



KAUNO TECHNOLOGIJOS UNIVERSITETAS
FUNDAMENTALIŲJŲ MOKSLŲ FAKULTETAS
TAIKOMOSIOS MATEMATIKOS KATEDRA

Kristina Berškienė

**INFORMACINĖS POSISTEMĖS,
APRAŠANČIOS INDIVIDUALIZUOTĄ
ŽMOGAUS FIZIOLOGINIŲ MODELĮ
FIZINIO KRŪVIO METU, KŪRIMAS**

Magistro darbas

Vadovai

prof. dr. Z. Navickas
prof. habil. dr. A. Vainoras

KAUNAS, 2005



KAUNO TECHNOLOGIJOS UNIVERSITETAS
FUNDAMENTALIŲJŲ MOKSLŲ FAKULTETAS
TAIKOMOSIOS MATEMATIKOS KATEDRA

TVIRTINU
Katedros vedėjas
prof. dr. J.Rimas
2005

**INFORMACINĖS POSISTEMĖS,
 APRAŠANČIOS INDIVIDUALIZUOTĄ
 ŽMOGAUS FIZIOLOGINĮ MODELĮ FIZINIO
 KRŪVIO METU, KŪRIMAS**

Taikomosios matematikos magistro baigiamasis darbas

Kalbos konsultantė
dr. J. Džežulskienė
2005

Vadovai
prof. dr. Z. Navickas
2005
prof. habil. dr. A. Vainoras(KMU)
2005

Recenzentai
prof. habil dr. G.Dzemyda (MII)
2005
prof. habil.dr. J.Poderys (LKKA)
2005

Atliko
FMMM – 3 gr. stud.
K. Berškienė
2005

KAUNAS, 2005

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, banko NORD/LB vyriausiasis analitikas

Zenonas Navickas, profesorius (KTU)

Arūnas Barauskas, UAB „Elsis“ generalinio direktoriaus pavaduotojas

Berškienė K. The development of information subsystem for individual human physiologic model during load: Master's work in Applied Mathematics/ supervisors prof. dr. Z. Navickas, prof. habil. dr. A. Vainoras; Department of Applied mathematics, Faculty of Fundamental Sciences, Kaunas University of Technology. – Kaunas, 2005. – 89 p.

SUMMARY

The statistics claim, that people who are not physically active live shorter and they are ill more often. Only an individual physical activity is useful. For choosing the right activity the doctors have to investigate the patient, analyse and store his data for the future.

The aim of this work was to describe mathematically the integral estimation model, using the model of physical load effect.

We created the information subsystem of human functional state model for individual person using the convolution of Meally and Moore automata. The result is the applied program for the consumer – physiologist.

This subsystem can be used in e – health for modeling the virtual doctor.

TURINYS

ĮVADAS.....	8
1. BENDROJI DALIS.....	10
1.1. Fiziniai pratimai, jų įtaka sveikatai ir darbingumui	10
1.2. Parametrai, vertinami veloergometriniu mėginio metu.....	12
1.3. Kompleksinis sveikatos būklės vertinimas	18
1.4. Formalizuotas fizinio krūvio poveikio modelis	21
1.5. Arterinio kraujospūdžio vertinimas fizinio krūvio metu.....	23
1.6. Paklaidos	24
1.7. Funkcijų aproksimavimas	25
1.7.1. Interpoliavimas	25
1.7.2. Suglodinimas.....	28
1.8. Skirtuminiai santykiai	31
1.9. Modeliavimas medicinoje – trumpa apžvalga.....	34
1.10. Milio ir Muro automatai.....	35
1.10.1. Milio automatas	36
1.10.2. Muro automatas.....	38
1.10.3. Milio ir Muro automatų sąsūkos pavyzdys	39
1.11. Programinė įranga	42
2. TIRIAMOJI DALIS	43
2.1. Informacinės posistemės apibūdinimas	43
2.2. Duomenys.....	43
2.3. Informacinės posistemės realizacija Milio ir Muro automatų sąsūka.....	45
2.4. Sistolinio kraujo spaudimo skaičiavimas	56
2.5. Programos aprašymas	57
IŠVADOS.....	61
LITERATŪRA.....	62
PRIEDAI	65
1 priedas. Integraliniai parametrai.....	65
2 priedas. Programos darbo langų pavyzdžiai.....	66
3 priedas. Programos teksto fragmentai	76

LENTELIŲ SĄRAŠAS

1.1 lentelė. Skirtuminiai santykiai.....	32
1.2 lentelė. Baigtiniai skirtumai	33
1.3 lentelė. Automato perėjimo lentelė	37
2.1 lentelė. Paciento duomenų pavyzdys	44
2.2 lentelė. Tirto kontingento duomenys.....	45
2.3 lentelė. Parametrai ir jų reikšmės, reiškiančios fizinių galimybių ribą	59

PAVEIKSLŲ SĄRAŠAS

1.1 pav. Fizinio aktyvumo ir sveikatos ryšys	11
1.2 pav. JT intervalas elektrokardiogramoje	13
1.3 pav. Širdies susitraukimo dažnio ir vystomo galingumo ryšys	17
1.4 pav. Kraujo tėkmės (ml/min) persiskirstymas organuose ramybės ir fizinio krūvio metu	19
1.5 pav. Integralios organizmo reakcijos į fizinių krūvių modelis	20
1.6 pav. Integralinio vertinimo modelis ir organizmo sistemų ryšius rodantys žymenys	21
1.7 pav. Fizinių pratimų poveikio modelis	22
1.8 pav. Tiesinio interpoliavimo polinomo grafiko pavyzdys	28
1.9 pav. Sistemos funkcionavimo diagrama	37
1.10 pav. Milio automato, aprašančio sumavimo operaciją, būsenų grafas	38
1.11 pav. Muro automato būsenų grafas	39
1.12 pav. Milio automato funkcionavimo diagrama	39
1.13 pav. Milio ir Muro automatų sąsūkos pavyzdys	40
1.14 pav. Sąsūkos darbo realizacija	40
1.15 pav. Aptarnavimo sistema	42
1.16 pav. Aptarnavimo sistemos formalusis aprašas	42
2.1 pav. Informacinės posistemės benrdoji principinė schema Milio ir Muro automatų sąsūka	46
2.2 pav. Informacinės posistemės darbo realizacija Milio ir Muro automatų sąsūka	47
2.3 pav. Sistolinio kraujo spaudimo dinamika	46

IŽANGA

Šių dienų statistika tvirtina, kad fiziškai neaktyvūs asmenys serga dažniau ir jų amžius yra trumpesnis, tačiau tai nereiškia, jog bet koks fizinis aktyvumas sergamumą sumažina. Nuo praeito amžiaus aštuntojo dešimtmečio pabaigos labai paplito įvairios fizinio aktyvumo formos. Visame pasaulyje veikia daug sportinių ar bendrojo fizinio aktyvumo klubų, kuriuose lankosi daug žmonių. Plintant šiems judėjimams, kyla vis daugiau problemų, susijusių su fizinio krūvio adekvatumu asmens būklei. Padėti gali tik sveikatos būklę atitinkantis fizinis aktyvumas. Šiam reikalavimui patenkinti būtina tinkamai kaupti ir nagrinėti konkretaus paciento duomenis, stebėti įvairių parametru, bei išvestinių dydžių dinamiką tolimesnių tyrimų metu, organizuoti grįžtamąjį ryšį tarp paciento būklės ir gydymo metodikos. Nuoseklus duomenų kaupimas ir analizė suteikia galimybę atsakyti kai kurių klinikinių dydžių matavimų, o tuos pačius rezultatus leidžia gauti iš kitų, tik žymiai paprastesnių tyrimų.

Elektroninės sveikatos (e – sveikatos) terminas nusako šiuolaikinių informacijos ir komunikacijos technologijų panaudojimą, tenkinant pacientų, sveikatos priežiūros specialistų, administratorių ir net politikų poreikius. Visuotinai priimtini yra keturi fundamentalūs e – sveikatos komponentai: klinikiniai tyrimai, sveikatos informacija, sveikatos priežiūros paslaugų modernizavimas ir sveikatos apsaugos profesinių žinių plėtra ir mokymas. Europos Sąjungos šalyse realizuojant įvairius e – sveikatos projektus lieka neišspręsta problema, kaip formalizuoti žmogaus, kaip dinaminės sistemos, modelį. Todėl daugelyje e – sveikatos strategijos projektų lieka neapimta virtualaus gydytojo pozicija, stengiantis integruoti visą gaunamą informaciją jau sveikatos apsaugos vieneto rėmuose (poliklinika, ligoninė).

Pagrindinis darbo tikslas – vadovaujantis integralios organizmo reakcijos į fizinį krūvį modeliu, sukurti ir formalizuoti individualiam asmeniui orientuotą informacinę posistemę.

Remdamiesi pirmaisiais trimis e – sveikatos komponentais, individualizuotu integralios organizmo reakcijos į fizinį krūvį ir formalizuotu fizinio krūvio poveikio modeliais, sukūrėme bei formalizavome Milio ir Muro automatų sąsūka (aprašėme matematinėmis informacinėmis sąvokomis), individualiam asmeniui orientuotą informacinę posistemę, kuri maksimaliai tenkina aptartus reikalavimus.

Sukurta informacinė posistemė apima paciento ir gydytojo lygmenis. Į protokolus įtraukti funkciniai tam tikrų fiziologinių parametru ryšiai, jų nustatymui naudotos MathCad ir SPSS programos. Pagal modelio formalizaciją parašyta taikomoji programa C++ Builder aplinkoje, atitinkanti gydytojų poreikius. Programoje įdiegti automatų protokolai skirti atskirų fiziologinių sistemų tiesioginių ir išvestinių parametru nagrinėjimui, klinikinių matavimų supaprastinimui,

adaptuoti žmogaus kaip dinaminės sistemos analizavimui, pasitelkiant virtualų gydytoją ir formuojant valdomą grįžtamąjį ryšį, formalizuojant gydymo procesą.

Su darbo tematika susijusi mokslinė veikla:

Pranešimai tarptautinėse konferencijose:

Biomedicininė inžinerija (1,2,3)

Medicon and Health telematics (4)

Pranešimai studentų konferencijoje:

Taikomoji matematika (5,6)

Pranešimai konferencijoje

Matematika ir matematikos dėstymas

„Informacinės sistemos e – sveikatai kūrimas, panaudojant Milio ir Muro automatų sąsūką“

„Informacinės dimensijos tyrimas, naudojant žmogaus organizmo fiziologinių funkcijų modelį“

Priimtas straipsnis į žurnalą „Medicina“

„Sistolinio kraujo spaudimo prognozavimo galimybė asmeniui, remiantis elektrokardiogramos parametrais“

Jaunųjų mokslininkų darbų parodoje – konkurse KTU TECHNORAMA' 2004 laimėta III – oji vieta už darbą „Fizinio krūvio efektų vertinimo metodikos“

1. BENDROJI DALIS

1.1. Fiziniai pratimai, jų įtaka sveikatai ir darbingumui

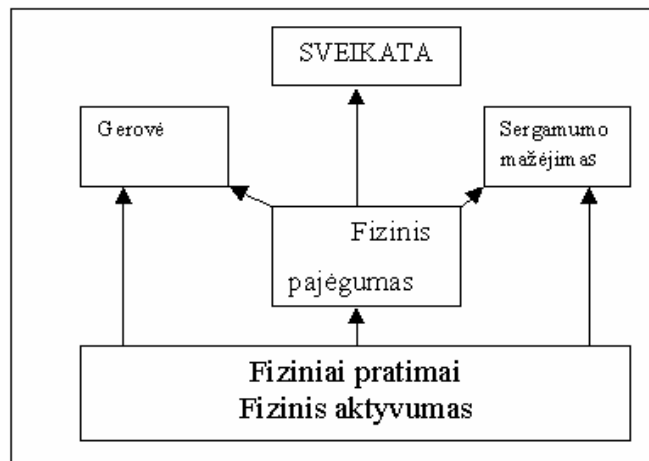
Pasaulinė sveikatos organizacija sveikatą apibūdina taip: „Sveikata – tai visapusiška individo, visuomenės fizinė, dvasinė ir socialinė gerovė”. Biomedicininis sveikatos apibūdinimas: „Sveikata yra žmogaus būklė, kurią sudaro socialinė ir psichinė dimensijos”. Tam tikri žmogaus fizinio pajėgumo komponentai turi savus pavadinimus. Fizinė ypatybė – tai žmogaus gebėjimas atlikti konkrečią judėjimo užduotį. Fizinės ypatybes lemia tokie veiksniai kaip žmogaus amžius, lytis, sveikatos būklė, gyvensena, fizinis aktyvumas.

Fizinis aktyvumas – visuma aktyvių judesių, kuriuos per tam tikrą laiką ir tam tikru intensyvumu atlieka raumenys ir sąnariai.

Fiziniai pratimai – sąmoningi, valingi žmogaus judesiai ir veiksmai, skirti fiziniam lavinimui ir gydymui.

Fizinis pajėgumas – tai gebėjimas patenkinamai atlikti fizinį darbą. Su sveikata susijusį fizinį pajėgumą galima suvokti kaip gebėjimų, sąveikaujančių su sveikata, derinį, kurį sudaro individualūs ypatumai ir funkcinės galimybės. Individualūs ypatumai dažniausiai yra genetiškai nulemti. Funkcinės galimybės susijusios su individo gebėjimais atlikti darbines užduotis. Žmogaus individualūs ypatumai lemia organizmo prisitaikymo prie fizinių krūvių dydį, o funkcinės galimybės – fizinio aktyvumo sukeltus pokyčius. Sveikatos ugdymo paskirtis – be rizikos didinant organizmo funkcinį pajėgumą, sukelti teigiamus sveikatos pokyčius.

Atliekant fizinius pratimus, aktyvėja beveik visų organizmo fiziologinių sistemų veikla. Didėjant organizmo, ypač aktyvių raumenų energijos poreikiui, aktyvėja energija aprūpinančių sistemų veikla. Organizmo adaptacija (prisitaikymas) prie fizinių krūvių vyksta dviem etapais. Krūvio metu, norint išlaikyti organizmo vidinės terpės pastovumą, vyksta greiti adaptaciniai pokyčiai, kurių metu mobilizuojami organizmo funkciniai, energetiniai bei plastiniai rezervai. Po krūvio ir reguliariai atliekant ištvermės pratimus, vyksta ilgalaikiai adaptaciniai pokyčiai. Tiek greitai, tiek ilgalaikiai adaptacijai būdingi bendrieji ir specialūs požymiai, t.y. didėja organizmo atsparumas ne tik fiziniam krūviui, bet ir kitiems aplinkos poveikiams. Dėl fizinių pratimų atsiradę adaptaciniai pokyčiai ne tik padidina organizmo funkcinę galimybę, bet ir mažina įvairių ligų riziką (žr. 1.1 pav.).



1.1 pav. Fizinio aktyvumo ir sveikatos ryšys

Ilgalaikės treniruotės sukelia adaptacinius organizmo pokyčius.

Griaučių raumenų sistemos pokyčiai:

- a) gerėja kraujotaka;
- b) didėja mioglobino kiekis;
- c) gerėja deguonies pasisavinimas;
- d) didėja metabolinis talpumas;
- e) susikaupia daugiau energetinių medžiagų;
- f) mažėja pagamintos pieno rūgšties kiekis.

Širdies ir kvėpavimo sistemos pokyčiai:

- a) didėja širdies ir kvėpavimo sistemų darbingumas (stiprėja širdies raumuo, greitėja jo susitraukimo greitis ir atsipalaidavimas);
- b) mažėja širdies susitraukimų dažnis;
- c) mažėja kraujo spaudimas;
- d) didėja širdies sistolinis ir minutinis kraujo tūris;
- e) didėja kraujo kiekis, eritrocitų ir hemoglobino kiekis;
- f) mažėja aterosklerozinių plokštelių kiekis arterijose;
- g) retėja, bet gilėja kvėpavimas.

Metaboliniai pokyčiai:

- a) mažėja riebalinio audinio kiekis;
- b) gerėja gliukozės pasisavinimas;
- c) gerėja kraujo lipidų sudėtis;
- d) apetitas atitinka žmogaus energijos poreikius.

1.2. Parametrai, vertinami veloergometriniu mėginio metu

Organizmo adaptacijos prie fizinių krūvių ypatumai ir galimybės bei adaptacijos procesų eigos operatyvi kontrolė yra aktualus sporto medicinos, klinikinės medicinos ir fiziologijos uždavinys. Šio uždavinio sprendimas neatskiriama susijęs su organizmo fiziologinės būsenos vertinimu (7,8,9). Šiuo metu plečiasi žmogaus organizmo kaip adaptyvios kompleksinės dinaminės sistemos samprata. Tokios sudėtingos sistemos funkcijų vertinimui reikalinga ir atitinkama diagnostinė sistema. Sportininko organizmo funkcionalumui, prisitaikymo galimybėms, adaptyvumui vertinti gana plačiai naudojami funkciniai mėginiai, kada organizmui suteikiamas tam tikras poveikis ir pagal jo atsaką bei reakciją sprendžiama apie organizmo būseną. Sporto medicinos, klinikinėje praktikoje vienu populiariausių tokio tipo tyrimų yra veloergometrija (10). Pastaroji naudojama kaip poveikio priemonė tam tikram, lengvai standartizuojamam, fiziniam aktyvumui realizuoti. Tiriamojo asmens reakcija į veloergometrinių krūvių sudėtinga, kompleksinė, daugelio sistemų funkcijas apimanti reakcija – tai ir raumenų veikla, būtina užduotam pajėgumui pasiekti, reguliacinių sistemų pokytis priderinant širdies ir kraujagyslių sistemą prie raumenų vykdomos funkcijos bei pačios širdies ir kraujagyslių sistemos pokytis pakankamai raumenų ir kitų sistemų hemodinamikai palaikyti. Aptarsime pagrindinius parametrus, gaunamus atlikus veloergometrinių mėginių:

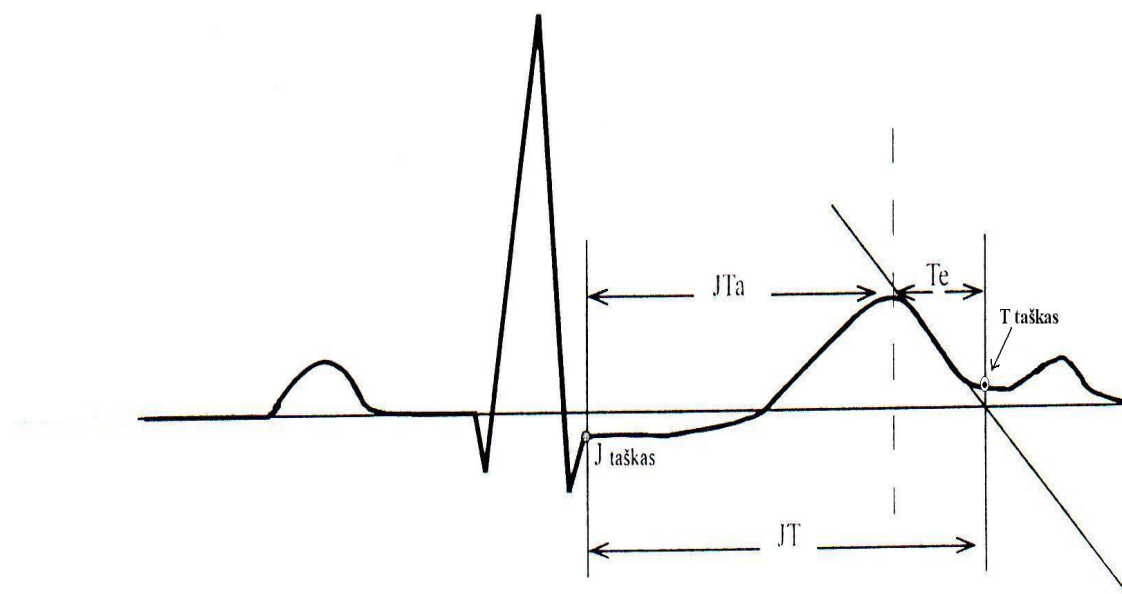
Iš elektrokardiogramos gaunamas širdies susitraukimų dažnis (*ŠSD*). Parametro kitimo ribos 20 – 250 kartų per minutę. Fizinio krūvio metu *ŠSD* didėja todėl, kad intensyviu širdies darbu, nes būtina patenkinti dirbančių raumenų poreikius. Po fizinio krūvio *ŠSD* ne iš karto grįžta į pradinį lygį. Atlikus 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.

Tiesiogiai matuojamas sistolinis arterinio kraujo spaudimas (*S*). Parametro kitimo ribos 80 – 250 mmHg. Fizinio krūvio metu padidėjus į širdį pritekančio veninio kraujo tūriui, padidėja galinis

diastolinis spaudimas skilvelyje, o taip pat sistolės metu išstumiamo kraujo kiekis. Miokardo susitraukimo galimybės ir spaudimas aortoje apsprendžia kraujo išstūmimo jėgą ir periferijos pasipriešinimą, kurį kraujas, tekėdamas arterijomis, turi nugalėti. Šie veiksniai turi įtakos sistolinio tūrio pokyčiams fizinio krūvio metu. Netreniruotų žmonių sistolinis kraujo tūris krūvio metu siekia 100 – 120 ml, sportuojančių žmonių padidėja iki 160 – 200 ml. Didėjant fizinio krūvio intensyvumui minutinis širdies tūris didėja ir dėl *ŠSD*. Treniruotos žmogaus širdies raumuo pasižymi didesnėmis susitraukimo galimybėmis, todėl sistolinis tūris didėja iki maksimalių žmogaus vystomų galingumų. Manoma, jog šis fiziologinis mechanizmas yra vienas iš svarbiausių sveikata stiprinančių pratimų efektų. Atlikta daug tyrimų, patvirtinančių ilgalaikių fizinių treniruočių įtaką sistolinio tūrio didėjimui tiek ramybės būsenoje, tiek fizinio krūvio metu.

Tiesiogiai matuojamas diastolinis arterinio kraujo spaudimas (*D*). Parametro kitimo ribos – 20 – 130 mmHg. Fizinio krūvio metu diastolinis arterinio kraujo spaudimas mažėja (tai įtakoja periferijos pasipriešinimas).

Intervalas elektrokardiogramoje nuo jungties taško J iki T bangos pabaigos (*JT* intervalas) – žr. 1.2 pav.



1. 2 pav. JT intervalas elektrokardiogramoje

Parametro kitimo ribos 0,14 – 0,36s. Fizinio krūvio metu šis intervalas trumpėja. *JT* intervalas apibūdina skilvelių repoliarizacijos trukmę. *JT* intervalas apsprendžia 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). *JT* intervalo pokyčiams įtakos turi reguliacinė nervų sistema. Yra žinoma, kad organizmo metaboliniai pokyčiai yra tamptariai 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 metaboline 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 (11).

ST amplitudė ($-ST$), parodanti kaip širdis aprūpinama krauju. ST pasislinkimo nuo izoelektrinės linijos neigiama amplitudė matuojama ST intervale 40 ms po JT taško ST sumuojant per 12 derivacijų. Parametro kitimo ribos nuo $-1,5$ iki 0 mV. Fizinio krūvio metu $-ST$ amplitudė mažėja. $-ST$ segmento pasislinkimas yra elektrokardiografinė miokardo išemijos išraiška. Fizinio krūvio mėginys yra svarbus klinikinio tyrimo metodas, kuris nusako širdies ir kraujagyslių sistemos funkcinę būklę, žmogaus fizinį pajėgumą bei išeminius širdies pokyčius. $-ST$ pokyčiai randami ir sveikiems žmonėms, kuriems nediagnozuota jokia širdies ir vainikinių kraujagyslių liga, tačiau iki šiol šių pokyčių genezė nėra tiksliai išaiškinta.

Veloergometrinio tyrimo metu išvystytas galingumas (N), matuojamas W. Tyrimo metu pasiektas maksimalus galingumas N_{max} , W.

Iš anksčiau aptartų parametrų skaičiuojami kiti – išvestiniai dydžiai. Aptarsime juos plačiau (10).

Laiko intervalas tarp dviejų širdies susitraukimų – RR intervalas (RR). Jis apskaičiuojamas tokiu būdu:

$$RR_i = \frac{60}{\check{SSD}_i}; \quad (1.1)$$

čia indeksas i reiškia, kad parametrai RR ir \check{SSD} imami prie to paties išvystyto galingumo N .

Parametras, siejantis reguliacinę ir aprūpinančiąją sistemas ir yra santykis JT/RR :

$$(JT/RR)_i = \frac{JT_i}{RR_i}; \quad (1.2)$$

čia indeksas i nurodo, kad visi formulėje pateikti parametrai gaunami esant tam pačiam galingumui N .

Ilgalaikių fizinių treniruočių metu sumažėja \check{SSD} ir bei sutrumpėja JT intervalas, t.y. širdžiai tenka mažesnis krūvis. Taigi santykis JT/RR mažėja. Vyrų širdys apytikriai trečdaliu yra didesnė už moterų, šis santykis mažesnis negu moterų. Kokios fiziologinės ypatybės lemia šias savybes, nėra žinoma (9).

Maksimalus širdies susitraukimų dažnis ($\check{S}SD_{max}$) kartais per minutę, laikomas žmogaus fizinių galimybių riba.

$$\check{S}SD_{max} = 220 - am\check{z}ius; \quad (1.3)$$

čia $am\check{z}ius$ matuojamas metais.

