enable', 'off');
%
    set(handles.i2_popupmenu, 'Enable', 'on');
end;

set(handles.splainai_pushbutton, 'Enable', 'off');
set(handles.RRgrafikai_pushbutton, 'Enable', 'off');
set(handles.JTgrafikai_pushbutton, 'Enable', 'off');
set(handles.RR1fazPlokstuma_pushbutton, 'Enable', 'off');
set(handles.JT1fazPlokstuma_pushbutton, 'Enable', 'off');
set(handles.RR2fazPlokstuma_pushbutton, 'Enable', 'off');
set(handles.JT2fazPlokstuma_pushbutton, 'Enable', 'off');
set(handles.SSD_pushbutton, 'Enable', 'off');
set(handles.isvestinis1_pushbutton, 'Enable', 'off');
set(handles.isvestinis2_pushbutton, 'Enable', 'off');
set(handles.RR_JT_pushbutton, 'Enable', 'off');
set(handles.RR_JT_2_pushbutton, 'Enable', 'off');
set(handles.RR_JT_3_pushbutton, 'Enable', 'off');

guidata(hObject, handles);

```

Glodumas.m

```

clc;
sk = 0; Aspo = []; Bspo = []; Avok = []; Bvok = [];
disp('Started');
for i = 1:20

Duomenys = [];
if i == 1 Duomenys = xlsread('G:\!MAG\duom1\ispanija3.xls');asmuo =
'Ispanija3';
else if i == 2 Duomenys = xlsread('G:\!MAG\duom1\ispanija4.xls');asmuo =
'Ispanija4';
else if i == 3 Duomenys = xlsread('G:\!MAG\duom1\ispanija5.xls');asmuo =
'Ispanija5';
else if i == 4 Duomenys = xlsread('G:\!MAG\duom1\ispanija6.xls');asmuo =
'Ispanija6';
else if i == 5 Duomenys = xlsread('G:\!MAG\duom1\ispanija7.xls');asmuo =
'Ispanija7';
else if i == 6 Duomenys = xlsread('G:\!MAG\duom1\ispanija8.xls');asmuo =
'Ispanija8';
else if i == 7 Duomenys = xlsread('G:\!MAG\duom1\ispanija9.xls'); asmuo =
'Ispanija9';
% vokieciai:
else if i == 8 Duomenys = xlsread('G:\!MAG\duom1\alexander.xls');asmuo =
'Alexander';
else if i == 9 Duomenys = xlsread('G:\!MAG\duom1\bernd.xls');asmuo = 'Bernd';
else if i == 10 Duomenys = xlsread('G:\!MAG\duom1\christian.xls');asmuo =
'Christian';
else if i == 11 Duomenys = xlsread('G:\!MAG\duom1\ftritz.xls');asmuo = 'Ftritz';

```



```

% Fur. tr. matricos kurimas
F = [];
for k = 0:N-1 % k eiluciu, t.y. k duomenu
    for m = 0:N-1 % m stulpeliu, t.y. m koef sumuoti reikes, kad apskaiciuot viena
        k duomeni
            a = cos(2*pi*k*m / N);
            b = sin(2*pi*k*m / N);
            F(k+1,m+1) = complex(a,-b) ;
        end
    end;

% transformuoti duomenys - yDuomenys
yDuomenys = [];
yDuomenys = (1/N) * xDuomenys *F ;

% apskaiciuojam elementu apsolutinius didumus "absolute value"
YDuomenys = [];
for i = 1:N
    YDuomenys(i) = abs(yDuomenys(i));
end;

% logaritminame YDuomenis - zDuomenys
zDuomenys = [];
for i =1:N
    zDuomenys(i) = log(YDuomenys(i));
end;
%figure(2);grid;plot(zDuomenys);

% susikuriu naturaliuju sk vektoriu, kuri logaritmuoju tuo paciu abs
x = [];
for i = 1:N;
    x(i) = i;
    x(i) = log(x(i));
end;
%plot(x,zDuomenys, 'o');

% maziausiu kvadratu metodas
% MKM x(i)ir y:=zDuomenys(i)
sumax = 0;
sumaxx = 0;
sumaxy = 0;
sumay = 0;
% pasiruosiu sumas
for i = 1:N
    sumax = sumax + x(i);
    sumaxx = sumaxx + x(i)*x(i);
    sumaxy = sumaxy + x(i)*zDuomenys(i);
    sumay = sumay + zDuomenys(i);
end;

b = (N*sumaxy - sumay*sumax) / (N*sumaxx - sumax*sumax);
a = (sumay - b*sumax)/N;

%isimenam visus gautus a ir b
if failas >= 1 && failas <= 7
    Aspo(failas) = a;
    Bspo(failas) = b;
else
    Avok(failas) = a;
    Bvok(failas) = b;
end;

```

```
disp(failas);
disp(asmuo);
disp('a = ');disp(a);
disp('b = ');disp(b);
disp('end of one of files calculations');
end;

plot(Aspo,Bspo,'o','Color','g');
%axis([-0.1 1.4 -1 1]);
hold on;
plot(Avok,Bvok,'*','Color','r');
title('Glodumo koeficiento vaizdavimas');
xlabel('a');
ylabel('b');

disp('end');
```