Kūno masės indeksas (KMI), lygus kūno masės, išreikštos kilogramais, ir ūgio, išreikšto metrais, kvadrato santykiui.

$$KMI = \frac{m}{h^2}; \quad (1.4)$$

čia m – kūno masė, kg;

h – ūgis, m.

Kūno masė laikoma normalia, jei KMI yra tarp 18 ir 26. Jeigu KMI didesnis už 26, žmogus turi antsvorio, jei KMI mažesnis negu 18 –svorio trūksta.

Organizmo reakciją į fizinį krūvį apibūdina žmogaus sveikatos suminis vertinimas (Sv). Jis apskaičiuojamas kaip Euklidinis nuotolis parametrų dN , dRR ir dJT erdvėje ir išreiškiamas procentais:

$$Sv = \frac{1}{k} \sqrt{dRR^2 + dJT^2 + dN^2} \cdot 100\%; \quad (1.5)$$

čia k – yra normalizuojantis parametras, priklausantis nuo tiriamojo lyties: vyrams 1,32, moterims 1,34;

dRR – RR intervalo pokytis, apskaičiuojamas taip:

$$dRR = \frac{RR_0 - RR_k}{RR_0 - \frac{60}{\check{S}SD_{max}}}; \quad (1.6)$$

čia RR_0 – laiko intervalas tarp dviejų širdies susitraukimų ramybės būsenoje, t.y. kai galingumas 0 W;

RR_k – laiko intervalas tarp dviejų širdies susitraukimų fizinio krūvio metu, kai yra pasiektas maksimalus galingumas.

$\check{S}SD_{max}$ – maksimalus širdies susitraukimų dažnis, apskaičiuojamas pagal (1.3) formulę

$dJT - JT$ intervalo sutrumpėjimas krūvio metu, apskaičiuojamas taip:

$$dJT = \frac{JT_0 - JT_k}{0,26}; \quad (1.7)$$

čia $JT_0 - JT$ intervalas, t.y. kai galingumas 0 W (ramybės būseną);

$JT_k - JT$ intervalas fizinio krūvio metu, kai yra pasiektas maksimalus galingumas.

dN – žmogaus išvystomo galingumo pokytis, skaičiuojamas pagal formulę:

$$dN = \frac{\sum_{i=1}^n N_i}{nN_{\max v}}; \quad (1.8)$$

čia n – skirtingų nelygių nuliui vystytų galingumų skaičius;

$N_i - i$ – asis žmogaus išvystytas galingumas;

$N_{\max v}$ – maksimalus išvystomas vidutinis galingumas, priklausantis nuo tiriamojo lyties ir fizinio pasirengimo. Profesionaliems sportininkams ir sveikiems vyrams $N_{\max v} = 200W$, moterims – 150W.

Pagal Sv reikšmę žmogaus sveikatos būklė gali būti klasifikuojama tokiu būdu:

- jei $0\% \leq Sv < 40\%$, yra ligos požymių;
- jei $40\% \leq Sv < 50\%$, ribinė sveikatos būklė;
- jei $50\% \leq Sv < 70\%$, sveikata normali;
- jei $70\% \leq Sv < 80\%$, sveikata gera;
- jei $80\% \leq Sv < 100\%$, puiki sveikatos būklė.

Parametras QRS – širdies skilvelių miokardo depolarizacija, matuojama sekundėmis.

Dviguba sandauga (DS):

$$DS = \check{SSD} \cdot S \cdot 10^{-2} \quad (1.9)$$

Galingumas, reikalingas organizmo funkcionalumui palaikyti – vidinis galingumas (N_{vid}), matuojamas W. Jis skaičiuojamas iš aproksimuotos tiesinės \check{SSD} ir N priklausomybės, kai širdies

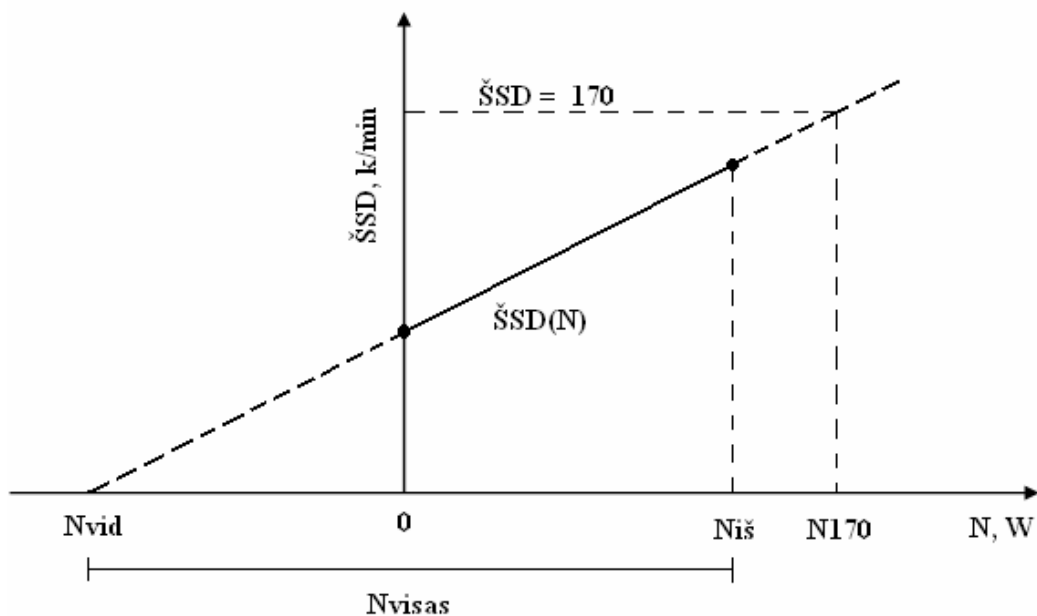
susitraukimų dažnis yra 0. Paprastai šis dydis perskaičiuojamas: apskaičiuojamas galingumas vatais, tenkančiais 1 kg asmens svorio, t.y. N_{vid} padalinamas iš tiriamojo svorio.

Išorinis galingumas ($N_{iš}$) – maksimalus išvystytas galingumas, išreikštas W. Šis dydis taip pat perskaičiuojamas: apskaičiuojamas galingumas vatais, tenkančiais 1 kg asmens svorio, t.y. $N_{iš}$ padalinamas iš tiriamojo svorio.

Širdies darbo našumo koeficientas ($n.k.$):

$$n.k. = \frac{N_{iš}}{N_{iš} + |N_{vid}|}. \quad (1.10)$$

Dydis $PWC170$, labiau žinomas sporte ir siejamas su širdies funkciniu pajėgumu. Jis apskaičiuojamas taip: iš aproksimuotos tiesinės $\check{S}SD$ ir N priklausomybės, kai $\check{S}SD = 170$ gaunamas ši širdies susitraukimo dažnį atitinkantis galingumas $N170$, kuris padalijamas iš asmens kūno svorio, kg (žr. 1.3 pav.).



1.3 pav. Širdies susitraukimo dažnio ir vystomo galingumo ryšys

Sistolinio kraujo spaudimo bei vystomo galingumo pokyčių santykis – dydis PPI :

$$PPI = \frac{1}{n} \sum_{i=1}^n \frac{S_{i+1} - S_i}{N_{i+1} - N_i} \quad (1.11)$$

Sistolinio spaudimo bei širdies susitraukimų dažnio pokyčių santykis – dydis PRI , sietinas su hipertenzinėmis reakcijomis į fizinį krūvį:

$$PRI = \frac{1}{n} \sum_{i=1}^n \frac{S_{i+1} - S_i}{\check{S}SD_{i+1} - \check{S}SD_i} \quad (1.12)$$

1.3. Kompleksinis sveikatos būklės vertinimas

Remiantis žmogaus ypatybėmis parenkamas individualus fizinis krūvis. Viena svarbiausių ypatybių yra amžius. Visą gyvenimo laikotarpį organizmas vystosi, kinta, prisitaiko prie įvairių gyvenimo sąlygų. Galima išskirti bent tris kokybiškai skirtingus etapus:

- a) organizmo augimas ir brendimas (sąlygiškai nuo 0 iki 30 metų);
- b) stabilus, nedidelių pokyčių etapas (nuo 30 iki 50 ar 60 metų);
- c) regresijos etapas (nuo 60 metų).

Panagrinėkime širdies ir kraujagyslių sistemos pagrindinės funkcijos aprūpinti krauju visus organus, t.y. hemodinamikos, ypatybes šiais gyvenimo etapais. Laikykime šią funkciją darbu, kurio reikia kraujo judėjimui organizme palaikyti:

$$A = f(\check{S}SD \times SV \times SYS); \quad (1.13)$$

čia A – darbas;

$\check{S}SD$ – širdies susitraukimų dažnis;

SV – sistolinis širdies tūris;

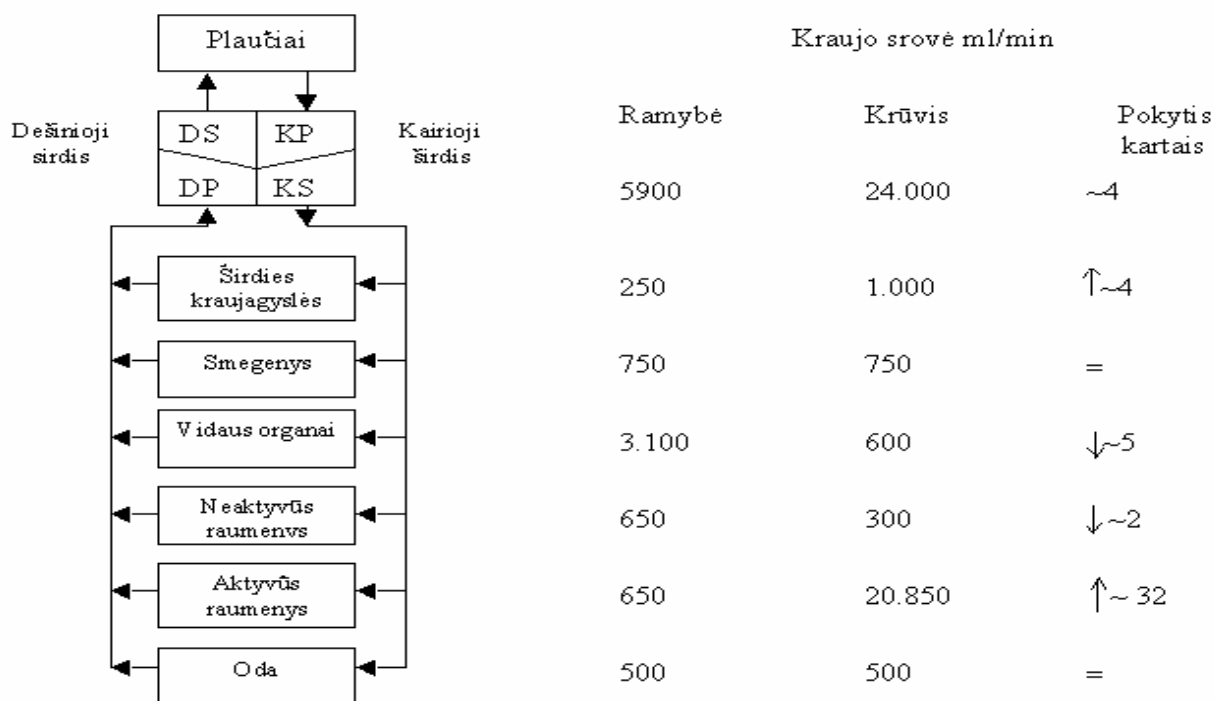
SYS – sistolinis kraujo spaudimas.

Ši lygtis betarpiškai susijusi su širdies atliekamu darbu, tekant kraujui per plaučius. Be abejo, ši lygtis nėra tiksli, nes darba, pernešant kraują, atlieka ne tik kairysis širdies skilvelis (tą rodo (1.13) formulė), bet ir kitos širdies kameros bei raumenys. Senstant jų santykis kinta, tačiau apytikriai jį galima nustatyti.

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 – (1.3) formulė: 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 SV 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 dėsniumus, subrendusiam organizmui reaguojant į fizinį aktyvumą (žr. 1.4 pav).



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

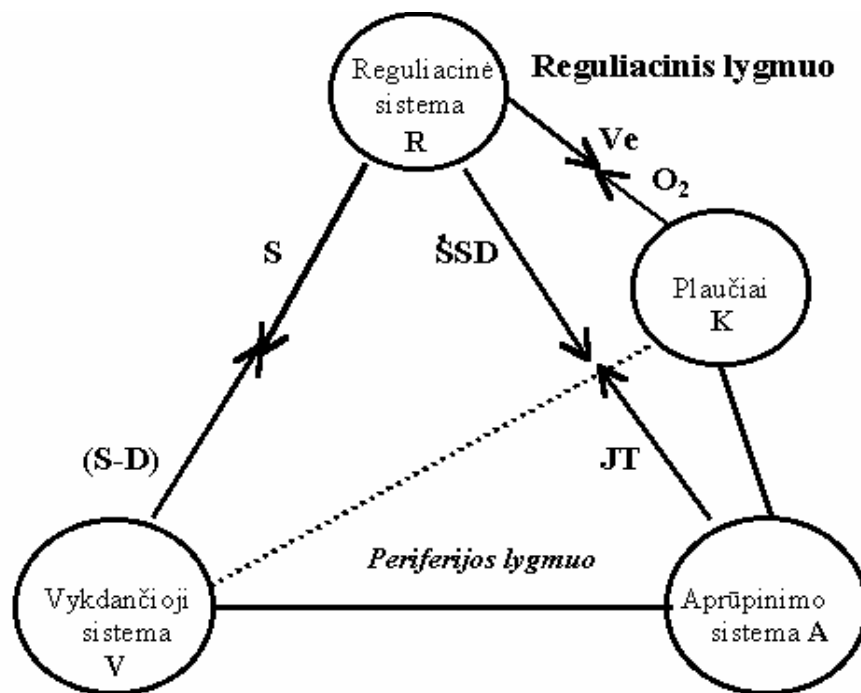
Pastebime, kad krūvio metu vyksta pokyčiai visame organizme, ne tik aktyviuose raumenyse. Visas organizmas prisitaiko prie kintančio krūvio. Skirtingose organų sistemose šie pokyčiai priešingų kryptių. Labiausiai stengiamasi padidinti kraujotaką per aktyvią raumenų grupę bei tą darbą lemiančių organų 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 persiskirstymo mechanizmą, kurio

pagrindinė paskirtis atlikti organizmo adaptaciją. Kuo didesnius pokyčius (aišku, neperžengdami tam tikrą fiziologinių pokyčių ribų) mes galime sukelti organizme, tuo prie didesnio krūvio organizmas gali prisitaikyti, t.y. organizmas bus veiklesnis, pajėgesnis.

Organizmui išvystant tam tikrą pajėgumą, jame vyksta kompleksinė, sisteminė reakcija, kurios pagrindinę dalį lemia aktyvinami elementai. Remdamiesi 1.4 paveikslu galime išskirti pagrindinius elementus:

- aktyvūs raumenys (V);
- Smegenys (R), kurie atlieka reguliacinę funkciją, apimdami visus reguliacinius lygmenis;
- širdies ir kraujagyslių sistema (A), aprūpinanti adekvačią hemodinamiką;
- kvėpavimo sistema (K), absorbuojanti deguonį bei išskirianti anglies dvideginį.

Šių keturių elementų funkciją galima apibendrinti kaip integralios organizmo reakcijos į fizinį krūvį modelį (1.5 pav.)



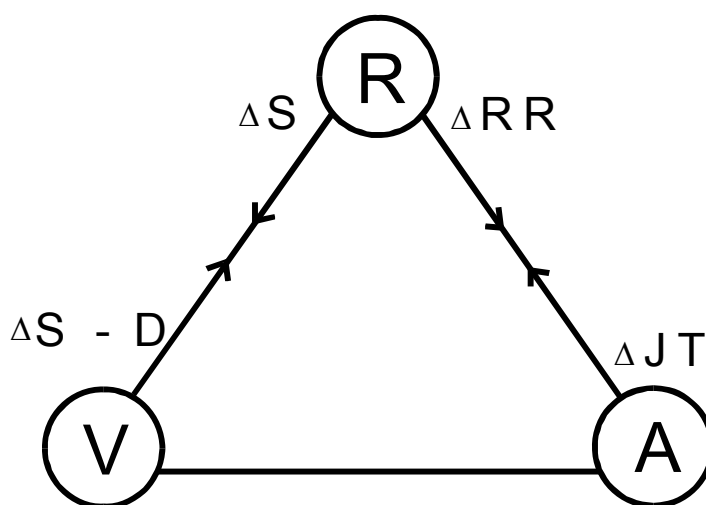
1.5 pav. Integralios organizmo reakcijos į fizinį krūvį modelis

Pagrindiniai modelio elementai: R – reguliacinė sistema, apimanti centrinę nervų sistemą, autonominių, humoralinių valdymą; V – veiklos sistema, fizinio aktyvumo metu – veiklių raumenų grupė; A – širdies ir kraujagyslių aprūpinimo sistema, tvarkanti centrinę hemodinamiką; K – kvėpavimo sistema, tvarkanti kraujo prisotinimą deguonimi.

Modelio parametrai: S – sistolinis arterinio kraujo spaudimas, D – diastolinis arterinio kraujo spaudimas, $\check{S}SD$ – širdies susitraukimų dažnis, JT – intervalas elektrokardiogramoje nuo jungties taško

J iki T bangos pabaigos (JT intervalas), V_e – įkvėpto oro tūris per minutę, O_2 – suvartoto deguonies tūris per minutę. Dydziai S , $\dot{S}SD$ ir V_e labiau sietini su reguliacija, o $S-D$ (sistolinio ir diastolinio kraujo spaudimo skirtumas), JT ir O_2 – su atskirų organų funkcija ir sukelia periferijos atsaką.

Santykius tarp šio modelio elementų, jų funkcijų, galima nusakyti daugeliu parametru. Modelyje nurodyti ir nežalojančiu būdu nustatomi, elementų funkciją atspindintys dydziai. Šį modelį iš dalies galima supaprastinti jungiant kvėpavimo ir širdies kraujagyslių sistemas į vieną aprūpinančią (A) sistemą. Per dešiniąją širdį, plaučius ir kairiąją širdį kraujas teka nuosekliai, nėra jokio praradimo ar papildymo, todėl hemodinamiką šioje organizmo dalyje lemia širdies funkcija ir kvėpavimo funkcija (absorbuoto deguonies kiekis). Taigi modelį galima supaprastinti (1.6 pav.).



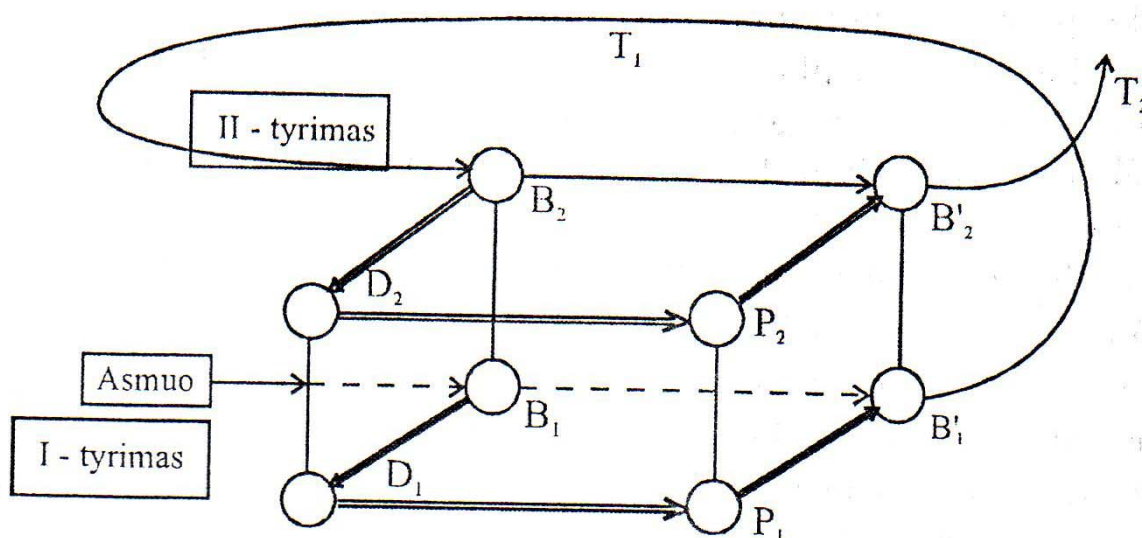
1.6 pav. Integralinio vertinimo modelis ir organizmo sistemų ryšius rodantys žymenys: V – vykdančioji sistema, R – reguliacinė sistema, A – aprūpinančioji sistema

1.4. Formalizuotas fizinio krūvio poveikio modelis

Vertindami pasirinktų parametru pokyčius, galime apibūdinti organizmo adaptaciją reikiamam krūviui atlikti. Tai galima priimti kaip matą integraliniams organizmo pokyčiams krūvio metu vertinti. Šio mato fiziologinė prasmė yra organizmo adaptacija. Jos dydis gali kompleksiskai parodyti organizmo veiklumą.

Veiklioji krūvio metu raumenų sistema vertinama pasiektu galingumu, pateikiamas jo normalizuotas vertinimas dN (1.8 formulė). Reguliacinė sistema vertinama intervalo RR normalizuotu pokyčiu, parodančiu širdies susitraukimo dažnį. Širdies veiklos intensyvėjimas krūvio metu nusakomas intervalo JT normalizuotu procentiniu pokyčiu. Šių trijų normalizuotų pokyčių įvertinimas Euklido

erdvėje kaip nuotolis ir sudaro suminį vertinimą S_v (1.5) formulė. Sveikiems asmenims šis dydis 60–70%, o esant patologijai jis labai sumažėja. Vertinant svarbu ne tik pats dydis, bet ir jį sudarančių parametų santykis. Suminio vertinimo pokyčiai gali parodyti organizmo reguliacinių procesų pokyčius, kurie patologinių reiškinių dar nesukelia, bet parodo, jog jie gali atsirasti. Remdamiesi modeliu ir atsižvelgdami į parametų pokyčius bei jų santykį, galime nustatyti ir tai, kokį organizmo atsaką gausime, taikydami jam vienokią ar kitokią poveikį (10). Šiam tikslui būtinas formalizuotas poveikio modelis (1.7 pav).



1.7 pav . Fizinio pratimo poveikio modelis

Modelio parametrai: B_1 – būklė prieš poveikį, B_2 – po jo; D_1 ir D_2 būklės vertinimas; P_1 ir P_2 – poveikių planai; B_1' ir B_2' – asmens būklės, lemiančios planų vykdymą po pirmojo ir antrojo tyrimo; T_1 – laiko tarpas per kurį įgyvendinamas pirmasis poveikio planas, intervalas tarp dviejų patikrinimų; T_2 – laiko tarpas, per kurį bus vykdomas antrasis poveikio planas.

Asmeniui pradėdant lankyti treniruotes ir pasiekus tam tikrą būklę B_1 , atliekami būtinieji jo būklės tyrimai. Po to jie vertinami, daroma išvada – diagnozė D_1 . Atsižvelgiant į išvadą, sudaromas poveikio planas P_1 , formuojama atitinkama asmens samprata, motyvacija planui vykdyti (paciento būklė B_1'). Sudarytas planas vykdomas laiko tarpą T_1 , per kurį pakinta paciento būklė (tampa B_2). Būklė B_2 po asmens tyrimo apibūdinama diagnoze D_2 , sudaromas naujas poveikio planas P_2 , formuojamas supratimo ir priėmimo pagrindas – būklė B_2' ir vykdomas naujas planas per laiką T_2 . Taikant šį modelį, reabilitaciją ar gydymą galima apibendrinti kaip funkciją, priklausančią nuo

nurodytų elementų, t.y. $F_{re}(B,D,P,B')$. Šios funkcijos kitimą galima nusakyti jos pokyčiu $dF_{re}(dB,dD,dP,dB')$. Funkcijos skirtumas dF_{re} nusako reabilitacijos rezultatą. Tai grįžtamasis reabilitacijos proceso ryšys, o jo kiekybinis vertinimas padeda kryptingai siekti teigiamo poveikio žmogaus organizmui.

Aptartas modelis tinkamas ne tik reabilitacijos proceso atveju, bet ir tuomet, kai norima pasiekti puikių sportinių rezultatų, išbandyti fizines galimybes.

1.5. Arterinio kraujospūdžio vertinimas fizinio krūvio metu

Žmogaus funkcinės galimybės glaudžiai susijusios su sistolinio arterinio kraujo spaudimo pokyčiais. Ramybės būsenoje šis parametras lengvai išmatuojamas, tačiau esant fiziniam krūviui išmatuoti sistolinį kraujo spaudimą yra labai sudėtinga (gaunamos didelės paklaidos), o kartais visai neįmanoma (12,13,14). Bendru atveju arterinį kraujo spaudimą galima išmatuoti invaziniais ir neinvaziniais metodais. Invazinis arterinio kraujo spaudimo matavimo metodas, nors ir labai tikslus, paprastai yra taikomas tik klinikos sąlygose. Tuo tarpu neinvaziniai matavimo metodai tinka naudojimui klinikoje ar poliklinikoje, tiriant ar stebint asmenų fiziologinę būseną fizinių pratimų atlikimo metu ar dirbant fizinį darbą. Šiuo metu yra žinomi tokie neinvazinio arterinio kraujo spaudimo matavimo metodai kaip osciliacinis, auskultacinis bei Doplerio. Matavimas Doplerio būdu daugiau taikomas klinikinėmis sąlygomis, nes tam reikalinga ultragarsinė, dažniausiai stacionari, aparatūra.

Gera žinoma parabolinis funkcinis ryšys tarp RR intervalo ir QT intervalo – didėjant širdies susitraukimo dažniui, QT intervalas trumpėja ir atvirkščiai. Šis faktas gali būti siejamas su reguliacinės sistemos aktyvumo padidėjimu, ko rezultate širdies metabolinis aktyvumas taip pat padidėja (15). Šiuos pokyčius gali atspindėti QT intervalas, arba mūsų darbe naudojamo JT intervalo kitimas. Tarkime reguliacinė sistema veikia keliomis kryptimis: viena, širdies kryptimi, kurią galime vertinti pagal širdies susitraukimo dažnio pokyčius, kita – simpatinės sistemos aktyvumo padidėjimas kraujagyslių sistemai, gali būti nusakomas sistolinio kraujo spaudimo padidėjimu (15,16,17). Akivaizdu, kad širdis, savo darbo metu, reaguos į abu faktorius: padidės širdies darbas bei metabolizmas miocituose.

Taigi labai svarbu ištirti sistolinio kraujo spaudimo apskaičiavimo galimybę konkrečiam asmeniui, naudojant tik elektrokardiogramos parametrus fizinio aktyvumo metu (3,4,5).

1.6. Paklaidos

Kompiuterių atsiradimas, spartus jų elementinės bazės ir programinės įrangos tobulėjimas verčia vis sudėtingesnius uždavinius „patikėti“ kompiuteriams. Tačiau pati skaičiavimo kompiuteriais specifiška lemia įvairaus pobūdžio paklaidas. Įvairių tipų kompiuteriuose skaičius skilčių, skirtų realiųjų skaičių mantisei bei eilei vaizduoti, gali būti skirtingas. Taigi sprendžiant tą patį uždavinį pagal tą patį algoritmą gali atsitikti taip, kad rezultatas skirtinguose kompiuteriuose bus skirtingas. Tas pats gali atsitikti ir naudojant skirtingą programinę įrangą (18,19).

Skaičiuojant dažniausiai susiduriama su trejopomis paklaidomis:

- a) pradinės informacijos paklaidomis,
- b) sprendimo metodo paklaidomis,
- c) apvalinimo paklaidomis.

Pradiniai skaičiavimo duomenys paprastai yra įvairių matavimų rezultatai, turintys paklaidas. Skaičiuojant kompiuteriu, be šių paklaidų, atsiranda papildomų paklaidų, nes kompiuteriuose vartojama dvejetainė skaičiavimo sistema ir mantisės skilčių skaičius yra baigtinis (18).

Sprendimo metodo paklaidos priklauso nuo uždavinio sprendimo metodo. Apvalinimo paklaidos priklauso nuo mantisei vaizduoti skirtų skilčių skaičiaus ir nuo skaičiavimo veiksmų.

Šiame darbe susiduriame su visų trijų rūšių paklaidomis. Pradinė informacija gauta matavimus registruojant kompiuteriu – jau čia neišvengtos ir apvalinimo paklaidos. Darbe apsiribosime sprendimo metodo paklaidų vertinimu – skaičiuosime paklaidas funkcijų aproksimavimui.

Skaičiaus absoliučioji paklaida yra tikslios ir apytikslės jo reikšmės skirtumas:

$$\Delta x = x - \bar{x} \quad ; \quad (1.14)$$

čia Δx – absoliučioji paklaida;

x – tiksli skaičiaus reikšmė;

\bar{x} – apytikslė skaičiaus reikšmė.

Praktikoje dažnai nepatogu vertinti absoliučiąją paklaidą – ypač kai tenka lyginti labai skirtingus dydžius. Tada skaičiuojama santykinė paklaida (δx) – absoliučiosios paklaidos ir apytikslės skaičiaus reikšmės santykis:

$$\delta x = \frac{\Delta x}{\bar{x}} \quad ; \quad (1.15)$$

Ši paklaida dažnai reiškiamą procentais (19).

1.7. Funkcijų aproksimavimas

Funkcijų aproksimavimo uždavinių gausu įvairiose matematikos, fizikos, biofizikos ir technikos srityse. Literatūroje nagrinėjama daugybė funkcijų aproksimavimo uždavinio sprendimo metodų (18). Tai paaiškinama tuo, kad praktikoje susiduriama su daugeliu šio uždavinio formuluočių.

Apibendrintą funkcijų aproksimavimo uždavinį galima suformuluoti taip: turint funkcijos $y = f(x)$ reikšmių lentelę (x_i, y_i) (čia $i = 0, n$, $y_i = f(x_i)$ ir $x_i \neq x_j$, kai $i \neq j$), reikia rasti funkcijos $f(x)$ reikšmę, kai $x = t$, o $t \in [x_0; x_n]$.

Funkcijos analizinė išraiška paprastai būna nežinoma arba labai sudėtinga. Mūsų darbe dažniausiai pasitaiko pirmasis variantas.

Atsižvelgiant į y_i reikšmių tikslumą, praktikoje taikomi du šio uždavinio sprendimo metodai:

- a) interpoliavimo metodas,
- b) suglodinimo metodas.

1.7.1. Interpoliavimas

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. Duomenis surašomi į lentelę, kuria išreiškiama funkcijos priklausomybė nuo argumento atskiruose taškuose. Gautoji lentelė naudojama teoriniam ar praktiniam reiškinio tyrimui (18,19).

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

$$F(x_i) = y_i, \quad i=0,1,\dots,n \quad (1.16)$$

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

Kaip matyti iš interpoliavimo uždavinio formuluotės, keičiant aproksimuojančiųjų funkcijų klasę K bei interpoliavimo sąlygas galima rasti įvairias interpoliacines išraiškas. Istoriskai pirmoji aproksimuojančiųjų funkcijų klasė buvo n – tosios eilės polinomų klasė. Pastaruoju metu plačiai naudojama splineų klasė – taip dažnai sumažinamas kompiuterinio skaičiavimo laikas (18).

Nors interpoliacinis polinomas yra vienintelis, tačiau istoriskai susiformavo nemažai jo išraiškų: Lagranžo, Aitkeno, Niutono, Gauso ir kt. Skaičiuojant kompiuteriais populiariausia yra Lagranžo interpoliacinė išraiška (18). Ją aptarsime plačiau.

Lagranžo interpoliacinė išraiška. Ieškosime n – tosios eilės daugianario $F(x)$ tenkinančio Lagranžo interpoliavimo sąlygas – (1.16 formulė). Nagrinėkime polinomus:

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

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

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

$$F(x) = \sum_{i=0}^n F_i(x)y_i. \quad (1.17)$$

Šis polinomas interpoliacinis, nes jis yra n – tosios eilės polinomas ir tenkina Lagranžo interpoliavimo sąlygas – (1.16) formulė – interpoliacinio polinomo išraiška vadinama Lagranžo interpoliacine išraiška, todėl literatūroje simbolis $F(x)$ keičiamas $L_n(x)$.

Jei atstumai tarp interpoliavimo taškų vienodi, t.y. $x_1 - x_0 = x_2 - x_1 = \dots = x_n - x_{n-1} = h$, tai Lagranžo interpoliacinį polinomą galima užrašyti kitokiu būdu. Šiuo atveju $x_k = x_0 + kh$. Pažymėję $x = x_0 + th$, $0 \leq t \leq n$, gauname

$$x - x_i = (t-i)h, \quad x_k - x_i = (k-i)h.$$

Vadinasi

$$F_k(x) = F(x_0 + th) = (-1)^{n-k} \frac{t(t-1)\dots(t-k+1)(t-k-1)\dots(t-n)}{k!(n-k)!} = (-1)^{n+k} \frac{t(t-1)\dots(t-n)}{n!(t-k)} \cdot \frac{n!}{k!(n-k)!}.$$

Pažymėję $C_n^k \cdot \frac{n!}{k!(n-k)!}$ ir įrašę $F_k(x_0+th)$ išraišką į (1.16) formulę, gauname

$$L_n(x) = L_n(x_0 + th) = (-1)^n \frac{t(t-1)\dots(t-n)}{n!} \cdot \sum_{k=0}^n (-1)^k C_n^k \frac{f(x_0 + kh)}{t-k}.$$

Niutono interpoliacinė išraiška. Pažymėkime

$$\omega_n = (x-x_0)(x-x_1)\dots(x-x_n) \quad (1.18)$$

$$L_n(x) = f(x_0) + f(x_0, x_1)\omega_0(x) + f(x_0, x_1, x_2)\omega_1(x) + \dots + f(x_0, x_1, \dots, x_n)\omega_{n-1}(x).$$

Šis polinomas vadinamas Niutono interpoliacine išraiška.

Tiesinis interpoliavimas. Tiesinio interpoliavimo formulė netiksli, tačiau dažnai taikoma praktikoje, kai tikslios interpoliacinės išraiškos yra nereikalingos, sunkiai apskaičiuojamos ar tiesiog nėra galimybių paraktiškai patikrinti, kokios paklaidos gaunamos tarpiniuose taškuose, naudojant vieną ar kitą interpoliacinę išraišką (18). Šio darbo programinėje įrangoje tiesinis interpoliavimas taikomas tais atvejais, kai tikslios interpoliacinės išraiškos nėra reikalingos tolimesniems skaičiavimams.

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 argumentu ir funkcijos reikšmėmis.

Užrašykime Lagranžo interpoliacinį polinomą:

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

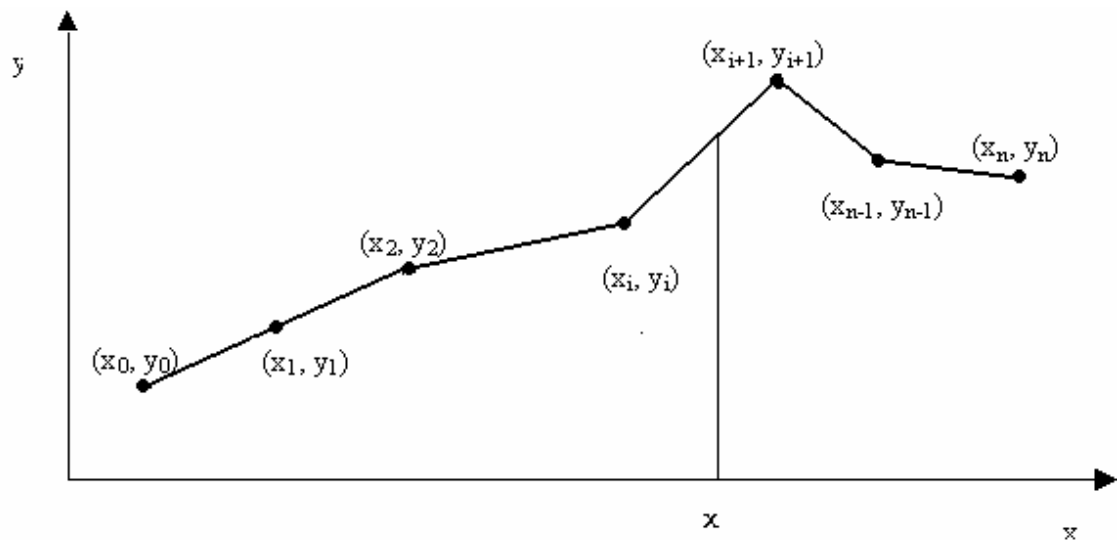
Jei $f(x_i)$ ir $f(x_{i+1})$ yra dvi minėtos funkcijos reikšmės, tai paėmę $n=1$, gauname:

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

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

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

Interpoliavimas pirmojo laipsnio polinomais vadinamas tiesiniu interpoliavimu. Tiesinio interpoliavimo formulės (polinomo) grafikas yra laužtė (1.8 pav), jungianti taškus $(x_i, f(x_i))$, ($i = 0, 1, \dots, n$).



1.8 pav. Tiesinio interpoliavimo polinomo grafiko pavyzdys

1.7.2. Suglodinimas

Nagrinėdami interpoliavimo uždavinį, funkciją $f(x)$, apibrėžtą reikšmių lentele stengėmės pakeisti tokia aproksimuojančiaja funkcija $F(x)$, kad taškuose x_i ($i = 0, 1, \dots, m$) $F(x_i)$ reikšmės būtų lygios $f(x_i)$ reikšmėms.

Tada, kai $f(x_i)$ reikšmės yra eksperimento rezultatai ir turi matavimo bei metodo paklaidų. Tokiais atvejais reikalauti, kad aproksimuojančioji funkcija $F(x)$ tenkintų sąlygą $F(x_i) = f(x_i)$, būtų neprotinga. Geriau rasti tokią funkciją $F(x)$, kuri pagal pasirinktą kriterijų geriausiai aproksimuotų $f(x)$. Toks funkcijos apskaičiavimo metodas vadinamas suglodinimu. Atsižvelgiant į suglodinimo kriterijų galima gauti įvairias $F(x)$ išraiškas.

Sakykime, funkcija $y = f(x)$ nusakyta reikšmių lentele (x_i, \tilde{y}_i) , čia $i = 1, 2, \dots, m$. Simbolis \tilde{y}_i rodo, kad $f(x)$ reikšmės taškuose x_i yra apytikslės. Taip pat žinoma aproksimuojančios funkcijos $F(x)$ analizinė išraiška (dažnai vadinama empirine funkcija): $F(x, a_0, \dots, a_n)$, čia a_i – nežinomi parametrai ($i = 1, 2, \dots, m$) 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. Paprastai laikoma, kad aproksimavimo santykinė procentinė paklaida neturėtų viršyti 5 % (18, 20). Šiuo metu populiariausi šie suglodinimo metodai:

- a) pasirinktų taškų metodas,
- b) vidurkių metodas,
- c) mažiausių kvadratų metodas.

Mažiausių kvadratų metodas. Sakykime, kad argumento x_i reikšmę atitinka eksperimentu gauta y_i reikšmė. Įrašę x_i reikšmę į empirinės funkcijos išraišką, gausime funkcijos reikšmę $f(x_i, a, b, c, \dots)$, paprastai nelygią y_i , nes y_i reikšmėms turi įtaką eksperimento paklaidos. Skirtumą $y_i - f(x_i, a, b, c, \dots)$ vadiname empirinės funkcijos nuokrypiu. Mažiausių kvadratų metodo reikalavimas yra toks: parametrai a, b, c, \dots turi būti tokie, kad nuokrypių kvadratų suma būtų minimali. Taigi

$$S = \sum_{i=1}^n (y_i - f(x_i, a, b, c, \dots))^2 \rightarrow \min.$$

Tokiu atveju funkcijos S dalinės išvestinės kintamųjų a, b, c, \dots atžvilgiu turi būti lygios nuliui:

$$\frac{\partial S}{\partial a} = 0, \quad \frac{\partial S}{\partial b} = 0, \quad \frac{\partial S}{\partial c} = 0, \dots$$

Išdiferencijavę S parametru a, b, c, \dots atžvilgiu ir prilyginę gautas dalines išvestines nuliui, gauname lygčių sistemą, sudarytą iš tiek lygčių, kiek parametru turi funkcija $y = f(x, a, b, c, \dots)$:

$$\begin{cases} \sum_{i=1}^n (y_i - f(x_i, a, b, c, \dots)) \frac{\partial f}{\partial a} = 0, \\ \sum_{i=1}^n (y_i - f(x_i, a, b, c, \dots)) \frac{\partial f}{\partial b} = 0, \\ \dots \end{cases} \quad (1.19)$$

Panagrinėsime dažniausią atvejį, kai funkcija $y = f(x, a, b, c, \dots)$ yra tiesinė, t.y.

$$y = ax + b.$$

Kadangi $\frac{\partial f}{\partial a} = x$, $\frac{\partial f}{\partial b} = 1$, tai iš (1.19) sistemos gauname sistemą

$$\begin{cases} \sum_{i=1}^n (y_i - (ax_i + b))x_i = 0, \\ \sum_{i=1}^n (y_i - (ax_i + b)) = 0, \end{cases}$$

kuri pertvarkoma į tokią:

$$\begin{cases} a \sum_{i=1}^n x_i^2 + b \sum_{i=1}^n x_i = \sum_{i=1}^n x_i y_i, \\ a \sum_{i=1}^n x_i + nb = \sum_{i=1}^n y_i. \end{cases}$$

Pasinaudoję Kramerio formulėmis, gauname:

$$a = \frac{\sum_{i=1}^n x_i \sum_{i=1}^n y_i - n \sum_{i=1}^n x_i y_i}{\left(\sum_{i=1}^n x_i\right)^2 - n \sum_{i=1}^n x_i^2},$$

$$b = \frac{\sum_{i=1}^n x_i y_i \sum_{i=1}^n x_i - \sum_{i=1}^n y_i \sum_{i=1}^n x_i^2}{\left(\sum_{i=1}^n x_i\right)^2 - n \sum_{i=1}^n x_i^2}.$$

Kai empirinė funkcija yra hiperbolinė, t.y.

$$y = \frac{k}{x} + l.$$

$\frac{\partial f}{\partial k} = \frac{1}{x}, \frac{\partial f}{\partial l} = 1$, tai iš (1.19) sistemos gauname sistemą

$$\begin{cases} \sum_{i=1}^n (y_i - (\frac{k}{x_i} + l)) \frac{1}{x_i} = 0, \\ \sum_{i=1}^n (y_i - (\frac{k}{x_i} + l)) = 0, \end{cases}$$

kurią pertvarkome į tokią:

$$\begin{cases} k \sum_{i=1}^n \frac{1}{x_i^2} + l \sum_{i=1}^n \frac{1}{x_i} = \sum_{i=1}^n \frac{y_i}{x_i}, \\ k \sum_{i=1}^n \frac{1}{x_i} + nl = \sum_{i=1}^n y_i. \end{cases}$$

Šiame darbe naudosime mažiausių kvadratų metodą gauti funkcines priklausomybes tarp tam tikrų integralios organizmo reakcijos į fizinį krūvį parametrų. Empirinę funkciją parinksime remdamiesi medikų patirtimi, kai kuriais atvejais taikydami vadinamąjį ištiesinimo metodą, bei reikalaudami, kad vidutinė santykinė procentinė aproksimacijos paklaida neviršytų 5 % (3,4,6).

Siekdami sumažinti apvalinimo paklaidas, šiame darbe sudarytoms tiesinių lygčių sistemoms spręsti naudosime Kramerio metodą – sprendinį gausime atlikę baigtinių skaičių aritmetinių veiksmų.

1.8. Skirtuminiai santykiai

Tiriant tam tikrus reiškinius praktikoje dažnai patogu ne tik stebėti atitinkamo dydžio kitimus, bet ir tirti kitimo greitį. Tada, kai nėra žinomos parametrų priklausomybių analizinės išraiškos, kitimo greičiams ir pagreičiams nagrinėti paprastai naudojami baigtiniai skirtumai ir skirtuminiai santykiai.

Sakykime, turime funkcijos $f(x)$ reikšmes $f(x_0), f(x_1), \dots, f(x_n)$.

Apibrėžimas. Funkcijos pokyčio tarp dviejų gretimų duotųjų reikšmių ir atitinkamo argumento pokyčio santykį, t. y. dydį

$$f(x_i, x_{i+1}) = \frac{f(x_{i+1}) - f(x_i)}{x_{i+1} - x_i} \quad (1.20)$$

vadiname pirmos eilės skirtuminiu santykiu.

Turėdami $n+1$ funkcijos reikšmę, randame n pirmos eilės skirtuminių santykių:

$$f(x_0, x_1) = \frac{f(x_1) - f(x_0)}{x_1 - x_0}, \quad f(x_1, x_2) = \frac{f(x_2) - f(x_1)}{x_2 - x_1}, \quad \dots, \quad f(x_{n-1}, x_n) = \frac{f(x_n) - f(x_{n-1})}{x_n - x_{n-1}}.$$

Pagal juos apskaičiuojame antros eilės skirtuminius santykius:

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

Apibrėžimas. K – tos eilės skirtuminiu santykiu vadiname $k - 1$ eilės skirtuminių santykių pokyčio santykį su argumento pokyčiu:

$$f(x_i, x_{i+1}, \dots, x_{i+k}) = \frac{f(x_{i+1}, x_{i+2}, \dots, x_{i+k}) - f(x_i, x_{i+1}, \dots, x_{i+k-1})}{x_{i+k} - x_i}, \quad (k=1, 2, \dots) \quad (1.21)$$

Skirtuminius santykius patogiau surašyti į lentelę (1.1 lentelė)

1.1 lentelė

Skirtuminiai santykiai

x_i	$f(x_i)$	$f(x_i, x_{i+1})$	$f(x_i, x_{i+1}, x_{i+2})$...
x_0	$f(x_0)$			
x_1	$f(x_1)$	$f(x_0, x_1)$		
x_2	$f(x_2)$	$f(x_1, x_2)$	$f(x_0, x_1, x_2)$	
x_3	$f(x_3)$	$f(x_2, x_3)$	$f(x_1, x_2, x_3)$...
...

Užrašysime keletą skirtuminių santykių savybių.

Teorema. K – tos eilės skirtuminių santykių galima išreikšti formule:

$$f(x_i, x_{i+1}, \dots, x_{i+k}) = \sum_{j=0}^k \frac{f(x_{i+j})}{(x_{i+j} - x_i) \dots (x_{i+j} - x_{i+j-1})(x_{i+j} - x_{i+j+1}) \dots (x_{i+j} - x_{i+k})} \quad (1.22)$$

Iš šios teoremos išplaukia išvados:

1. funkcijų sumos arba skirtumo skirtuminiai santykiai yra lygūs tų funkcijų skirtuminių santykių sumai arba skirtumui;
2. pastovų daugiklį galima iškelti prieš skirtumo ženklą;
3. k – tos eilės skirtuminis santykis yra savo argumentų simetrinė funkcija.

Praktikoje dažnai pasitaiko, kad tam tikro dydžio argumentas kinta vienodu žingsneliu. Taigi, jei atstumai tarp interpoliavimo taškų vienodi, tai vietoj skirtuminių santykių patogiau 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

$$\Delta^k f(x_i) = \Delta^{k-1} f(x_{i+1}) - \Delta^{k-1} f(x_i), \quad (k=1, 2, 3, \dots) \quad (1.23)$$

Baigtiniai skirtumai surašomi į 1.2 lentelę:

1.2 lentelė

Baigtiniai skirtumai

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_3	$f(x_3)$	$\Delta f(x_2)$	$\Delta^2 f(x_1)$	$\Delta^3 f(x_0)$	
...

Pateiksime baigtinių skirtumų savybes.

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

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

2 lema. *Baigtiniai skirtumai išreiškiami formule*

$$\Delta^k f(x_i) = \sum_{m=0}^l (-1)^m C_l^m \Delta^{k-1} f(x_{i+l-m}), l \leq k. \quad (1.24)$$

3 lema. *Baigtiniai skirtumai ir skirtuminiai santykiai susieti priklausomybe:*

$$\frac{\Delta^k f(x_i)}{k! h^k} = f(x_i, x_{i+1}, \dots, x_{i+k}). \quad (1.25)$$

1.9. Modeliavimas medicinoje – trumpa apžvalga

Europos Sąjungos šalyse realizuojant įvairius e – sveikatos projektus (21) lieka neišspręsta problema, kaip formalizuoti žmogaus, kaip dinaminės sistemos, modelį. Todėl daugelyje e – sveikatos strategijos projektų lieka neapimta virtualaus gydytojo pozicija, stengiantis integruoti visą gaunamą informaciją jau sveikatos apsaugos vieneto rėmuose (poliklinika, ligoninė). Žinoma, sukurti imitacinį žmogaus modelį labai sudėtinga, tačiau modeliuoti atskiras posistemes įdomu ir prasminga.

Pastaruoju laiku ypač susirūpinta žmogaus sveikatos stiprinimu, palaikymu. Dažnai problemos sprendimui pasitelkiamas įvairaus pobūdžio fizinis aktyvumas. Visą pastarąjį šimtmetį mokslininkai tyrinėja ir aktyviai diskutuoja nagrinėdami įvairius organizmo pokyčius, bei jų mechanizmus, kuriuos sukelia fizinis aktyvumas. Dažniausiai nesutariama dėl žmogaus organizmo sistemų vaidmens svarbos fizino krūvio metu. Pastaraisiais metais tobulėjant kompiuterinei technikai ir populiarėjant imitaciniam modeliavimui daugėja mokslinių darbų, susietų su žmogaus fiziologinių sistemų modeliavimu. Daug darbų šia tema paskelbęs italų mokslininkas M. Ursino (22). Pastaruoju metu jo vadovaujama autorių grupė stengiasi tobulinti (pridedami vis nauji parametrai) imitacinį matematinį modelį, apimantį kardiovaskulinės sistemos atsaką į aerobinį krūvį. M. Ursino pirmąkart apie modelį paskelbė dar 1998 m. Šiuo metu modelis pateikiamas kaip elektros garandinės analogas, apimantis širdies dažnį, arterinį kraujospūdį, plaučių ventiliaciją, kraujagyslių kraujotaką, raumenų darbą ir reguliacinius mechanizmus

aerobinių pratimų atlikimo metu. Šių sistemų darbą apibūdina daugelis parametru, kurie pagal tam tikrą literatūrą (kaip nurodo autoriai), parenkami kaip konstantos. Įvertinant tą aplinkybę, kad kiekvienas žmogus yra individuali sistema ir jo organizmo reakcija į fizinį krūvį labai sudėtinga, galima teigti, kad modelis individualiam asmeniui nėra tinkamas. Modelyje iš anksto postuluojami ryšiai tarp tam tikrų sistemų parametru, parametru kitimo dėsniai, kurie nėra pagrįsti tiesiogine klinicine statistika. Šių ryšių teisingumo mūsų sąlygomis praktiškai patikrinti neįmanoma, nes kai kuriems matavimams atlikti tiesiog nėra sąlygu. Vadinasi paklaidos neišvengiamos. Literatūroje aprašyti ir kiti panašūs imitaciniai modeliai (23,24,25).

Literatūroje randame daug mokslinių darbų, kuriuose apsiribojama vienos konkrečios žmogaus organizmo sistemos nagrinėjimu (26), tačiau mus domina ir šių sistemų tarpusavio sąveika.

Imitacinių modelių kūrimui naudojama įvairi programinė įranga: pradedant specializuota imitaciniam modeliavimui (Simulink), baigiant profesionaliomis programavimo kalbomis (25).

Mūsų darbe nagrinėjamas integralios organizmo reakcijos į fizinį krūvį modelis, remiantis formalizuoto poveikio strategija. Čia ir iškyla pagrindinė problema – kaip šį modelį apibūdinti tokiomis matematinėmis informacinėmis sąvokomis, kad paskui būtų galima sudaryti taikomąją programą, skirtą skaičiuoti norimas modelio charakteristikas. Šio modelio aprašymui, mūsų manymu, tinka Milio ir Muro automatų sąsūka, užsiduodant, kad vieno automato išėjimo signalai tampa kito automato įėjimo signalais (27,28). Modelio nagrinėjimui pakanka sudaryti atitinkamas siurjekcijas, kuriomis remiantis kuriama taikomoji programa. Trumpai apžvelgsime Milio ir Muro automatus klasikinėje automatų teorijoje.

1.10. Milio ir Muro automatai

Milio automato pavadinimas kilo nuo jo jo steigėjo G. H. Mealy pavardės, vieno iš būsenų automatų pradininko, kuris darbus šia tema paskelbė dar 1956 m. Po G. H. Mealy darbų paskelbimo atsirado ir Muro automato sąvoka. Šis automato pavadinimas kilo nuo jo tyrėjo E. F. Moore pavardės, taip pat vieno iš būsenų automatų pradininko, darbus šia tema paskelbusio 1965 m.

Šiuo metu gausu literatūros, kurioje kalbama apie Milio ir Muro automatus. Bene plačiausiai šiuos automatus, jų rūšis bei veikimą nagrinėja Vilfredas Braueris knygoje „Įvadas į baigtinių automatų teoriją“. Šiandieniniame moksle Milio ir Muro automatai yra gana populiarūs, tačiau išsamių darbų apie šių automatų apjungimą beveik nėra. A.A. Shalyto (29) baigtinius automatus naudojo orientuotame programavime, programiniuose protokoluose. K. Svozil (30) bandė apskaičiuoti keletą racionaliai įsivaizduojamų papildomumų formų ar loginių struktūrų, jungdams Milio ir Muro automatus nuosekliai ir lygiagrečiai.

Plačiausia Milio ir Muro automatai praktikoje yra taikomomi aptarnavimo sistemų, kompiuterinių tinklų formaliojo specifikavimo, kompiuterinių protokolų sudarymo, teisingumo tikrinimo ir imitacinio modeliavimo uždaviniuose (31,32).

Klasikinėje automatų teorijoje Milio ir Muro automatai turi baigtinį skirtingų būsenų bei signalų skaičių, tačiau juos naudojant konkretaus modelio aprašymui, būsenų ir įėjimo signalų skaičių tikslinga papildyti iki begalinio skaičiaus. Automatuose vykstantiems procesams tirti paprastai įvedama laiko sąvoka, tačiau mūsų atveju įvedamas tik įvykių eiliškumo skaitiklis. Be to, šių automatų sąsūka reiškia, kad vieno automato išėjimo signalai tampa kito automato įėjimo signalais.

1.10.1. Milio automatas

Apibrėžimas. *Baigtinis Milio automatas – tai sistema $ML = (Z, X, Y, f, g)$, kur Z, X ir Y baigtinės aibės, o f ir g – vaizdavimai:*

Z – baigtinė automato būsenų aibė;

X – baigtinė automato įėjimų signalų aibė;

Y – baigtinė automato išėjimų signalų aibė;

$f: Z \times X \rightarrow Z$ – automato perėjimų funkcija;

$g: Z \times X \rightarrow Y$ – automato išėjimų funkcija (čia vaizdavimas g – siurjekcija).

Jei automata ML išskirta viena būsena z_0 , vadinama pradine, toks automatas gali būti vadinamas inicialiniu ir žymimas (ML, z_0) .

1 pavyzdys. Poveikis sistemai gali būti teigiamas arba neigiamas. Sistema nereaguoja į neigiamąjį poveikį ir reaguoja tik į kas antrą teigiamąjį poveikį.

Tegu įėjimų aibė $X = \{x_1, x_2\}$; čia x_1 – neigiamasis poveikis; x_2 – teigiamasis poveikis.

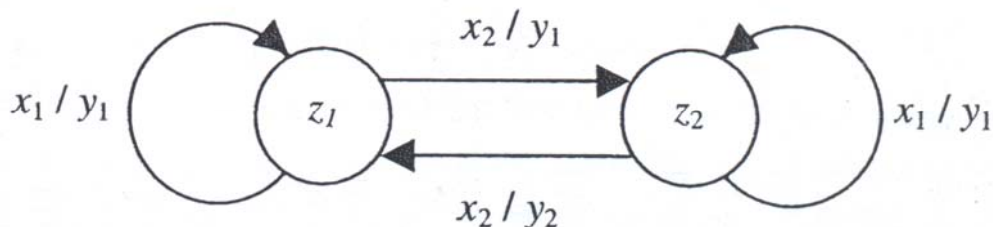
Išėjimų aibė $Y = \{y_1, y_2\}$; čia y_1 – nėra reakcijos; y_2 – yra reakcija.

Būsenų aibė $Z = \{z_1, z_2\}$; čia z_1 – būsena po antrojo (lyginio) teigiamojo poveikio arba sistemos funkcionavimo pradžia; z_2 – būsena po pirmojo (nelyginio) teigiamojo poveikio.

Jei sistema yra būsenoje z_1 (būsena po antrojo (lyginio) teigiamojo poveikio arba sistemos funkcionavimo pradžia) ir ateina signalas x_2 (teigiamasis poveikis), tai išėjime pasirodys signalas y_1 – nėra reakcijos, o sistema pereis į būseną z_2 (būsena po pirmojo teigiamojo poveikio). Jei automatas yra būsenoje z_2 ir įėjime yra signalas x_2 , tai išėjime pasirodys signalas y_2 , o sistema pereis į būseną z_1 (reakcija į paskutinį teigiamąjį poveikį). Jei įėjime yra signalas x_1 , tai išėjime nepriklausomai nuo būsenos gaunamas signalas y_1 , o sistemos būsena lieka nepakitusi.

Paprastai Milio automatas aprašomas orientuotuoju grafu (būsenų diagrama) arba būsenų matricomis (automato lentelėmis). Išnagrinėtos sistemos funkcionavimą pavaizduosime orientuotuoju

grafu (1.9 pav.) . Šiame grafe viršūnės atitinka sistemos būsenas, rodyklės – įėjimo signalo poveikį. Virš rodyklių žymime įėjimo ir išėjimo signalus.



1.9 pav. Sistemos funkcionavimo diagrama

Automato funkcionavimas pavaizduotas orientuotoju grafu dažnai dar vadinamas perėjimų grafu. Kadangi funkcijų f ir g argumentų bei reikšmių aibės baigtinės, jos gali būti aprašomos ir lentelėmis. Paprastai abi lentelės sujungiamos į vieną $f \times g: Z \times X \rightarrow Z \times Y$, vadinamą perėjimų arba tiesiog automato lentelę. Toliau pateikiama pavyzdyje nagrinėto automato perėjimų lentelė (žr. 1.3 lentelę).

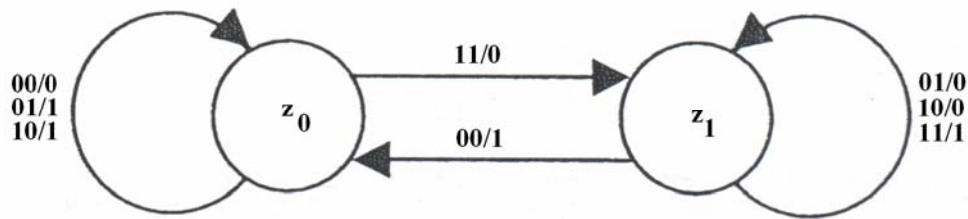
1.3 lentelė

Automato perėjimų lentelė

	x_1	x_2
z_1	z_1/y_1	z_2/y_1
z_2	z_2/y_1	z_1/y_2

Nagrinėjant baigtinį automatą, būtina formaliai aprašyti jo elgseną – nustatyti, kaip automatas reaguoja į baigtines įėjimo signalų sekas. Jei šių sekų aibė begalinė, baigtinėmis priemonėmis (lentelėmis, grafais) aprašyti automato reakcijas į sekas neįmanoma. Tokiais atvejais apibrėžiami perėjimo ir išėjimo atvaizdavimai, kai į įėjimą siunčiamos signalų sekos. Šie atvaizdavimai paprastai apibrėžiami iteraciniu būdu, remiantis atvaizdavimais f ir g .

2 pavyzdys. Iš pradžių aprašykime Milio automatą, kuris atlieka sumavimo operaciją su sveikaisiais neneigiamais skaičiais dvejetainėje skaičiavimo sistemoje. Veiksmai atliekami nuosekliai. Šio automato būsenų grafas pateiktas 1.10 paveiksle.



1.10 pav. Milio automato, aprašančio sumavimo operaciją, būsenų grafas

Būsena z_1 reiškia vaizdavimą į kitą klasę. Automatas pradeda dirbti būdamas būsenoje z_0 .

1.10.2. Muro automatas

Apibrėžimas. *Baigtinis Muro automatas* – tai sistema $MR = (Z, X, Y, f, h)$, kur Z , X ir Y netuščios baigtinės aibės, o f ir g – vaizdavimai:

Z – baigtinė automato būsenų aibė;

X – baigtinė automato įėjimų signalų aibė;

Y – baigtinė automato išėjimų signalų aibė;

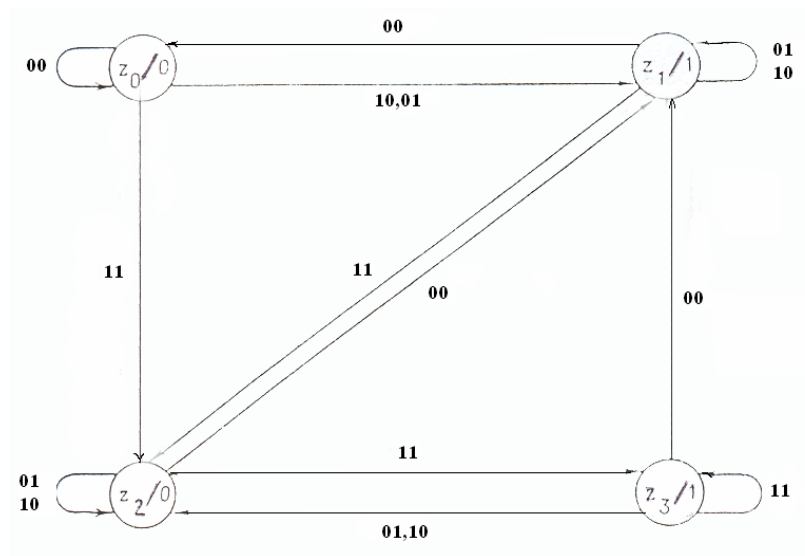
$f: Z \times X \rightarrow Z$ – automato perėjimų funkcija;

$h: Z \rightarrow Y$ – automato išėjimų funkcija (čia vaizdavimas h – siurjeksija).

Jei automate MR išskirta viena būsena z_0 , vadinama pradine, toks automatas gali būti vadinamas inicialiniu ir žymimas (MR, z_0) .

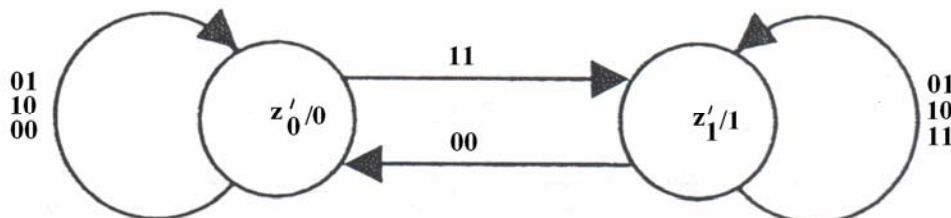
Klasikinio Muro automato išėjimai priklauso tik nuo esamos būsenos, kai tuor tarpu Milio automato išėjimai priklauso ir nuo esamos būsenos ir nuo įėjimo signalo. Matematiškai Milio ir Muro automatai yra ekvivalentūs – jie gali būti pertvarkyti iš vieno į kitą. Tačiau paprastai būsenų skaičius Muro automate bus didesnis arba bus lygus lyginant su būsenų skaičiumi Milio automate.

1 pavyzdys. Sukurkime Muro automatą, kurio darbo principas būtų panašus kaip 1.10.1 skyrelio 2 pavyzdyje. Automatas turi keturias būsenas: po dvi kiekvienam išėjimui 0 ir 1, tam, kad viena iš tokių būsenų „atsimintų“ 1 perėjimą į kitą klasę, o kita – ne (1.11 pav.).



1.11 pav. Muro automato būsenų grafas

2 pavyzdys. Sukurkime Muro automatą, kuris įvedus 2 sveikuosius neineigiamus dvejetainius skaičius nuspręstų, ar šių skaičių suma mažesnė už 2^n (n – maksimalus įvedamų skaičių ilgis). Rezultatas įvedus skaičius turi būti 0, jeigu sąlyga išpildyta. Orientuotas šio automato būsenų grafas pateiktas 1.12 pav.

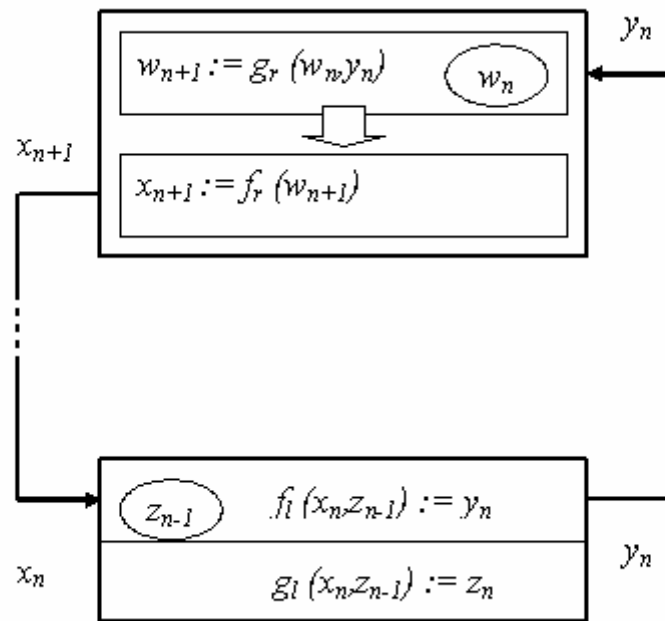


1.12 pav. Muro automato funkcionavimo diagrama

Šis automatas gali būti gaunamas iš 1 pavyzdyje pateikto Muro automato, jei apjungtume būsenas z_0 ir z_1 į vieną naują būseną z'_0 , o būsenas z_2 ir z_3 į vieną naują būseną z'_1 ir atitinkamai pekeistume perėjimų ir išėjimų funkcijas.

1.10.3. Milio ir Muro automatų sąsūkos pavyzdys

Milio ir Muro automatų sujungimą, kai vieno automato išėjimo signalai tampa kito automato įėjimo signalais vadinsime jų sąsūka (27, 28). Panagrinėkime pavyzdį, sudarydami Milio ir Muro automatų sąsūką, pavaizduotą 1.13 paveiksle.



1.13 pav. Milio ir Muro automatų sąšukos pavyzdys

Šios sąšukos darbą nusakančios siurjekcijos:

$$g_r : W \times Y \rightarrow W,$$

$$f_r : W \rightarrow X,$$

$$f_l : X \times Z \rightarrow Y,$$

$$g_l : X \times Z \rightarrow Z.$$

čia W – Muro automato būsenų aibė, X – Muro automato išėjimo – Milio įėjimo signalų aibė, Y – Milio išėjimo – Muro įėjimo signalų aibė, Z – Milio automato būsenų aibė. Be to, $x_n \in X$, $y_n \in Y$, $z_n \in Z$, $w_n \in W$.

Tada automatų sąšukos darbo procesą galima pavaizduoti taip:

$$\begin{bmatrix} - \\ - \\ - \\ z_0 \end{bmatrix} \Rightarrow \begin{bmatrix} w_1 \\ f_r(w_1) = x_1 \\ f_l(x_1, z_0) = y_1 \\ g_l(x_1, z_0) = z_1 \end{bmatrix} \Rightarrow \begin{bmatrix} g_r(w_1, y_1) = w_2 \\ f_r(w_2) = x_2 \\ f_l(x_2, z_1) = y_2 \\ g_l(x_2, z_1) = z_2 \end{bmatrix} \dots$$

1.14 pav. Sąšukos darbo realizacija

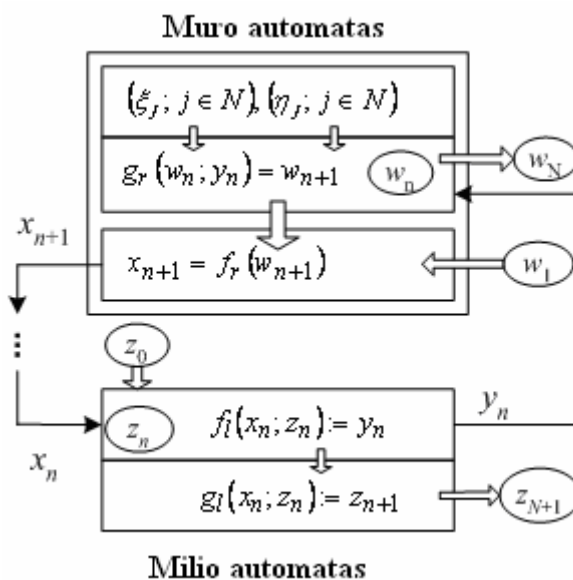
Automatų sąsūka pradeda veikti nustačius pradines būsenas w_1 ir z_0 ir gali dirbti neapibrėžtą laiką. Pastebėsime, kad paleidžiant dirbti konkrečias Muro ir Milio automatų sąsūkas, reikalinga tarpusavyje suderinti Milio ir Muro automatų pradines būsenas w_1 ir z_0 . Priešingu atveju gali būti gauti nekorektiški rezultatai.

Literatūroje (28,29) Milio ir Muro automatų sąsūka buvo panaudota aptarnavimo sistemos, susidedančios iš vieno paraiškų srauto ir vieno aptarnavimo kanalo aprašymui (1. 15 pav.).



1.15 pav. Aptarnavimo sistema

Į sistemą atsitiktiniais (arba determinuotais) laiko momentais $\xi_1, \xi_1 + \xi_2, \xi_1 + \xi_2 + \xi_3, \dots$ ateina paraiškos. Ateidama į sistemą, paraiška atsineša aptarnavimo trukmės laikus $\eta_1, \dots, \eta_n, \dots$. Aptarnavimo sistemoje įvykstančius įvykius žymėsime: A_k – atėjo k-toji paraiška; B_k – pradėjo aptarnauti k-ąją paraišką; C_k – baigta aptarnauti k-toji paraiška. Laiko momentas t_n nurodo išėjimo signalo iš Muro automato laiko momentą. z_n nusako paraiškų skaičių, esantį sistemoje įskaitant ir aptarnaujamąją paraišką. Aptarnavimo sistemą aprašanti automatų sąsūką gali būti pavaizduota taip:



1.16 pav. Aptarnavimo sistemos formalusis aprašas

1.11. Programinė įranga

Integruotai duomenų analizei ir integralios organizmo reakcijos į fizinių krūvių modelio aprašymui matematinėmis informacinėmis sąvokomis reikalinga universali programavimo kalba, suteikianti galimybę duomenis analizuoti grafiniu ir skaitmeniniu būdu. Įprasti statistiniai duomenų analizės metodai šiuo atveju praktiškai netinka, kadangi kiekvieno žmogaus organizmas yra individualus, o individualių matavimų statistikai sukaupti paprastai būna gerokai per mažai. Funkcinių ryšių tarp integralinio vertinimo modelio parametrų iliustarvimui būtina grafinė duomenų analizė. Tais atvejais, kai medicininė praktika negali tiksliai pasakyti, kokie yra skirtingų parametrų ryšiai, geriausiai tinka palyginamoji grafinė ir skaitmeninė analizė. Ypač svarbu individualaus paciento duomenų saugojimas ir pateikimas, galimybė juos palyginti su senesniais to paties asmens tyrimais ar tam tikros grupės duomenimis.

Šiems ir kitiems uždavinams spręsti ypač svarbi paprasta ir patogi vartotojo sąsaja, suderinama su operacine sistema Windows.

Šiuos reikalavimus puikiai atitinka C++ Builder – objektiškai orientuotų programų kūrimo sistema, kurios priemonėmis galima paruošti įvairaus lygio ir paskirties programas: tiek mėgėjiškas, tiek profesionalias (33). Programos C++ Builder terpėje kuriamos, įtraukiant į programą reikalingus objektus (komponentus), suteikiant jiems reikalingas savybes ir programuojant C++ kalba reakcijas į įvykius, kurie gali įvykti programos vykdymo metu. Visų tų dalių sujungimą į vientisą C++ kalbos sintaksę atitinkančią programą automatiškai atlieka C++ Builder sistema.

2. TIRIAMOJI DALIS

2.1. Informacinės posistemės apibūdinimas

Dar Hipokratas 460 – 400 metais prieš Kristaus gimimą sakė: „Visos atliekančios kurių nors funkciją kūno dalys gerai išlavėja, išlieka sveikos ir lėčiau sensta, jei jos naudojamos saikingai ir atlieka užduotis, prie kurių yra įpratusios. Tačiau jei šios kūno dalys nieko neveikia ir paliekamos be darbo, jos neišlavėja, yra linkusios į ligas ir greitai pasensta”. Pastarąjį dešimtmetį atlikti moksliniai tyrimai bei įvairios studijos parodė, kad nepakankamas fizinis aktyvumas tiek sąlyginai sveikiems žmonėms, tiek sergantiems širdies ir kraujagyslių sistemos ligomis gali sukelti nepageidaujamų pasekmių. Ir atvirkščiai, tinkamai parinktas ir adekvačiais dozuojamas fizinis krūvis gali pagerinti gyvenimo kokybę, t.y. fizinio krūvio toleravimą, sumažinti ligos sukeltų simptomų intensyvumą ar neleisti jiems atsirasti. Adekvačiam fizinio krūvio parinkimui patenkinti būtina tinkamai kaupti ir nagrinėti konkretaus paciento duomenis, stebėti įvairių parametru, bei įvairių išvestinių dydžių dinamiką tolimesnių tyrimų metu, neapsiriboti vienos fiziologinės sistemos tyrimu, suprantant, kad žmogus – kompleksinė adaptivi sistema.

Europos Sąjungos šalyse realizuojant įvairius e – sveikatos projektus (21) lieka neišspręsta problema, kaip formalizuoti žmogaus, kaip dinaminės sistemos, modelį. Todėl daugelyje e – sveikatos strategijos projektų lieka neapimta virtualaus gydytojo pozicija, stengiantis integruoti visą gaunamą informaciją jau sveikatos apsaugos vieneto rėmuose (poliklinika, ligoninė).

Vadovaudamiesi e – sveikatos komponentais (2), integralios organizmo reakcijos į fizinį krūvį ir formalizuotu fizinio krūvio poveikio modeliais (žr. 1.3 ir 1.4 skyrelius), sukūrėme bei formalizavome Milio ir Muro automatų sąsūką (aprašėme matematinėmis informacinėmis sąvokomis), kaip individualiam asmeniui orientuotą informacinę posistemę.

2.2. Duomenys

Integralios organizmo reakcijos į fizinį krūvį modelio nagrinėjimui naudoti veloergometriniai mėginiai. Širdies funkciniais rodikliais vertinti naudota Kauno medicinos universitete Kardiologijos institute sukurta kompiuterizuota 12-os standartinių derivacijų elektrokardiogramų (EKG) analizės sistema „Kaunas – krūvis”. Buvo taikyta veloergometrinio pakopomis didėjančio provokacinio fizinio krūvio metodika. Testai buvo atlikti naudojantis formalizuotu poveikio modeliu. Veloergometrija

atlikta, taikant pradinę 50 W krūvį (N), jį didinant kas minutę po 50 W vyrams ir profesionalioms sportininkėms, kitiems – po 25 W. Krūvis tęsiamas iki submaksimalaus širdies susitraukimo dažnio ($\check{S}SD$) arba kol atsiranda ribojantys pagal AHA (Amerikos širdies asociacija) rekomendacijas požymiai. Buvo vertinami ir analizuojami šie parametrai:

- širdies susitraukimų dažnis $\check{S}SD$,
- sistolinis arterinio kraujo spaudimas S ,
- diastolinis arterinio kraujo spaudimas D ,
- **JT** intervalas 12 – oje derivacijų,
- – ST amplitudė,
- QRS intervalas.

Visi šie parametrai buvo registruojami po vieną kartą ramybėje, esant skirtingam galingumui (galingumas didinamas kas minutę) ir atsigavimo metu (atsigavimo metu parametrai taip pat registruoti kas minutę – maksimali trukmė – 5 min.). Asmeniui buvo galima registruoti daugiausiai 7 vieno parametro matavimus fizinio krūvio metu ir daugiausiai 5 atsigavimo metu.

Duomenų pavyzdys, kai galingumas kas minutę keliamas po 50 W, pateiktas 2.1 lentelėje.

2.1 lentelė

Paciento duomenų pavyzdys

	0 W	50 W	100 W	150 W	...	300 W	1 min	2 min	...	5 min
$\check{S}SD$	60	80	100		...		80	70	...	60
S	120	130	140		...		130	130	...	120
D	80	80	70		...		60	70	...	70
JT	3.26	3.18	3.05		...		3.06	3.2	...	3.25
– ST	–0.05	–0.06	–0.11		...		–0.08	–0.06	...	–0.06
QRS	0.07	0.07	0.08		...		0.08	0.08	...	0.08

Pastebėsime, kad vieno paciento duomenys rašomi vienoje duomenų bylos eilutėje tokia tvarka, kaip parodyta 2.1 lentelėje. Prieš šiuos duomenis yra bendroji informacija apie pacientą: pavardė, vardas, tyrimo numeris, tyrimo data, lytis, amžius, svoris, ūgis, sveikatingumo grupės požymis, galingumo kėlimo laiptelio dydis (25 ar 50 W), maksimalus išvystytas galingumas.

Modelio tyrimui buvo pasirinkti skirtingi tiriamieji (iš viso 612 tyrimų): profesionalūs sportininkai (jie buvo grupuojami pagal lytį), sveikata besirūpinančios moterys – sportininkės mėgėjos, nesportuojantys, bet save sveikais laikantys asmenys (tiriamieji buvo suskirstyti į grupes pagal lytį ir amžių) bei išemine širdies liga sergantys vyrai. Tiro kontingento duomenys pateikti 2.2 lentelėje.

2.2 lentelė

Tirto kontingento duomenys

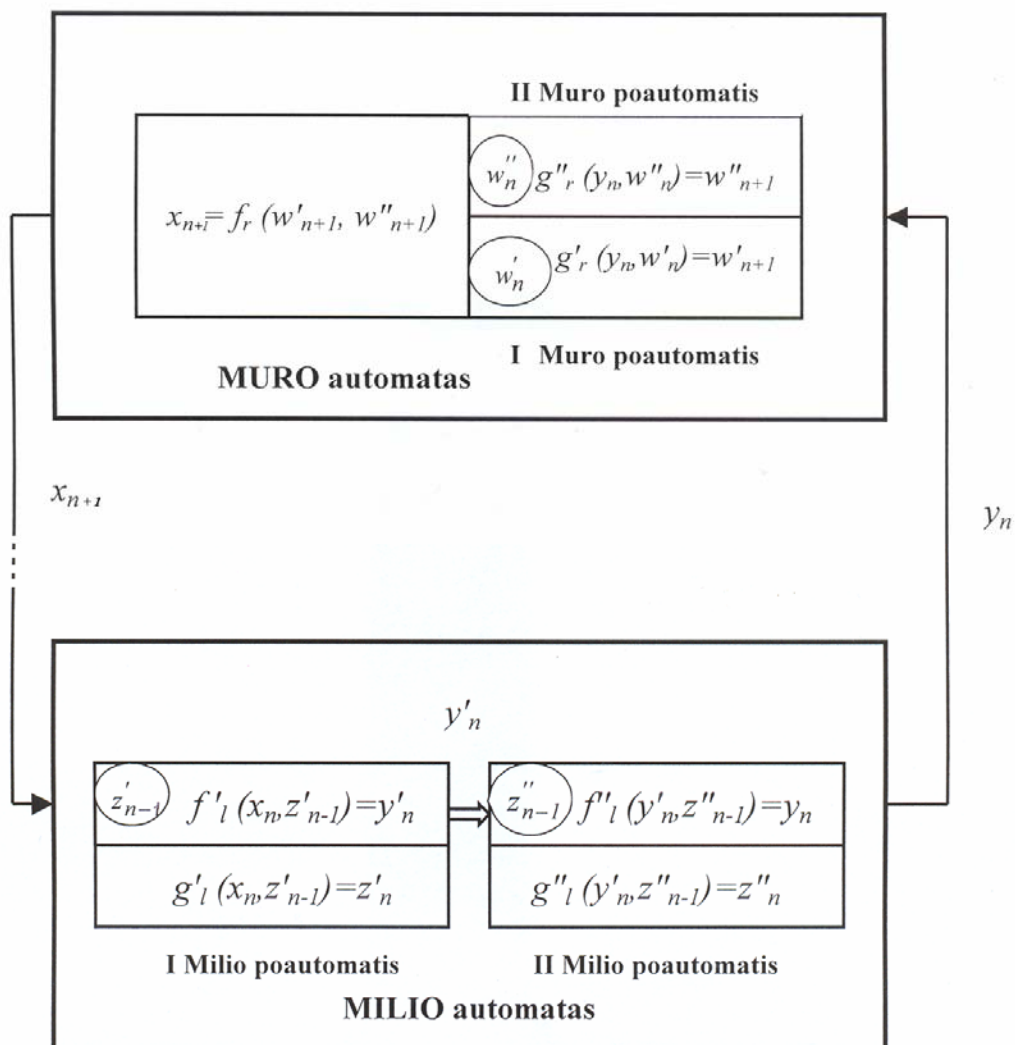
Grupė	Amžiaus vidurkis ± SEM	Tyrimų skaičius
Sportininkės mėgėjos	33,35 ± 0,97	20
Profesionalūs sportininkai	23,56 ± 0,39	161
Profesionalės sportininkės	24,88 ± 0,65	49
20 – 30 metų nesportuojantys vyrai	28,37 ± 0,56	26
20 – 30 metų nesportuojančios moterys	25,17 ± 0,44	66
30 – 40 metų nesportuojantys vyrai	35,36 ± 0,35	56
30 – 40 metų nesportuojančios moterys	35,14 ± 0,28	100
40 – 50 metų nesportuojantys vyrai	43,97 ± 0,60	29
40 – 50 metų nesportuojančios moterys	44,52 ± 0,50	44
Išemine širdies liga sergantys vyrai	52,44 ± 1,56	61

2.3. Informacinės posistemės realizacija Milio ir Muro automatų sąsūka

Bendrają principinę sąsūkos schemą sudarėme remdamiesi individualizuotu integralios organizmo reakcijos į fizinį krūvį ir formalizuotu fizinio krūvio poveikio modeliais iš Milio ir Muro automatų, kurių kiekvienas susideda iš 2 poautomatų.

Milio automato vykdomi veiksmai atitinka virtualų tyrimą: I Milio poautomatis skirtas pradinių duomenų, gautų tiesiogiai išmatavus tam tikrus parametrus, sutvarkymui. II Milio poautomatis skirtas išvestinių ir integralinių parametru skaičiavimui bei šių duomenų sutvarkymui. Pastebėsime, kad abu Milio poautomatai dirba nuosekliai: II poautomatis nepradedą dirbti tol, kol negauna signalo iš I Milio poautomatėio.

Muro automatas atitinka virtualaus gydytojo sprendimą: asmens sveikatos vertinimą ir tolimesnių sprendimų koordinavimą. I Muro poautomatis skirtas suminio vertinimo skaičiavimui, II – sveikatos būklės įvertinimui. Šie abu poautomatai taip pat dirba nuosekliai. Bendroji principinė sąsūkos schema pateikta 2.1 paveiksle.



2.1 pav. Informacinės posistemės bendroji pricipinė schema Milio ir Muro automatų sąsūka

Formalizuokime automatų sąsūką:

Z' – I Milio poautomato būsenų aibė, Z'' – II Milio poautomato būsenų aibė, W' – I Muro poautomato būsenų aibė, W'' – II Muro poautomato būsenų aibė. Y' – I Milio poautomato išėjimo II Milio poautomato įėjimo signalų aibė, Y – Milio automato išėjimo Muro automato įėjimo signalų aibė, X – Muro automato išėjimo – Milio įėjimo signalų aibė. Be to $x_n \in X, y'_n \in Y', y \in Y, z'_n \in Z', z''_n \in Z'', w'_n \in W', w''_n \in W''$. Automatų sąsūkos siurjekcijos:

$$g'_r: W' \times Y \rightarrow W',$$

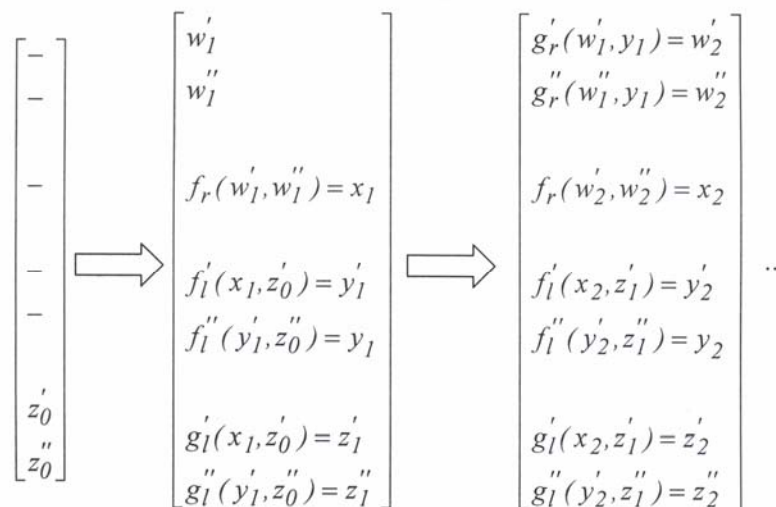
$$g''_r: W'' \times Y \rightarrow W''$$

$$f_r: W', W'' \rightarrow X,$$

$$\begin{aligned}
 f'_1: X \times Z' &\rightarrow Y', \\
 f''_1: Y' \times Z'' &\rightarrow Y, \\
 g'_1: X \times Z' &\rightarrow Z', \\
 g''_1: X \times Z'' &\rightarrow Z''.
 \end{aligned}
 \tag{2.1}$$

Automatų sąsūkos darbo pradžioje $n = 1$. Sąsūka pradeda veikti nustačius pradines būsenas w'_1, w''_1, z'_0, z''_0 ir gali dirbti neapibrėžtą laiką. Reikalinga tarpusavyje suderinti Milio ir Muro automatų pradines būsenas. Priešingu atveju gali būti gauti nekorektiški rezultatai.

Automatų sąsūkos darbo procesą galima pavaizduoti taip:



2.2 pav. Informacinės posistemės darbo realizacija Milio ir Muro automatų sąsūka

Pateiksime kiekvieno poautomatžio charakteristikas.

Iš kompiuterio laikmenos (ar įvedamus klaviatūra) duomenis susitarkime žymėti pagal parametro pavadinimą (2.2. skyrelis) atitinkamai: $\check{S}SD^k_j$ – kur indeksas k žymi fizinio krūvio etapą, o indeksas j – krūvio laiptelio numerį (j kinta nuo 0 iki 6). Analogiškai $\check{S}SD^a_l$ – kur indeksas a žymi atsigavimo etapą, o indeksas l – atsigavimo minutės eilės numerį (1 kinta nuo 0 iki 4). Simboliu Inf žymėsime duomenų nebuvimą (tiriamųjų išvystytas galingumas ir atsigavimo minučių skaičius skirtingas). Analogišką žymėjimą taikysime ir kitiems parametrams, nes informacinės posistemės kūrimui naudojome ne tik tiesiogiai iš veloergometrinio mėginio gautus parametrus, bet juos papildėme išvestiniais, skaičiuodami juos iš tiesiogiai išmatuotų: $RR, JT/RR, (S-D), (S-D)/S, DS$, bei integraliniais parametrais, kuriems bus įvedamas tik tyrimo numerio indeksas i : $Sv_i, Nvid_i, Niš_i, Nvisas_i, PWC170_i, N170_i, PPI_i, PRI_i, n.k._i$. Pirsono koreliacijos koeficientu ($Sdcorr_i$), skaičiuojamu tarp sistolinio ir

diastolinio kraujospūdžio viso tyrimo metu. Taip pat įvedėme tam tikrus funkcinius ryšius (juos aptarsime vėliau) tarp atskirų parametrų naudodamiesi mažiausių kvadratų metodu.

Aprašysime vaizdavimus:

$Z' \in Z$ (sveikųjų neneigiamų skaičių aibė) – I Milio poautomačio būsenų aibė, kurios elementai atlieka paciento tyrimų skaitiklio vaidmenį.

$z'_0 = 0$ (pradinė I Milio poautomačio būseną).

$Z'' \in Z$ (sveikųjų neneigiamų skaičių aibė) – II Milio poautomačio būsenų aibė.

$z''_0 = 0$ (pradinė II Milio poautomačio būseną)

$i = 0$ (tyrimo numerio skaitiklis, $i = n - 1$)

I Milio poautomačio būsenų perėjimo funkcijos aprašymas

$$g'_i: z'_{i+1} = \begin{cases} z'_i + 1, & \text{kai } x_i > 0 \text{ arba } i = 0, \text{ čia } i - \text{paciento tyrimų numeris } i = 0, 1, \dots, \infty \\ \text{STOP.} \end{cases}$$

I Milio poautomačio išėjimo funkcija f'_i susideda iš keliolikos funkcijų. Išėjimo signalo formavimas atliekamas iš eilės pagal išėjimo funkcijų numeraciją:

$$f'_i = (f'_{i0}, f'_{i1}, f'_{i2}, f'_{i3}, f'_{i4}, f'_{i5}, f'_{i6}, f'_{i7}, f'_{i8}, f'_{i9}, f'_{i0}, f'_{i1}, f'_{i2}, f'_{i3}, f'_{i4}, f'_{i5}, f'_{i6}, f'_{i7}, f'_{i8}, f'_{i9}).$$

Šios funkcijos darbo rezultatas išėjimo signalas y'_n . Pateikiame detalesnę signalo sudarymą:

$$f'_{i10}: \text{Lytis}_i = \begin{cases} 1, & \text{jei tiriamasis vyras} \\ 0, & \text{jei tiriamasis moteris} \end{cases}, \quad i = z'_i \text{ (paimama iš duomenų laikmenos arba įvedama}$$

klaviatūra)

Sudaromi visų pagrindinių parametrų vektoriai (jei tyrimų yra daugiau, sąsūkos darbo metu sudaromos matricos). Taigi II Milio poautomačio įėjimo signalų aibę Y' sudarys iš funkcijos f'_i gautų vektorių, kurių kai kurios komponentės yra įdėtieji masyvai, aibė.

Duomenys nuskaitomi į buferį, juos paimant iš duomenų laikmenos arba įvedant klaviatūra:

$$f'_{i11}: \check{SSD}_{ij}^k = \begin{cases} \check{SSD}_j^k, & \text{jeigu } \check{SSD}_j^k \neq \text{Inf}, j = 0, 1, \dots, 6, i = z'_i, \\ 999, & \text{jeigu } \check{SSD}_j^k = \text{Inf}, j = 0, 1, \dots, 6, i = z'_i. \end{cases}$$

$$f'_{i12}: \check{SSD}_{il}^a = \begin{cases} \check{SSD}_l^a, & \text{jeigu } \check{SSD}_l^a \neq \text{Inf}, l = 0, 1, \dots, 4, i = z'_i, \\ 999, & \text{jeigu } \check{SSD}_l^a = \text{Inf}, l = 0, 1, \dots, 4, i = z'_i. \end{cases}$$

$$f'_{13}: S_{ij}^k = \begin{cases} S_j^k, \text{jeigu } S_j^k \neq \text{Inf}, j=0,1,\dots,6, i=z'_i, \\ 999, \text{jeigu } S_j^k = \text{Inf}, j=0,1,\dots,6, i=z'_i. \end{cases}$$

$$f'_{14}: S_{il}^a = \begin{cases} S_l^a, \text{jeigu } S_l^a \neq \text{Inf}, l=0,1,\dots,4, i=z'_i, \\ 999, \text{jeigu } S_l^a = \text{Inf}, l=0,1,\dots,4, i=z'_i. \end{cases}$$

$$f'_{15}: D_{ij}^k = \begin{cases} D_j^k, \text{jeigu } D_j^k \neq \text{Inf}, j=0,1,\dots,6, i=z'_i, \\ 999, \text{jeigu } D_j^k = \text{Inf}, j=0,1,\dots,6, i=z'_i. \end{cases}$$

$$f'_{16}: D_{il}^a = \begin{cases} D_l^a, \text{jeigu } D_l^a \neq \text{Inf}, l=0,1,\dots,4, i=z'_i, \\ 999, \text{jeigu } D_l^a = \text{Inf}, l=0,1,\dots,4, i=z'_i. \end{cases}$$

$$f'_{17}: JT_{ij}^k = \begin{cases} JT_j^k, \text{jeigu } JT_j^k \neq \text{Inf}, j=0,1,\dots,6, i=z'_i, \\ 999, \text{jeigu } JT_j^k = \text{Inf}, j=0,1,\dots,6, i=z'_i. \end{cases}$$

$$f'_{18}: JT_{il}^a = \begin{cases} JT_l^a, \text{jeigu } JT_l^a \neq \text{Inf}, l=0,1,\dots,4, i=z'_i, \\ 999, \text{jeigu } JT_l^a = \text{Inf}, l=0,1,\dots,4, i=z'_i. \end{cases}$$

$$f'_{19}: ST_{ij}^k = \begin{cases} ST_j^k, \text{jeigu } ST_j^k \neq \text{Inf}, j=0,1,\dots,6, i=z'_i, \\ 999, \text{jeigu } ST_j^k = \text{Inf}, j=0,1,\dots,6, i=z'_i. \end{cases}$$

$$f'_{110}: ST_{il}^a = \begin{cases} ST_l^a, \text{jeigu } ST_l^a \neq \text{Inf}, l=0,1,\dots,4, i=z'_i, \\ 999, \text{jeigu } ST_l^a = \text{Inf}, l=0,1,\dots,4, i=z'_i. \end{cases}$$

$$f'_{111}: QRS_{ij}^k = \begin{cases} QRS_j^k, \text{jeigu } QRS_j^k \neq \text{Inf}, j=0,1,\dots,6, i=z'_i, \\ 999, \text{jeigu } QRS_j^k = \text{Inf}, j=0,1,\dots,6, i=z'_i. \end{cases}$$

$$f'_{112}: QRS_{il}^a = \begin{cases} QRS_l^a, & \text{jeigu } QRS_l^a \neq Inf, l=0,1,\dots,4, i=z'_i, \\ 999, & \text{jeigu } QRS_l^a = Inf, l=0,1,\dots,4, i=z'_i. \end{cases}$$

$f'_{113}: Zing_i = l_dydis, i = z'_i$ (i-ojo tyrimo protokolo galingumo kėlimo laiptelio dydis 25 W arba 50 W, paimamas iš duomenų laikmenos)

$f'_{114}: Nmax_i = N_max, i = z'_i$ (i-ojo tyrimo metu išvystytas maksimalus galingumas, W, paimamas iš duomenų laikmenos arba įvedamas klaviatūra)

$f'_{115}: n_i = \begin{cases} N_max/Zing_i + 1, & \text{jeigu } Zing_i = 50, \\ N_max/Zing_i, & \text{jeigu } Zing_i = 25. \end{cases}, i = z'_i$ (i-ojo tyrimo metu išvystytų skirtingų galingumų skaičius).

$$f'_{116}: N_{ij} = \begin{cases} 0, & \text{kai } j = 0; \\ 50, & \text{kai } j = 1; \\ 50 + Zing_i, & 1 < j < 7. \end{cases}, i = z'_i \text{ (sudaromas i-ojo tyrimo metu išvystytų skirtingų galingumų vektorius).}$$

$f'_{117}: At_i = \text{atsigavimo minučių skaičius}, i = z'_i$ (i-ojo tyrimo metu atsigavimo trukmė minutėmis, paimama iš duomenų laikmenos arba įvedamas klaviatūra).

$f'_{118}: m = \text{svoris}, i = z'_i$ (i-ojo tyrimo metu paciento svoris, kg, paimamas iš duomenų laikmenos arba įvedamas klaviatūra).

$f'_{119}: a_i = \text{amžius}, i = z'_i$ (i-ojo tyrimo metu paciento amžius, metais, paimamas iš duomenų laikmenos arba įvedamas klaviatūra).

II Milio poautomatžio būsenų perėjimo funkcijos aprašymas:

$$g''_1: z''_{i+1} = \begin{cases} z''_i + 1, & \text{kai } y_{i+1} \neq Inf, \text{ čia } i \text{ paciento tyrimų numeris } i = 0, 1, \dots, \infty \\ STOP \end{cases}$$

$f''_1 = (f''_{11}, f''_{12}, f''_{13}, f''_{14}, f''_{15}, f''_{16}, f''_{17}, f''_{18}, f''_{19}, f''_{10}, f''_{111}, f''_{112}, f''_{113}, f''_{114}, f''_{115}, f''_{116}, f''_{117}, f''_{118}, f''_{119}, f''_{120}, f''_{121}, f''_{122}, f''_{123}, f''_{124}, f''_{125})$ – II Milio poautomatžio išėjimo funkcija (išėjimo signalai). Išėjimo signalo (y_n) formavimas atliekamas iš eilės pagal išėjimo funkcijų numeraciją. Taigi II Milio

poautomatėio išėjimo signalų aibę Y sudarys iš funkcijos f''_i gautų vektorių, kurių kai kurios komponentės yra įdėtieji masyvai, aibė.

Sudaromi visų išvestinių parametrų (2.2. skyrelis) vektoriai (jei tyrimų yra daugiau, sąsūkos darbo metu gaunamos matricos):

$$f''_{11}: RR_{ij}^k = \begin{cases} \frac{60}{\check{S}SD_{ij}^k}, \text{ jeigu } \check{S}SD_{ij}^k \neq 999, j=0,1,\dots,6, i=z_i'', \\ 999, \text{ jeigu } \check{S}SD_{ij}^k = 999, j=0,1,\dots,6, i=z_i''. \end{cases}$$

$$f''_{12}: RR_{il}^a = \begin{cases} \frac{60}{\check{S}SD_{il}^a}, \text{ jeigu } \check{S}SD_{il}^a \neq 999, l=0,1,\dots,4, i=z_i'', \\ 999, \text{ jeigu } \check{S}SD_{il}^a = 999, l=0,1,\dots,4, i=z_i''. \end{cases}$$

$$f''_{13}: (JT/RR)_{ij}^k = \begin{cases} \frac{JT_{ij}^k}{RR_{ij}^k}, \text{ jeigu } RR_{ij}^k \neq 999, j=0,1,\dots,6, i=z_i'', \\ 999, \text{ jeigu } RR_{ij}^k = 999, j=0,1,\dots,6, i=z_i''. \end{cases}$$

$$f''_{14}: (JT/RR)_{il}^a = \begin{cases} \frac{JT_{il}^a}{RR_{il}^a}, \text{ jeigu } RR_{il}^a \neq 999, l=0,1,\dots,4, i=z_i'', \\ 999, \text{ jeigu } RR_{il}^a = 999, l=0,1,\dots,4, i=z_i''. \end{cases}$$

$$f''_{15}: (S-D)_{ij}^k = \begin{cases} S_{ij}^k - D_{ij}^k, \text{ jeigu } S_{ij}^k \neq 999 \text{ ir } D_{ij}^k \neq 999, j=0,1,\dots,6, i=z_i'', \\ 999, \text{ jeigu } S_{ij}^k = 999 \text{ arba } D_{ij}^k = 999, j=0,1,\dots,6, i=z_i''. \end{cases}$$

$$f''_{16}: (S-D)_{il}^a = \begin{cases} S_{il}^a - D_{il}^a, \text{ jeigu } S_{il}^a \neq 999 \text{ ir } D_{il}^a \neq 999, l=0,1,\dots,4, i=z_i'', \\ 999, \text{ jeigu } S_{il}^a = 999 \text{ arba } D_{il}^a = 999, l=0,1,\dots,4, i=z_i''. \end{cases}$$

$$f''_{17}: ((S-D)/S)_{ij}^k = \begin{cases} (S_{ij}^k - D_{ij}^k) / S_{ij}^k, \text{ jeigu } S_{ij}^k \neq 999 \text{ ir } D_{ij}^k \neq 999, j=0,1,\dots,6, i=z_i'', \\ 999, \text{ jeigu } S_{ij}^k = 999 \text{ arba } D_{ij}^k = 999, j=0,1,\dots,6, i=z_i''. \end{cases}$$

$$f''_{18}: ((S-D)/S)_{il}^a = \begin{cases} (S_{il}^a - D_{il}^a) / S_{il}^a, \text{ jeigu } S_{il}^a \neq 999 \text{ ir } D_{il}^a \neq 999, j=0,1,\dots,4, i=z_i'', \\ 999, \text{ jeigu } S_{il}^a = 999 \text{ arba } D_{il}^a = 999, j=0,1,\dots,4, i=z_i''. \end{cases}$$

$$f''_{19}: DS_{ij}^k = \begin{cases} \check{S}SD_{ij}^k \cdot S_{ij}^k \cdot 10^{-2}, \text{ jeigu } S_{ij}^k \neq 999 \text{ ir } \check{S}SD_{ij}^k \neq 999, j=0,1,\dots,6, i=z_i'', \\ 999, \text{ jeigu } S_{ij}^k = 999 \text{ arba } \check{S}SD_{ij}^k = 999, j=0,1,\dots,6, i=z_i''. \end{cases}$$

$$f''_{10}: DS_{il}^a = \begin{cases} \check{S}SD_{il}^a \cdot S_{il}^a \cdot 10^{-2}, \text{ jeigu } S_{il}^a \neq 999 \text{ ir } \check{S}SD_{il}^a \neq 999, j=0,1,\dots,4, i=z_i'', \\ 999, \text{ jeigu } S_{il}^a = 999 \text{ arba } \check{S}SD_{il}^a = 999, j=0,1,\dots,4, i=z_i''. \end{cases}$$

$\check{S}SD$ priklausomybę nuo N aproksimavome tiesine funkcija mažiausių kvadratų metodu. Aproksimacijos parametrų radimui naudota empirinė formulė:

$$\check{S}SD(N) = K \cdot N + L. \quad (2.2)$$

čia K, L – aproksimacijos parametrai. Jie skaičiuojami kiekvienam asmens tyrimui (K_i, L_i , kur i – tyrimo numeris).

$$f''_{111}: K_i = \frac{\sum_{m=0}^{n_i-1} N_{im} \sum_{m=0}^{n_i-1} \check{S}SD_{im}^k - n_i \sum_{m=0}^{n_i-1} N_{im} \cdot \check{S}SD_{im}^k}{\left(\sum_{m=0}^{n_i-1} N_{im} \right)^2 - n_i \sum_{m=0}^{n_i-1} N_{im}^2}, \quad i=z_i''.$$

$$f''_{112}: L_i = \frac{\sum_{m=0}^{n_i-1} N_{im} \cdot \check{S}SD_{im}^k \sum_{m=0}^{n_i-1} N_{im} - \sum_{m=0}^{n_i-1} \check{S}SD_{im}^k \sum_{m=0}^{n_i-1} N_{im}^2}{\left(\sum_{m=0}^{n_i-1} N_{im} \right)^2 - n_i \sum_{m=0}^{n_i-1} N_{im}^2}, \quad i=z_i''.$$

Žinodami arterinio kraujospūdžio tiesioginio matavimo sunkumus, remdamiesi savo įgyta patirtimi (2,3,4) funkcinę priklausomybę tarp DS ir JT fizinio krūvio metu postulavome kaip hiperbolinį ir tiesinį ryšį, priklausomai nuo išvystyto galingumo:

$$DS(JT) = \begin{cases} \frac{A}{JT} + B, & \text{kai } N > 50W, \\ M \cdot JT + D, & \text{kai } N \leq 50W. \end{cases} \quad (2.3)$$

čia A, B, D, M – aproksimavimo metu gauti parametrai.

Naudojantis vieno tyrimo metu gautų dydžių pagrindu apskaičiuotais parametrais A, B, D, M , kito tyrimo metu galima skaičiuoti sistolinį spaudimą (nereikia tiesioginių matavimų).

Aproksimacijos parametrai A, B, D, M skaičiuojami kiekvienam asmens tyrimui A_i, B_i, M_i, D_i , čia i – tyrimo numeris:

$$f''_{113}: M_i = \frac{DS_{i1}^k - DS_{i0}^k}{JT_{i1}^k - JT_{i0}^k}, \quad i = z_i''.$$

$$f''_{114}: D_i = \frac{JT_{i0}^k \cdot DS_{i1}^k - JT_{i1}^k \cdot DS_{i0}^k}{JT_{i1}^k - JT_{i0}^k}, \quad i = z_i''.$$

$$f''_{115}: A_i = \frac{(n_i - 1) \sum_{m=1}^{n_i-1} \frac{DS_{im}^k}{JT_{im}^k} - \sum_{m=1}^{n_i-1} DS_{im}^k \sum_{m=1}^{n_i-1} \frac{1}{JT_{im}^k}}{(n_i - 1) \sum_{m=1}^{n_i-1} \frac{1}{(JT_{im}^k)^2} - \left(\sum_{m=1}^{n_i-1} \frac{1}{JT_{im}^k} \right)^2}, \quad i = z_i''.$$

$$f''_{116}: B_i = \frac{\sum_{m=1}^{n_i-1} \frac{1}{(JT_{im}^k)^2} \sum_{m=1}^{n_i-1} DS_{im}^k - \sum_{m=1}^{n_i-1} \frac{DS_{im}^k}{JT_{im}^k} \sum_{m=1}^{n_i-1} \frac{1}{JT_{im}^k}}{(n_i - 1) \sum_{m=1}^{n_i-1} \frac{1}{(JT_{im}^k)^2} - \left(\sum_{m=1}^{n_i-1} \frac{1}{JT_{im}^k} \right)^2}, \quad i = z_i''.$$

Integralinių parametų skaičiavimas:

$$f''_{117}: Nvid_i = \frac{L_i}{K_i \cdot m_i}, \quad i = z_i'';$$

$$f''_{118}: Niš_i = \frac{N \max_i}{m_i}, \quad i = z_i'';$$

$$f''_{119}: N170_i = \frac{170 - L_i}{K_i}, \quad i = z_i''.$$

$$f''_{120}: PWC170_i = \frac{N170_i}{m_i}, \quad i = z_i''.$$

$$f''_{121}: Nvisas_i = N_{i\check{s}} + N_{vid}, \quad i = z_i''.$$

$$f''_{122}: PPI_i = \frac{1}{(n_i - 1)} \sum_{m=0}^{n_i-2} \frac{S_{i(m+1)}^k - S_{im}^k}{N_{i(m+1)} - N_{im}}, \quad i = z_i''.$$

$$f''_{123}: PRI_i = \frac{1}{(n_i - 1)} \sum_{m=0}^{n_i-2} \frac{S_{i(m+1)}^k - S_{im}^k}{\check{S}SD_{i(m+1)}^k - \check{S}SD_{im}^k}, \quad i = z_i''.$$

$$f''_{124}:$$

$$SDcorf_i = \frac{\frac{1}{(n_i + At_i)} \left(\sum_{m=0}^{n_i-1} S_{im}^k D_{im}^k + \sum_{m=0}^{At_i} S_{im}^a D_{im}^a \right) - \frac{1}{(n_i + At_i)} \left(\sum_{m=0}^{n_i-1} S_{im}^k + \sum_{m=0}^{At_i} S_{im}^a \right) \cdot \frac{1}{(n_i + At_i)} \left(\sum_{m=0}^{n_i-1} D_{im}^k + \sum_{m=0}^{At_i} D_{im}^a \right)}{\sqrt{\frac{1}{(n_i + At_i)} \left(\sum_{m=0}^{n_i-1} (S_{im}^k)^2 + \sum_{m=0}^{At_i} (S_{im}^a)^2 \right) - \left(\frac{1}{(n_i + At_i)} \left(\sum_{m=0}^{n_i-1} S_{im}^k + \sum_{m=0}^{At_i} S_{im}^a \right) \right)^2}} \times$$

$$\times \frac{1}{\sqrt{\frac{1}{(n_i + At_i)} \left(\sum_{m=0}^{n_i-1} (D_{im}^k)^2 + \sum_{m=0}^{At_i} (D_{im}^a)^2 \right) - \left(\frac{1}{(n_i + At_i)} \left(\sum_{m=0}^{n_i-1} D_{im}^k + \sum_{m=0}^{At_i} D_{im}^a \right) \right)^2}}, \quad i = z_i''.$$

$$f''_{125}:$$

$$Sv_i = \begin{cases} \frac{1}{1,32} \sqrt{\left(\frac{RR_{i0}^k - RR_{i(n_i-1)}^k}{RR_{i0}^k - \frac{60}{220 - a_i}} \right)^2 + \left(\frac{JT_{i0}^k - JT_{i(n_i-1)}^k}{0,26} \right)^2 + \left(\frac{\sum_{m=1}^{n_i-1} N_{im}}{(n_i - 2) \cdot 200} \right)^2} \cdot 100\%, \text{ kai Lytis } i = 1, \text{ Zing } i = 50; \\ \frac{1}{1,34} \sqrt{\left(\frac{RR_{i0}^k - RR_{i(n_i-1)}^k}{RR_{i0}^k - \frac{60}{220 - a_i}} \right)^2 + \left(\frac{JT_{i0}^k - JT_{i(n_i-1)}^k}{0,26} \right)^2 + \left(\frac{\sum_{m=1}^{n_i-1} N_{im}}{(n_i - 2) \cdot 200} \right)^2} \cdot 100\%, \text{ kai Lytis } i = 2, \text{ Zing } i = 50; \\ \frac{1}{1,32} \sqrt{\left(\frac{RR_{i0}^k - RR_{i(n_i-1)}^k}{RR_{i0}^k - \frac{60}{220 - a_i}} \right)^2 + \left(\frac{JT_{i0}^k - JT_{i(n_i-1)}^k}{0,26} \right)^2 + \left(\frac{\sum_{m=1}^{n_i-1} N_{im}}{(n_i - 2) \cdot 150} \right)^2} \cdot 100\%, \text{ kai Lytis } i = 1, \text{ Zing } i = 25; \\ \frac{1}{1,34} \sqrt{\left(\frac{RR_{i0}^k - RR_{i(n_i-1)}^k}{RR_{i0}^k - \frac{60}{220 - a_i}} \right)^2 + \left(\frac{JT_{i0}^k - JT_{i(n_i-1)}^k}{0,26} \right)^2 + \left(\frac{\sum_{m=1}^{n_i-1} N_{im}}{(n_i - 2) \cdot 150} \right)^2} \cdot 100\%, \text{ kai Lytis } i = 2, \text{ Zing } i = 25. \end{cases}$$

Muro automato aprašymas

$W' = \{0;1;2;3\}$ – I Muro poautomato būsenų aibė, reiškianti tiriamo asmens sveikatos suminį vertinimo palyginimą su prieš tai buvusiu tyrimu.

$$g'_r : w'_{i+1} = \begin{cases} 0, & \text{kai } i = 0; \\ 1, & \text{kai } Sv_{i+1} < Sv_i, i > 0; \\ 2, & \text{kai } Sv_{i+1} = Sv_i, i > 0; \\ 3, & \text{kai } Sv_{i+1} > Sv_i, i > 0. \end{cases}$$

čia i – tyrimo numeris, Sv_i – i -ojo tyrimo suminis sveikatos vertinimas.

$w'_0 = 0$ (pradinė I Muro poautomato būseną)

$W'' = \{0;1;2;3;4;5\}$ II Muro poautomato būsenų aibė, reiškianti sveikatos suminio vertinimo grupę.

$w''_0 = 0$ (pradinė II Muro poautomato būseną)

$$g''_r : w''_{i+1} = \begin{cases} 0, & \text{kai } Sv_i = 0; \\ 1, & \text{kai } 0 < Sv_i < 40; \\ 2, & \text{kai } 40 \leq Sv_i < 50; \\ 3, & \text{kai } 50 \leq Sv_i < 70; \\ 4, & \text{kai } 70 \leq Sv_i < 80; \\ 5, & \text{kai } 80 \leq Sv_i \leq 100. \end{cases}$$

Muro automato išėjimo signalų aibė – X (sveikieji neneigiami skaičiai). Jos elementai gaunami tokiu būdu

$$f''_r : x_{i+1} = \begin{cases} 0, & \text{jeigu } w''_{i+1} = 0; \\ w''_{i+1} \# w'_{i+1}, & \text{kitais atvejais.} \end{cases}$$

čia simbolis $\#$ reiškia konkatenaciją, pavyzdžiui, $1\#0 = 10$.

Taip apibrėžę Muro automato išėjimo signalų aibę, paprastai apibūdiname paciento sveikatos būklę. Pavyzdžiui, jeigu $x_{i+1} = 31$, tai reiškia, kad šio tyrimo metu paciento būklė normali (patenkinama), tačiau ji pablogėjo lyginant su praeitu tyrimu.

Taip apibrėžiant Muro automato išėjimus, paprasta modifikuoti informacinės posistemės struktūrą – galima įvesti daugiau Muro poautomatų, kurie vertintų ir nagrinėtų kitus integralinius parametrus. Tokiu atveju paciento būklę būtų galima apibūdinti išsamiau. Tačiau, kiti integraliniai parametrai mažai tyrinėti medicinoje, jų skaičiavimo metodikos skiriasi nuo pateiktos šiame darbe,

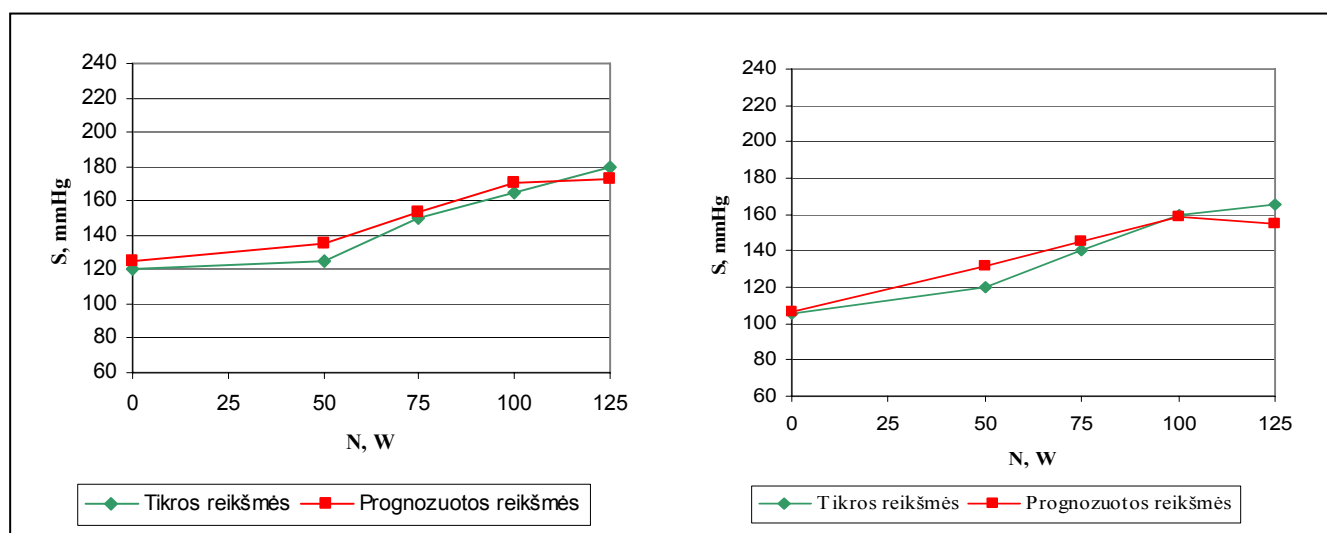
todėl nusakyti šių prasnę ir kitimo ribas sudėtinga (34). Integraliniai parametrai skirtingoms tiriamųjų grupėms pateikti 1 priede.

2.4. Sistolinio kraujo spaudimo skaičiavimas

Fizinio krūvio metu tiesiogiai matuoti kraujospūdį sudėtinga. Kurdami informacinę posistemę atsižvelgėme į šias problemas. Pagal (2.3) formulę pirmojo tyrimo metu gautų dydžių pagrindu apskaičiuotais parametrais A, B, M, D tolimesnių tyrimų metu galima apskaičiuoti sistolinį kraujo spaudimą (iš tos pačios formulės išreiškus S).

Norėdami patikrinti metodo efektyvumą moterims sportininkėms mėgėjomis apskaičiuotą sistolinio kraujo spaudimo reikšmę lyginome su antrojo tyrimo metu matuota realia reikšme ir iš jų absoliutinio skirtumo skaičiavome prognozės santykinę paklaidą procentais. Nustatėme, kad stipriai keičiantis tiriamojo funkcinėi būklei (S_v pasikeičia ne mažiau 10%), skaičiavimų santykinė procentinė paklaida išauga iki 20%. Jei tyrimai atliekami dažnai (vieno mėnesio laikotarpiu) ir tiriamojo funkcinė būklė keičiasi mažiau nei 10% sistolinis arterinis kraujo spaudimas gali būti apskaičiuojamas tiksliau – vidutiniškai 10% tikslumu. Detalesnė sistolinio kraujo spaudimo apskaičiavimo analizė pateikta literatūroje (3,4,5).

Vaizdumo dėlei pateikiame dviejų moterų (sportininkių mėgėjų grupė) realaus ir apskaičiuoto sistolinio spaudimo grafikus (2.3 pav.).



2.3 pav. Sistolinio kraujo spaudimo dinamika

Pateikta metodika leidžia vertinti sistolinį kraujo spaudimą ekstremalaus fizinio aktyvumo sporte metu, kuomet tiesioginis matavimas yra neįmanomas, tačiau EKG registruojama. Aišku, kad

pristatytas tyrimas yra tik pirmas žingsnis ieškant didelės problemos – neinvazinių arterinio kraujo spaudimo matavimo su priimtinu tikslumu – sprendimų.

2.5. Programos aprašymas

Pagal Milio ir Muro automatų sąsūkos formalųjį aprašą buvo sukurta taikomoji programa, kurios darbo principas artimas virtualaus gydytojo darbui. Kadangi labai sudėtinga sukurti virtualų gydytoją, kuris puikiai formuluotų įvairiapuses diagnozes buvo pasirinktas dvigubas programos darbo principas: Milio ir Muro automatų sąsūka realizuota kaip virtualus gydytojas (Milio automatui tenka paciento duomenys, o Muro – išvados apie paciento sveikatos būklę), greta pateikiant vizualizacinę medžiagą kiekvienai sąsūkos darbo pakopai, iliustruojant ją papildomais dydžiais. Vienas pagrindinių programos bruožų – galima modifikuoti neišardant struktūros, papildyti automatų protokolus.

Milio ir Muro automatų sąsūka buvo pritaikyta visiems tyrimams.

Pradiniai duomenys saugomi duomenų bazėje, kurią sudaro trys lentelės: „Gydytojai.db”, „Pacientai.db”, „Irasai.db”. Lentelių pavadinimai tiesiogiai atspindi jų saugomos informacijos turinį, t.y. lentelėje „Gydytojai.db” saugoma informacija apie gydytoją, „Pacientai.db” – informacija apie pacientus, o lentelėje „Irasai.db” – tai visi pacientams atlikti tyrimai ir jų duomenys.

Programa pritaikyta dirbti vienam gydytojui (nėra prisijungimo slaptažodžio), tačiau organizuota struktūra lengvai leistų programą modifikuoti ir kiekvienas skirtingas gydytojas galėtų dirbti tik su savo pacientų duomenimis.

Programa buvo kuriama bendradarbiaujant su kolege Dovile Ašeriškyte, kadangi mūsų atliekamų darbų vadovas ir užsakovas yra tas pats asmuo. Jis pageidavo, kad programinė dalis būtų apjungta. Todėl pirmame programos lange pažymėta, jog darbą atliko du asmenys. Šalia pavardžių yra mygtukai, nukreipiantys į skirtingas, savarankiškas programos dalis (žr. 1 pav. 2 priede) .

Programos funkcijos:

Pagrindiniame programos lange pateikiami pradiniai duomenys (žr. 2 pav. 2 priede).

Taip pat lange yra mygtukai, visoms programos funkcijoms atlikti:

- Paspaudus mygtuką „*Naujas*” atsiveria naujas langas, skirtas įvesti naujo paciento duomenis.
- Paspaudus mygtuką „*Keisti*” atsiveria naujas langas su paciento duomenimis, kuriuos galima pakeisti.

- Paspaudus mygtuką „*Išvalyti*” ištrinami visi paciento duomenys (visi tyrimai). Prieš trindama duomenis, programa prašo veiksmo patvirtinimo.
- Paspaudus mygtuką „*Importuoti*” atsiveria dialogo langas, leidžiantis atidaryti tam tikros struktūros tekstinę bylą ir jos duomenis perkelti į duomenų bazę.
- Paspaudus mygtuką „*Eksportuoti*” atsiveria dialogo langas, leidžiantis sukurti tam tikros struktūros tekstinę bylą, į kurią surašomi duomenys, ir ją išsaugoti norimoje kompiuterio disko vietoje.
- Laukelyje „*Ieškoti*” galima atlikti paciento paiešką pagal pavardę.
- Mygtukas „*Filtrai*” atidaro naują langą, skirtą norimos grupės tyrimų parinkimui. Galima peržiūrėti įvairią informaciją, ieškant tendencijų būdingų atitinkamos grupės tiriamiesiems. Tam tereikia pasirinkti tiriamųjų lytį (yra galimybė analizuoti abiejų lyčių tiriamuosius) bei grupę iš sąrašų, pateikiamų ekrane.
- Paspaudus mygtuką „*Naujas*” atsiveria naujas langas, leidžiantis įvesti naujo tyrimo duomenis (žr. 3 pav. 2 priede). Pastebėsime, kad įvedant duomenis atliekama duomenų kontrolė: negalima palikti tuščių laukų, vietoje skaičių vesti simbolių ir atvirksiai. Vartotojo patogumui kai kurie laukai standartizuoti. Nuo maksimalaus galingumo, atsigavimo trukmės priklauso kiek laukų reikės užpildyti (nereikalingi paslepiami). Jeigu įvedamas ne pirmasis paciento tyrimas, galima skaičiuoti, o ne matuoti sistolinį kraujo spaudimą
- Paspaudus mygtuką „*Ištrinti tyrimą*” ištrinami vieno tyrimo duomenys. Prieš trindama duomenis, programa prašo veiksmo patvirtinimo.
- Paspaudus mygtuką „*Analizuoti*” pradeda veikti automatų sąsūka. Jos darbo rezultatai pateikiami naujame paciento duomenų analizės lange (žr. 4,5,6 pav. 2 priede). Šis langas susideda iš 6 skirtingų puslapių, skirtų pradinių, išvestinių, integralinių duomenų peržiūrai (Milio automato išėjimo signalas), pradinių, išvestinių duomenų grafinei analizei fizinio krūvio ir atsigavimo metu (naudojama interpoliacija), $\dot{SSD}(N)$ analizei, sistolinio ir diastolinio arterinio kraujo spaudimo analizei, įvairių parametrų kombinacijų

ir Sv analizei (Muro automato išėjimas). Pastebėsime, kad greta pačių parametų kitimo grafikų pateikiami jų kitimo greičio ir pagreičio grafikai (gaunami naudojantis baigtiniais skirtumais ir skirtuminius santykiais, taip pat pateikiamos naudotų aproksimacijų funkcinės išraiškos, aproksimacijų santykinės procentinės paklaidos. Muro automato išėjimo signalo arba Sv analizės dalyje pateikiamos išvados, individualizuoto žmogaus fiziologinio modelio komponentų procentinis įvertinimas. Programa papildyta kiekvieno fizinio krūvio laiptelio analize – ieškoma fizinių galimybių riba (W), pagal atskirus parametrus:

2.3 lentelė

Parametrai ir jų reikšmės, reiškiančios fizinių galimybių ribą

Parametras	Fizinių galimybių riba
$\dot{S}SD$	> 220 – amžius, sportininkams > 0,85 (220 – amžius) , sveikiems asmenims > 0,75 (220 – amžius) , esant patologijai
S	> 240 arba jei S krenta bent 5 mmHg
D	> 120
JT	< 0,16 arba procentinis pokytis tarp dviejų laiptelių mažesnis nei 5 %

Ši analizė neįtraukta į sąsūkos protokolus, nes nėra visiškai vienareikšmė. Sunku patikrinti, kodėl tyrimas buvo tęsiamas, kai buvo peržengtos šios ribos, arba nutrauktas, nors jokių požymių nepastebėta. Čia įtakos turi ir tiriamojo savijauta, ir charakterio savybės (pasiduoda ar bando tęsti tyrimą).

- Paspaudus mygtuką „Palyginti“ taip pat dirba automatų sąsūka. Šio mygtuko galimybės panašios kaip ir prieš tai buvusiame punkte, tačiau atsiranda analizuojamo ir palyginamojo tyrimo pasirinkimas. Taigi galima palyginti duomenų dinamiką tyrimų metu (žr. 7,8 pav. 2 priede).
- Paspaudus mygtuką „Sv dinamika“ atskirame lange pateikiama suminio vertinimo dinamika visų paciento tyrimų metu (žr. 9 pav. 2 priede).

- Paspaudus mygtuką „Grupės“ atskirame lange pateikiama programos duomenų struktūra: tiriamieji skirstomi pagal lytį ir tam tikrą grupę. Duomenys pateikiami absoliučiomis reikšmėmis ir procentais (žr. 10 pav. 2 priede).

Programoje įdiegti Milio ir Muro automatų sąsūkos protokolai gali būti lengvai ir greitai modifikuojami, todėl nesudėtinga programą tobulinti.

IŠVADOS

1. Sukurta metodika, leidžianti aprašyti matematinėmis informacinėmis sąvokomis žmogaus integralios organizmo reakcijos į fizinių krūvių modelį, vadovaujantis formalizuota poveikio strategija.
2. Pagal Milio ir Muro automatų sąsūką realizuota taikomoji programa, skirta profesionaliam vartotojui – žmogaus fiziologijos specialistui. Sukurta automatų sąsūka leidžia formuoti valdomą grįžtamąjį ryšį formalizuojant gydymo procesą.
3. Ištirti parametrų $\dot{V}O_2(N)$, $DS(JT)$ funkciniai ryšiai – pasiūlyta metodika, leidžianti vertinti sistolinį kraujo spaudimą medikams priimtinu 10 % tikslumu ekstremalaus fizinio aktyvumo sporte metu, kuomet tiesioginis matavimas yra neįmanomas, tačiau EKG registruojama.
4. Sukurta informacinė posistemė gali būti pritaikyta e – sveikatos kūrime, modeliuojant virtualų gydytoją.

LITERATŪRA

1. 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.
2. Ašeriškytė D. Maksimalaus išvystomo 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.
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. 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.
6. Ašeriškytė D. Funkcinis ryšys tarp dvigubos sandaugos ir JT intervalo / Ašeriškytė D., Matusevičiūtė K., Vainoras A., Navickas Z. // Taikomoji matematika: V studentų konferencijos pranešimų medžiaga [Kauno technologijos universitetas, 2004 m. gegužės 8 d.] Kaunas, 2004. p. 10 – 12.
7. Halm MA. Primary prevention programs to reduce heart disease risk in women / Halm MA., Denker J. // Clinical Nurse Spec 2003 Mar; 17(2): 101–9.
8. O’Sullivan SE. The effects of exercise and training on human cardiovascular reflex control / O’Sullivan SE, Bell C. J. // Auton Nerv Syst 2000; 81:16–24.
9. Poderys J. Kineziologijos pagrindai – Mokomoji knyga. Kaunas, 2004. p. 234.
10. Vainoras A. Veloergometrija ir sisteminių vertinimų galimybė / Vainoras A., Gargasas L., Jaruševičius G., Šilanskienė A., Miškinis V., Ruseckas R., et. al. // Lithuanian journal of Cardiology 1999; 6(4): p. 760–763.

11. Gargasas L. JT interval changes during bicycle ergometry / Gargasas L., Vainoras A., Schvela H., Jarusevicius G., Ruseckas R., Miskinis V. // Abstracts of the 2nd International Congress Polish Cardiac Society 1998; 4(6): p.153.
12. Dubach P. Blood pressure reaction during exertion / Dubach P., Goebbels U., Ratti R. //Schweiz Med Wochenshr 1996; 126(33): p. 1400–1402.
13. Braam R.L . Home blood pressure measurement with oscillometric upper–arm devices / Braam R.L, Thien T.// Neth J Med 2003; 61(10): p. 307–312.
14. Palatini P. Blood pressures behavior during physical activity // J Sports Med 1998; 5(6): p. 353–374.
15. Glancy D.L. Exercise electrocardiography / Glancy D.L, Patterson C. // La State Med. Soc. 2003; 155(1): p. 26–35.
16. Vincent K. R. Strength training and hemodinamyc responses to exercise / Vincent K. R, Vincent H.K, Brath R. W, Bhatnagar V., Lowenthal D.T.Am J// Geriatr Cardiol 2003; 12(2): p. 97–106.
17. Okuda T. Effect of aiging on heart rate and blood pressure variability’s in healthy men and women / Okuda T., Satoh A, Oogaku S., Sekine M., Kitajima I.// Rinsho Byori 2002; 50(2): p. 186–190.
18. Plukas K. Skaitiniai metodai ir algoritmai. Kaunas, 2001. 549 p. ISBN 955 – 03 – 061 –5.
19. Kvedaras B. Skaičiavimo metodai / B. Kvedaras, M. Sapagovas. Vilnius, 1974. 515 p.
20. Pekarskas V. Diferencialinis ir integralinis skaičiavimas II dalis. Kaunas, 2000. 419 psl. ISBN 9986 – 13 – 716 – 0.
21. D.Silber. E–sveikata, ataskaita Ministrų lygio konferencijai // Briuselis, 2003. ISBN 90 – 6779 – 180 – 6.
22. Magosso E. Cardiovascular response to dynamic aerobic exercise: a mathematical model / Magosso E., Ursino M.// Medical & Biological Engineering & Computing, 2002, Vol. 40, p. 660 – 672.
23. Matematinis modelis „Pneuma“ ir jo dokumentacija. Prieiga per internetą:
<<http://bmsr.usc.edu/Software/Pneuma/overviewpneuma.html>>.
24. Matematinis modelis „Physbe“ ir jo dokumentacija. Prieiga per internetą:
<<http://www.mathworks.com/products/demos/simulink/physbe>>.
25. Li X. Computer simulation of the baroregulation in response to moderate dynamic exercise/ Li X., Bai J.// Medical & Biological Engineering & Computing, 2001, Vol. 49, p. 480 – 487.
26. Aboy M. A Novel Statistical Model for Simulation of Pressure Signals / Aboy M, McNames J, Thong T.// 17th Biennial International Eurasip Conference Biosignal Proceedings [Bruno University of Technology, 2004]. Vutium Press, 2004. p. 364–366.

27. Zvirioniene A. Aptarnavimo sistemos reiškimas Muro ir Milio automatų sąsūka dekartio sandauga / Zvirioniene A., Navickas Z., Vrubliauskas A.// Informacinės technologijos: tarptautinės konferencijos pranešimų medžiaga [Kaunas, 2005]. Kaunas, 2005. p. 14 – 18.
28. Zvirioniene A. Telecommunication Systems Analysis Using the Convolution of Moore and Mealy Automata / Zvirioniene A., Navickas Z., Rindzevicius R.// Elektronika ir elektrotechnika. ISSN 1392 – 1215. 2005 nr. 3(59) p. 64 – 70.
29. Shalyto A.A. Logic Control and "Reactive" Systems: Algorithmization and Programming. // Automation and Remote Control 2001; Vol.62, No.1: p.1–29.
30. Svozil K. Analogues of quantum complementarity in the theory of automata. Stud. Hist. Philos. Sci. Part B: Stud. Hist. Philos. Mod. Phys; University of Technology Vienna, Austria; 1998; 29(1); p. 61–80.
31. Pranevičius H. Kompiuterinių tinklų protokolų formalusis specifikavimas ir analizė: agregatinis metodas. Kaunas, 2003. 96 p. ISBN 9955 – 09 – 560 – 1.
32. Brauer V. The automate theory. Moscow; 1987. p. 275.
33. Blonskis J. Programavimo C++ Builder pavyzdžiai / Blonskis J., Bukšnaitis V., Jusas V., Marcinkevičius R., Smolinskas J. Kaunas, 2002. 291 psl. ISBN 9986 – 965 – 83 –7.
34. 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 psl.

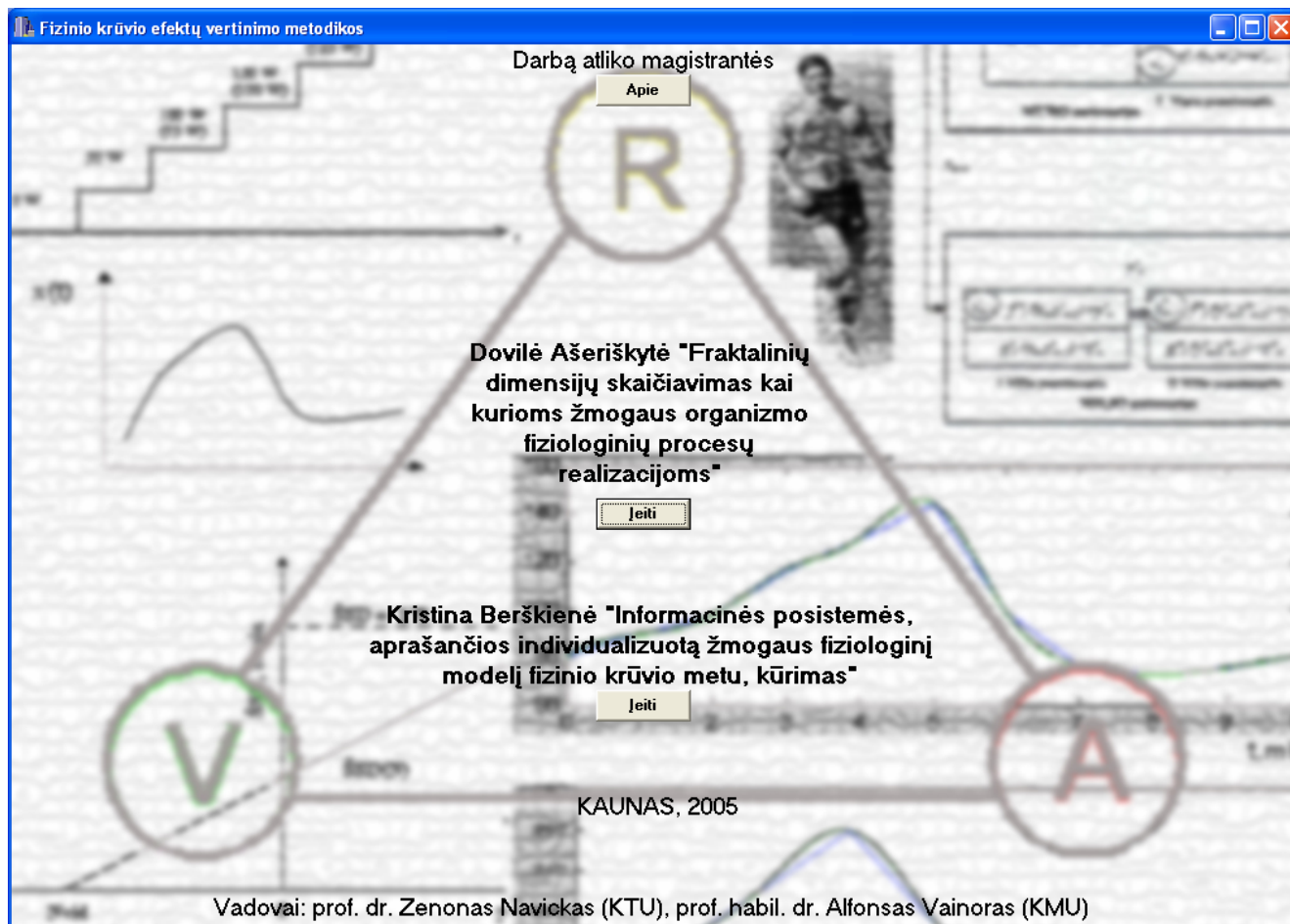
1 PRIEDAS

Integraliniai parametrai

Integralinių parametru vidurkiai ir standartiniai nuokrypiai tirtoms grupėms

Grupė	Nvid	Niš	Nvisas	n.k.	PWC170	PPI	PRI
	± σ	± σ	± σ	± σ	± σ	± σ	± σ
Sportininkės mėgėjos	2,28 ± 0,76	2,35 ± 0,34	5,12 ± 1,00	0,46 ± 0,05	2,59 ± 0,52	4,38 ± 0,98	0,84 ± 0,36
Profesionalūs sportininkai	2,28 ± 0,83	3,02 ± 0,68	5,31 ± 0,82	0,51 ± 0,08	3,08 ± 0,51	3,52 ± 0,75	1,21 ± 0,69
Profesionalės sportininkės	2,88 ± 1,02	2,88 ± 0,64	5,75 ± 1,44	0,51 ± 0,07	3,09 ± 0,82	3,84 ± 1,03	1,04 ± 0,46
20 – 30 metų nesportuojantys vyrai	2,79 ± 0,43	2,91 ± 0,62	5,71 ± 0,82	0,51 ± 0,06	2,96 ± 0,73	3,36 ± 0,76	1,21 ± 0,28
20 – 30 metų nesportuojančios moterys	2,13 ± 0,64	2,15 ± 0,39	4,28 ± 0,89	0,51 ± 0,08	2,89 ± 0,43	3,82 ± 1,04	0,87 ± 0,49
30 – 40 metų nesportuojantys vyrai	2,75 ± 0,80	2,64 ± 0,55	5,39 ± 0,88	0,49 ± 0,11	2,84 ± 0,61	3,83 ± 1,06	1,19 ± 0,44
30 – 40 metų nesportuojančios moterys	2,07 ± 0,58	1,91 ± 0,44	3,97 ± 0,85	0,48 ± 0,07	2,01 ± 0,56	4,36 ± 0,86	1,12 ± 0,87
40 – 50 metų nesportuojantys vyrai	3,02 ± 0,61	2,61 ± 0,5	5,63 ± 0,72	0,47 ± 0,07	3,21 ± 0,73	3,83 ± 0,94	1,45 ± 0,46
40 – 50 metų nesportuojančios moterys	2,01 ± 0,79	1,81 ± 0,47	3,82 ± 1,01	0,49 ± 0,12	1,97 ± 0,61	4,59 ± 1,26	1,27 ± 0,96
Išemine širdies liga sergantys vyrai	2,69 ± 1,2	1,8 ± 0,62	4,49 ± 1,15	0,41 ± 0,1	2,71 ± 1,49	5,21 ± 0,21	1,51 ± 0,74

Programos darbo langų pavyzdžiai



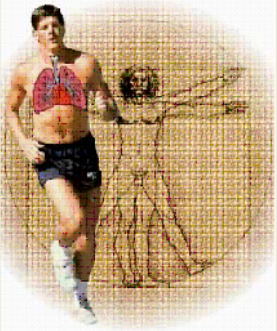
1 pav. Pradinis programos darbo langas

Individualizuoto žmogaus fiziologinio modelio tyrimas

Naujas Keisti Išvalyti Importuoti Eksportuoti Filtrai Grįžti

Ieškoti

Pavardė, vardas	Paciento numeris	Požymis	Lytis
► Abramavicius Paulius	1	s	1
Adomaitis Rimgaudas	215	n	1
Aleknaviciute Sigute	282	n	2
Aleksynas Algimantas	91	i	1
Alisauskas Vytenis	200	n	1
Almonaityte Vaida	256	n	2
Ambrasienė Juratė	319	n	2
Antolevicius Leonas	92	i	1
Atas Mindaugas	173	n	1
Augunas Andrius	68	s	1
Augustiniene Vilma	320	n	2
Avulis Bronius	93	i	1
Babenskas Vincentas	94	i	1
Bagdonaviciene Ruta	257	n	2
Bagdonavicius Vytautas	95	i	1
Balcietai Laimutis	96	i	1
Balciunaite Kornelija	2	s	2
Balezentis Vytautas	97	i	1



Naujas tyrimas Ištrinti tyrimą Analizuoti Palyginti Sv dinamika Grupės

Jrašo numeris	Tyrimas	Data	Amžius	Ūgis	Svoris	Sv	Žingsnis	Nmax	Požymis	HR_0	HR_50	HR_100	HR_150	HR_200	HR_250	HR_300	HR_1	
►	2	1	2003.06.26	17	202	90	69	50	250	s	81	90	98	117	130	145	-	9:
	3	2	2004.10.04	18	202	92	72	50	250	s	68	78	87	106	120	137	-	9:
	4	3	2004.11.17	18	202	94	73	50	250	s	68	82	94	106	127	139	-	8:

2 pav. Pagrindinis programos darbo langas

Individualizuoto žmogaus fiziologinio modelio tyrimas

Naujas Keisti Išvalyti Importuoti Eksportuoti Filtrai Grįžti

Ieškoti

Pavardė, vardas	Paciento numeris	Požymis	Lytis
▶ Abramavicius Paulius			
Adomaitis Rimgaudas			
Aleknaviciute Sigute			
Aleksynas Algimantas			
Alisauskas Vytenis			
Almonaityte Vaida			
Ambrasiene Jurate			
Antolevicius Leonas			
Atas Mindaugas			
Augunas Andrius			
Augustiniene Vilma			
Avulis Bronius			
Babenskas Vincentas			
Bagdonaviciene Ruta			
Bagdonavicius Vytautas			
Balciunaitė Kornelija			
Balciunaitė Kornelija			
Balezentis Vytautas			

Naujo tyrimo duomenų įvedimas

Įveskite šiuos naujo tyrimo duomenis:

Tyrimo data (yyyy.mm.dd) 2005.05.30

Paciento amžius 25

Ūgis, cm 182

Svoris, kg 75

Laiptelio dydis, W 50

Maksimalus galingumas, W 200

Atsigavimo laikas 5 min

Grupė prof. sportas

Skaičiuoti S

	Fizinis krūvis					Atsigavimas				
	0 W	50 W	100	150	200	1 min	2 min	3 min	4 min	5 min
SSD, k/min	123	90								
S, mm/Hg										
D, mm/Hg										
JT, s										
ST, s										
QRS, s										

Įvesti Baigti

Naujas tyrimas	I
▶	
2	
3	
4	

Įrašo numeris	T
▶	
2	
3	
4	

HR_300	HR_1
-	9:
-	9:
-	8:

3 pav. Naujo tyrimo duomenų įvedimo langas

Paciento tyrimo duomenų analizė

Paciento pavardė, vardas: **Abramavicius Paulius** Tyrimo nr. **1** <- Tyrimas Tyrimas ->
 Amžius metais: **17** Grupė: **prof. sportas** <- Pacientas Pacientas -> Grįžti
 Tyrimo data: **2003.06.26**

Duomenys | Parametrų grafikai | ŠSD(N) priklausomybė | S ir D analizė | Sv analizė | Parametrų kombinacijos

Pradiniai duomenys:

Krūvio etapas

Parametras	0 W	50 W	100 W	150 W	200 W	250 W	300 W
ŠSD, k/min	81	90	98	117	130	145	-
S, mmHg	114	124	140	158	198	210	-
D, mmHg	70	60	40	40	10	10	-
JT, s	3.25	3.18	2.99	2.75	2.54	2.40	-

Atsigavimo etapas

Parametras	1 min	2 min	3 min	4 min	5 min
ŠSD, k/min	99	77	73	76	81
S, mmHg	190	162	150	118	120
D, mmHg	10	10	30	60	58
JT, s	2.68	3.06	3.34	3.32	3.39

Išvestiniai duomenys:

Krūvio etapas

Parametras	0 W	50 W	100 W	150 W	200 W	250 W	300 W
RR	0.74	0.67	0.61	0.51	0.46	0.41	-
JT/RR	4.39	4.77	4.88	5.36	5.50	5.80	-
(S-D)	44	64	100	118	188	200	-
(S-D)/S	0.39	0.52	0.71	0.75	0.95	0.95	-
DS	92.34	111.60	137.20	184.86	257.40	304.50	-

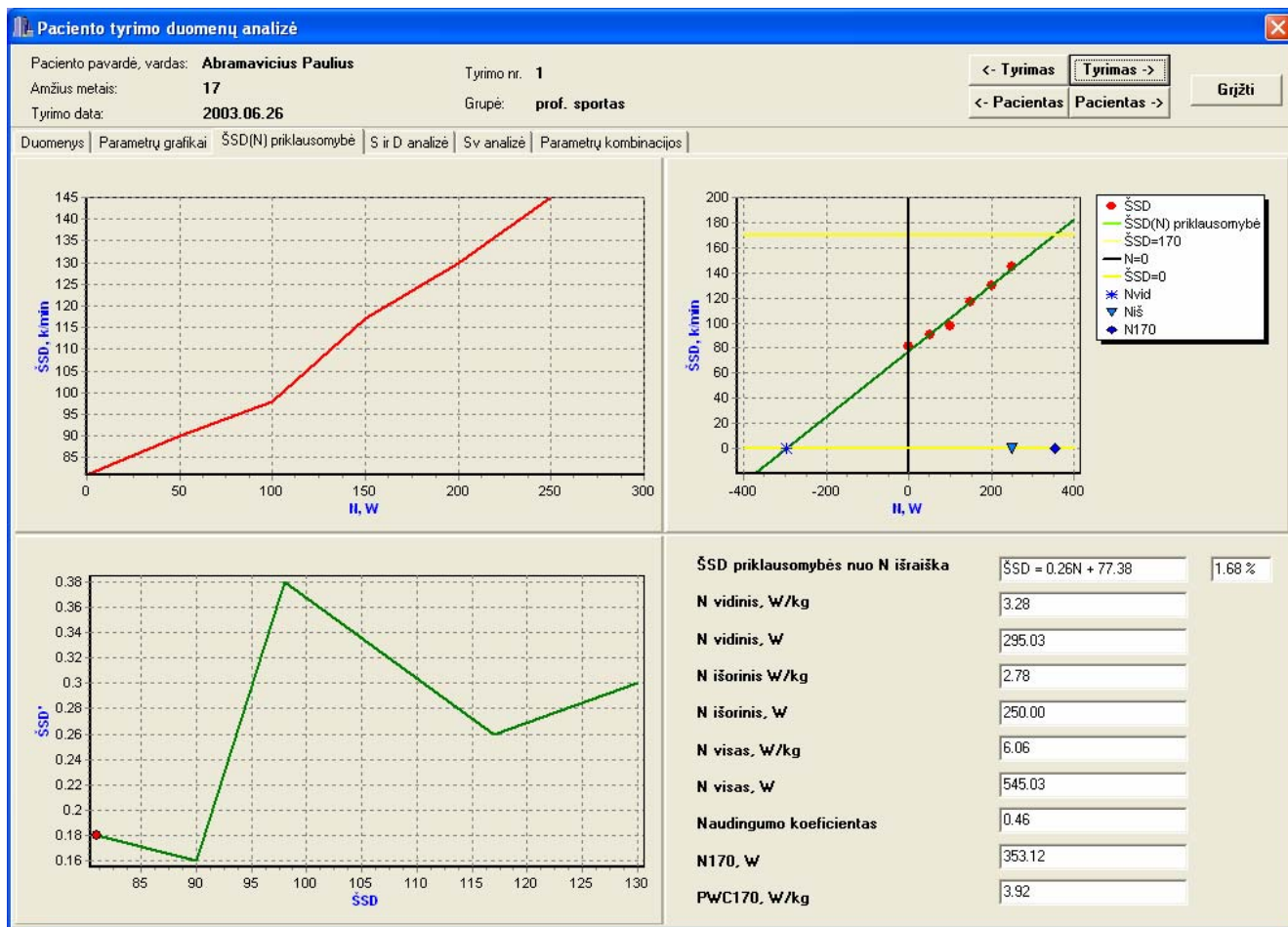
Atsigavimo etapas

Parametras	1 min	2 min	3 min	4 min	5 min
RR	0.61	0.78	0.82	0.79	0.74
JT/RR	4.42	3.93	4.06	4.21	4.58
(S-D)	180	152	120	58	62
(S-D)/S	0.95	0.94	0.80	0.49	0.52
DS	188.10	124.74	109.50	89.68	97.20

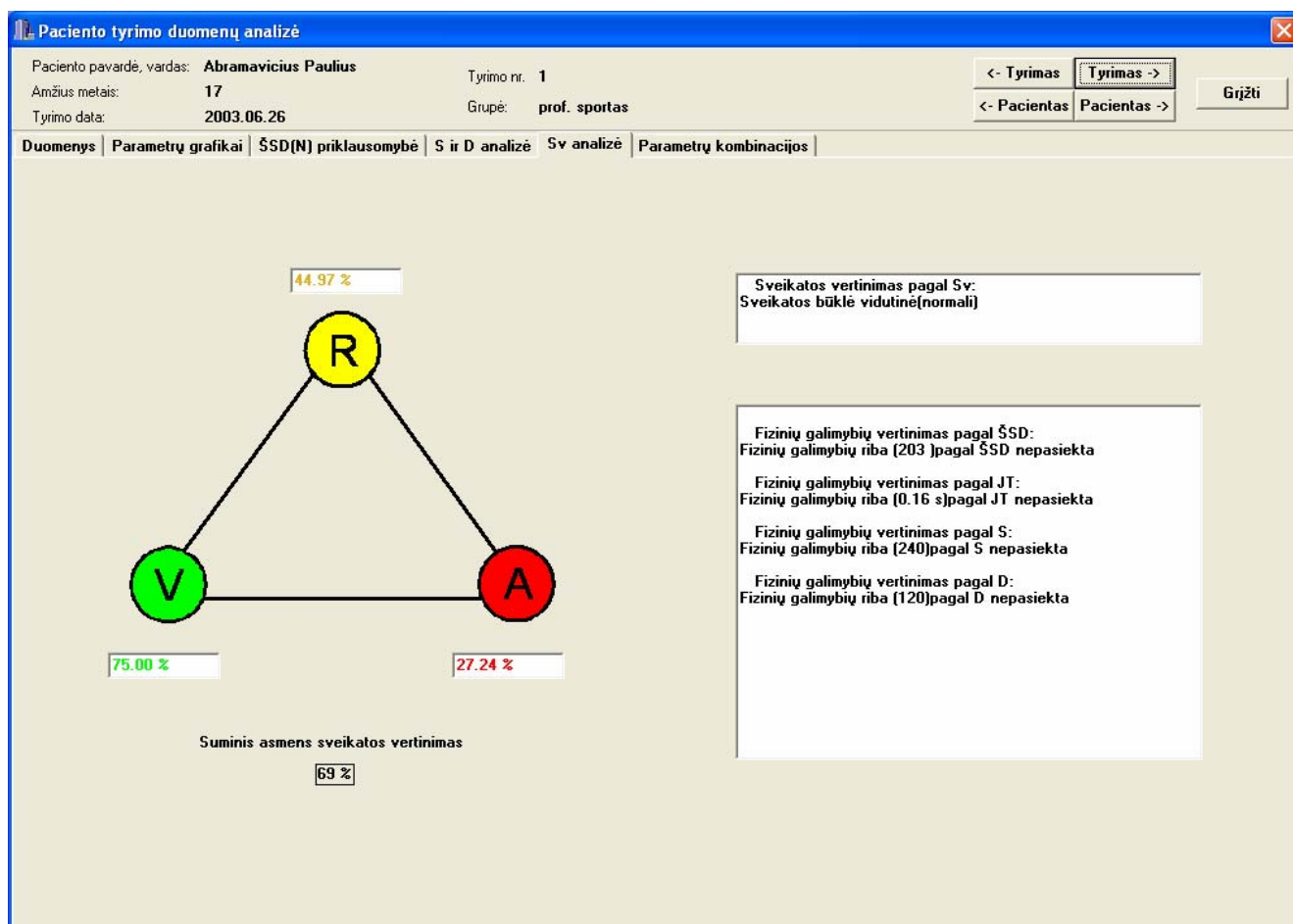
Integralūs rodikliai

Sv, %	Nvid, W/kg	Nis, W/kg	Nvis, W/kg	N.k.	N170 W	PWC170 W	PPI mmHg/k	PRI mmHg/k	SD corr	K	L	Pak.IHR(N)	M	N	A
69	3.28	2.78	6.06	0.46	353.12	3.92	3.84	1.59	-0.93	0.26	77.38	0.02	-275.14	986.00	19

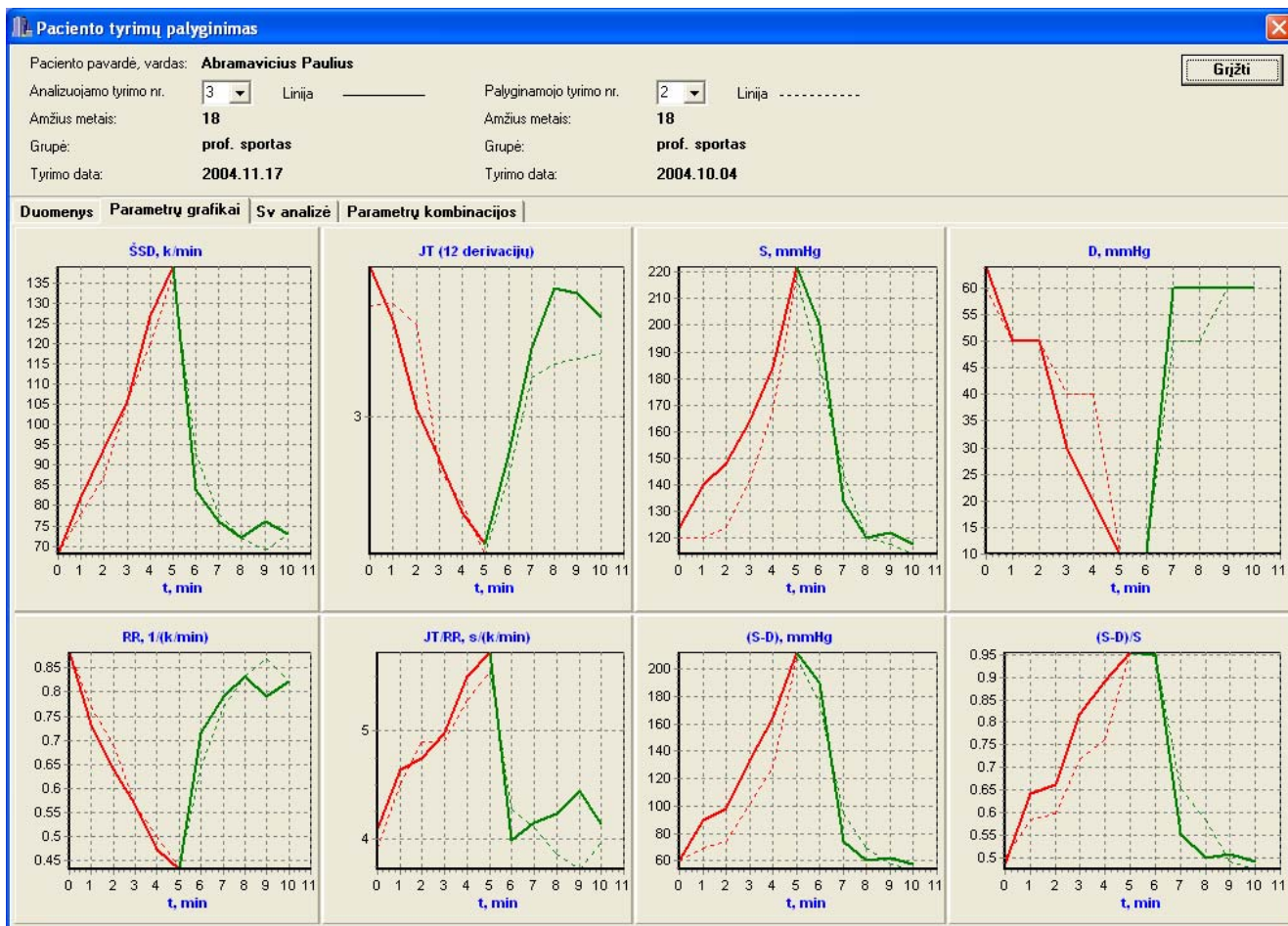
4 pav. Milio automato išėjimo signalo iliustracinis langas



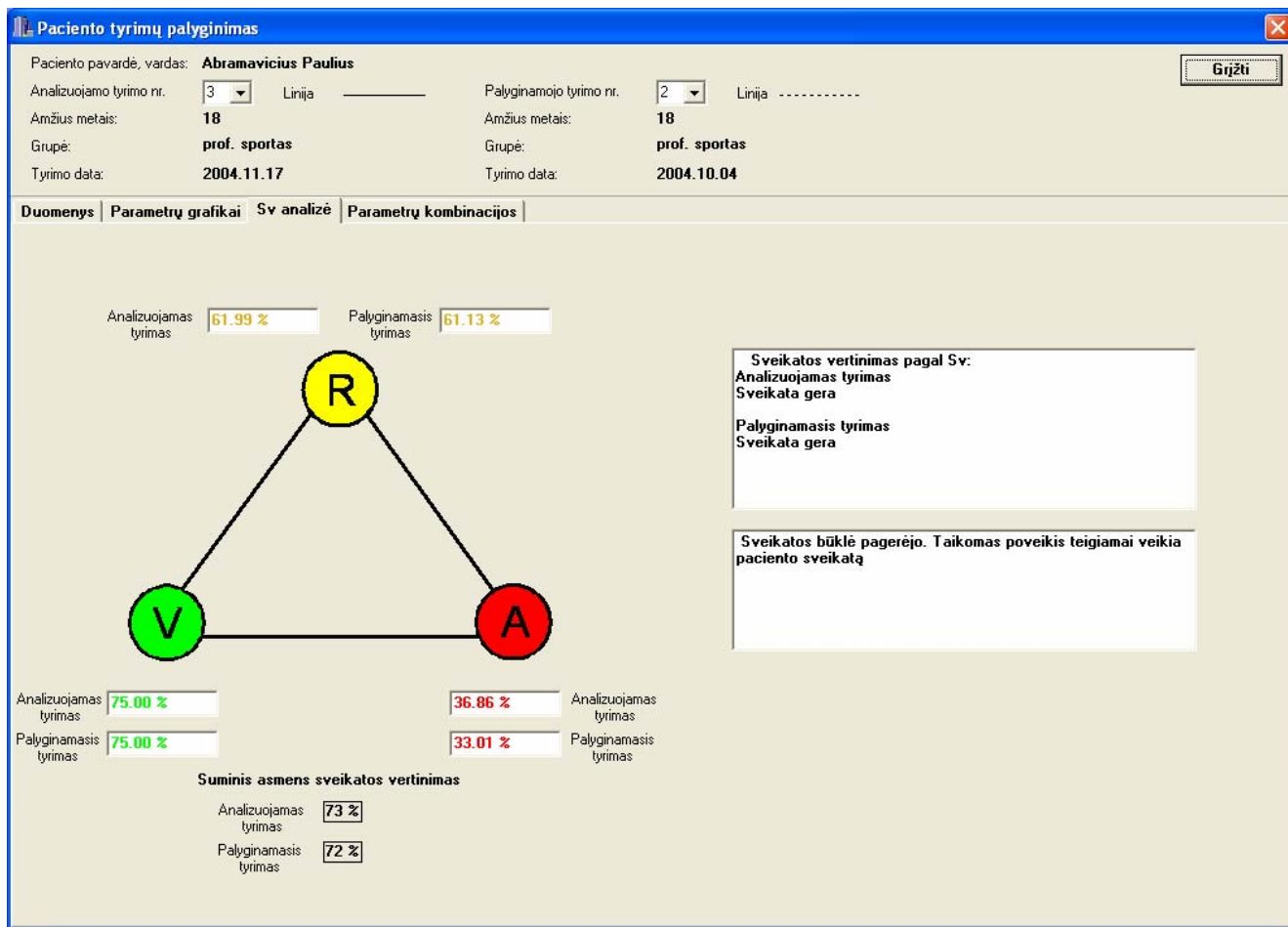
5 pav. ŠSD(N) priklausomybės analizės langas



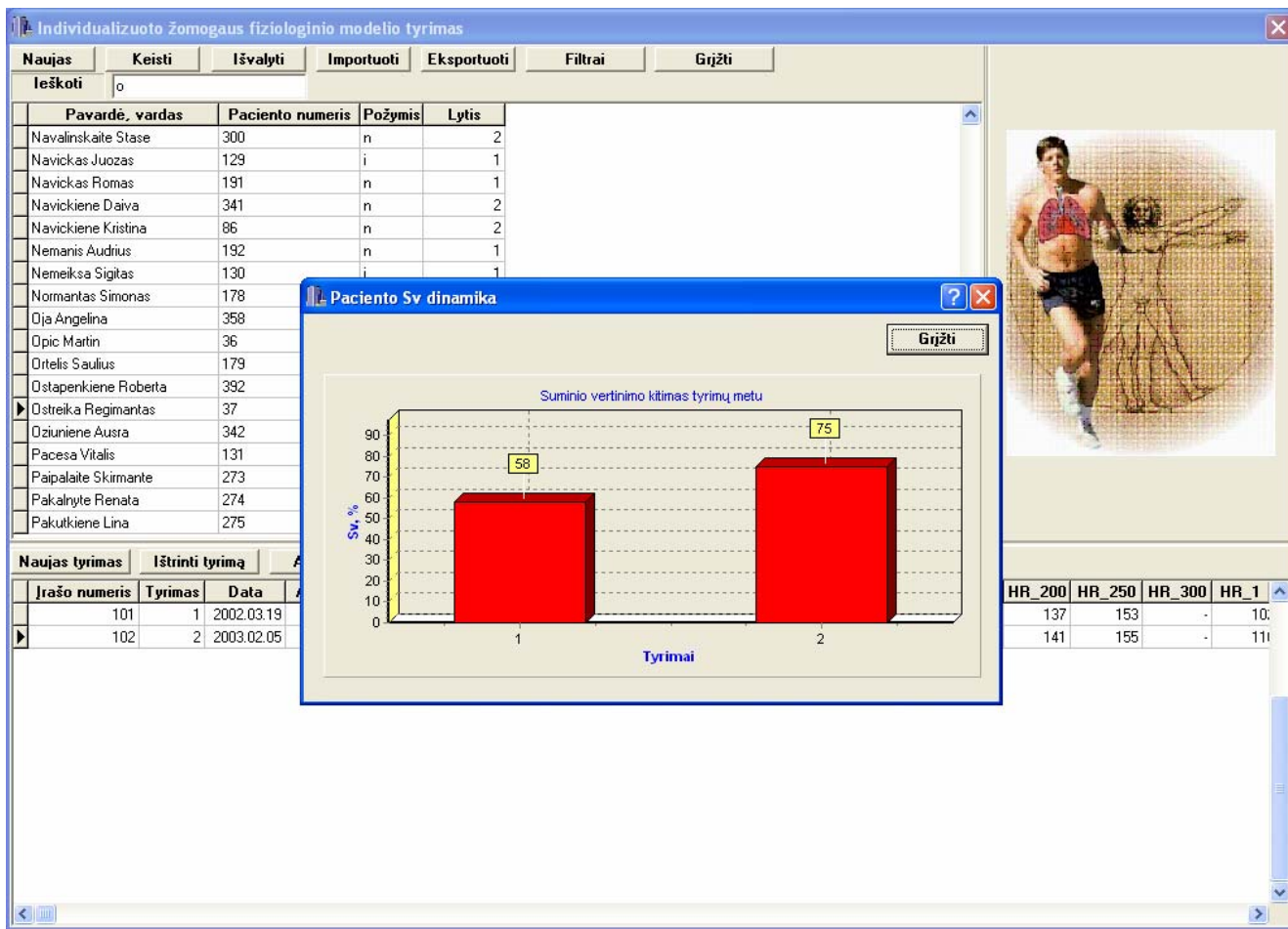
6 pav. Virtualaus gydytojo sprendimo langas



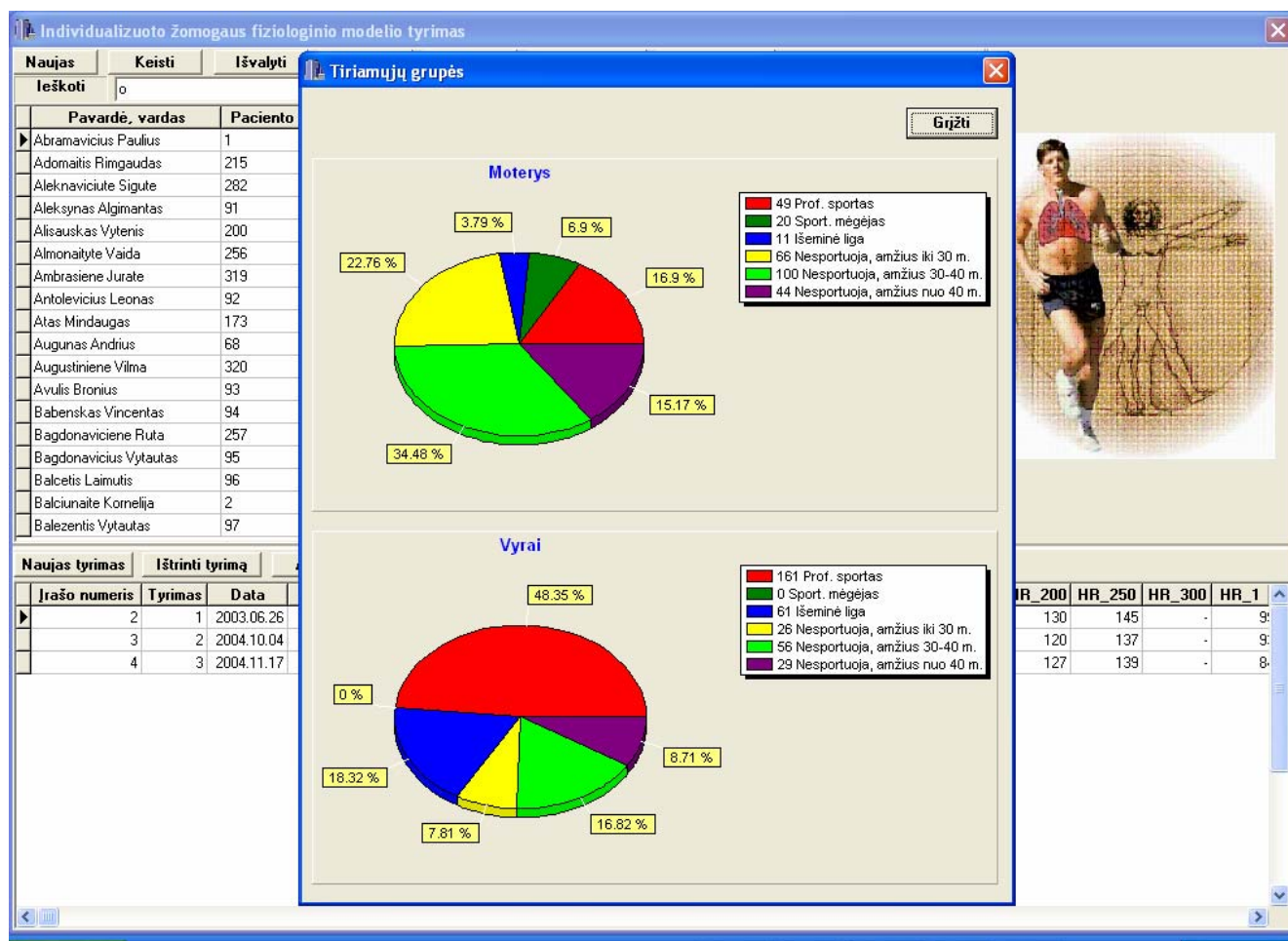
7 pav. Dviejų tyrimų duomenų palyginimas



8 pav. Sveikatos būklės vertinimo palyginimas



9 pav. Sv dinamika tyrimų metu



10 pav. Programos duomenų grupės

Programos teksto fragmentai

Milio ir Muro automatų sąsūakai aprašyti naudojami duomenų tipai:

```

struct Tyrimas
{
    int amzius;
    int svoris;
    int ugis;
    int Nmax;
    int zing;
    int sk;
    int tyrimas;
    AnsiString poz;

    int HRk[ik];
    int HRA[ia];

    int Sk[ik];
    int Sa[ia];

    int Dk[ik];
    int Da[ia];

    float JTk[ik];
    float JTa[ia];

    float RRk[ik];
    float RRA[ia];

    float JTRRk[ik];
    float JTRRa[ia];

    int SDk[ik];
    int SDa[ia];

    float SDSk[ik];
    float SDSa[ia];

    float DSk[ik];
    float DSA[ia];
};

struct Tyrimasi
{
    int Nmaxi;
    int zingi;
    int ski;

    float HRki[ik-1];
    float Ski[ik-1];
    float Dki[ik-1];
    float JTKi[ik-1];
    float RRki[ik-1];
    float JTRRki[ik-1];
    float SDki[ik-1];
    float SDSki[ik-1];
    float DSKI[ik-1];
};

struct TyrimasPr
{
    int amziusp;
    int svorisp;
    int ugisp;
    int Nmaxp;
    int zingp;
    int skp;
    int tyrimasp;
    AnsiString pozp;
}

```

```

int HRkp[ik];
int HRap[ia];

int Skp[ik];
int Sap[ia];

int Dkp[ik];
int Dap[ia];

float JTkp[ik];
float JTap[ia];

};

struct Compl
{ float drr;
  float djt;
  float dN;
  int tyr;
  int Sv;
  float Nvid;
  float Nis;
  float Nvis;
  float Nk;
  float PWC170;
  float PPI;
  float PRI;
  float SDcorr;
  float HR_N;
  float DS_JT;
  float S_P;
  float N170;
  float K,L, M, N, A, B;
};

    Tyrimas *Tm;
    Tyrimas *T;
    Tyrimasi *Ti;
    Compl *C;
    TyrimasPr *Tpr;

```

Milio ir Muro automatų sąšūka

```

#include <vcl.h>
#include <IniFiles.hpp>
#pragma hdrstop

#include "PagK.h"
#include "NaujK.h"
#include "AnalK.h"
#include "PalK.h"
#include "Npacientas.h"
#include "DBDM.h"
#include "Config.h"
#include "Slaptazodis.h"

#include "Header.h"
#include <stdio.h>
#include <string.h>
#include <math.h>
#include <dir.h>
#include <FileCtrl.hpp>

//-----
#pragma package(smart_init)
#pragma link "Excel_2K_SRV"
#pragma resource "*.dfm"
TPagK1 *PagK1;
const SELDIRHELP = 1000;
//-----
void __fastcall TPagK1::Button1Click(TObject *Sender)
{
    Pacientas = new TPacientas(this);
    Pacientas->Button1->Tag = 1;
    Pacientas->Tag = PagK1->Tag;
}

```

```

    Pacientas->ShowModal();
    delete Pacientas;
    Pacientas = NULL;
}
//-----

void __fastcall TPagK1::Button2Click(TObject *Sender)
{
    if ( DBDM1->PacientaiT->RecordCount > 0){
        Pacientas = new TPacientas(this);
        Pacientas->Button1->Tag = 2;
        Pacientas->ShowModal();
        delete Pacientas;
        Pacientas = NULL;
    }
}
//-----

AnsiString TPagK1:: PatientNr()
{
    AnsiString Nr = DBDM1->PacientaiT->FieldByName("PacientoNumeris")->AsString;
    return Nr;
}

void __fastcall TPagK1::Edit1Change(TObject *Sender)
{
    if ( Edit1->Text != ""){
        TLocateOptions Opts;
        Opts.Clear();
        Opts << loCaseInsensitive << loPartialKey;
        if ( DBDM1->PacientaiT->Locate( "PavVard",Edit1->Text, Opts)) { }
    } else DBDM1->PacientaiT->First();
}
//-----

void __fastcall TPagK1::FormActivate(TObject *Sender)
{
    if ( pirmas ){
        TIniFile *ini;
        /* ini = new TIniFile(ExtractFilePath(Application->ExeName)+"Admin.INI" );
        if ( ini->ReadInteger("Login", "Login", 1) == 1){
            login->Visible = true;
            database->Visible = false;
            DB->Caption = "Apie...";
            PasswordDlg->ShowModal();
            if (PasswordDlg->Tag) { login->Visible = false; database->Visible = true;}
            else Close(); }
        else if ( ini->ReadInteger("Login", "Login", 1) == 2){
            login->Visible = false; database->Visible = true;
            DB->Tag = DBDM1->GydytojaiT->FieldByName("GydytojoNumeris")->AsInteger;
        }
        else Close();
        delete ini;
        pirmas = false;
        // PagK1->Caption = "Duomenu baze";

        ini = new TIniFile(ExtractFilePath(Application->ExeName)+DBDM1->GydytojaiT-
        >FieldByName("GydytojoNumeris")->AsString + ".INI" );
        P->Height = ini->ReadInteger("Panels", "Height", 200 );
        PPacientai->Width = ini->ReadInteger("Panels", "Width", 300);

        DBGrid1->Columns->Items[0]->Width = ini->ReadInteger ("db", "1C1", 150 );
        DBGrid1->Columns->Items[1]->Width = ini->ReadInteger ("db", "1C2", 90 );

        DBGrid2->Columns->Items[0]->Width = ini->ReadInteger ("db", "2C1", 200 );
        DBGrid2->Columns->Items[1]->Width = ini->ReadInteger ("db", "2C2", 150 );

        delete ini;
    }
}
//-----

void __fastcall TPagK1::FormKeyDown(TObject *Sender, WORD &Key,
    TShiftState Shift)
{
    if (database->Visible){
        if ( Key == 67 ) Config1->ShowModal();
    }
}
//-----

void __fastcall TPagK1::Button4Click(TObject *Sender)
{
    NaujK1 = new TNaujK1(this);
    NaujK1->ShowModal();
}

```

```

    delete NaujK1;
    NaujK1= NULL;
}
//-----

void __fastcall TPagK1::Button5Click(TObject *Sender)
{
    if (Application->MessageBox ( "Ar tikrai norite ištrinti šio tyrimo duomenis?"
    ,"Ispėjimas",MB_YESNO+MB_ICONINFORMATION+MB_DEFBUTTON1)==IDYES) {
        DBDM1->IrasaiT->Delete();
    }
}
//-----
/*void __fastcall TPagK1::Button8Click(TObject *Sender)
{
    int kiek = DBDM1->PacientaiT->RecordCount;
    DBDM1->PacientaiT->First();
    for (int i=0; i<kiek; i++) {
        int kiekis = DBDM1->IrasaiT->RecordCount;
        DBDM1->IrasaiT->First();
        for (int j=0; j<kiekis; j++) {
            DBDM1->IrasaiT->Edit();
            DBDM1->IrasaiT->FieldName("Pozymis")->AsString = DBDM1->PacientaiT-
>FieldName("Pozymis")->AsString;
            DBDM1->IrasaiT->Post();
            DBDM1->IrasaiT->Next();
        }
        DBDM1->PacientaiT->Next();
    }
}
//-----

TyrimasPr* TPagK1:: Duomenys (TObject *Sender)
{
    TyrimasPr *A;

    A=new TyrimasPr;
    if (A==NULL) { ShowMessage ("Trūksta atminties"); return NULL; }
    else {
        A->tyrimasp=DBDM1->IrasaiT->FieldName("Tyrimas")->AsInteger;
        A->amziusp=DBDM1->IrasaiT->FieldName("Amzius")->AsInteger;
        A->svoris=DBDM1->IrasaiT->FieldName("Svoris")->AsInteger;
        A->ugisp=DBDM1->IrasaiT->FieldName("Ugis")->AsInteger;
        A->Nmaxp=DBDM1->IrasaiT->FieldName("Nmax")->AsInteger;
        A->zingp=DBDM1->IrasaiT->FieldName("L_dydis")->AsInteger;
        if (A->Nmaxp<=300){
            if (A->zingp==25)
                A->skp=A->Nmaxp/A->zingp;
            else
                A->skp=A->Nmaxp/A->zingp+1; }
        else
            A->skp=7;

        A->pozp=DBDM1->IrasaiT->FieldName("Pozymis")->AsString;

        A->HRkp[0]=DBDM1->IrasaiT->FieldName("HR_0")->AsInteger;
        A->HRkp[1]=DBDM1->IrasaiT->FieldName("HR_50")->AsInteger;
        A->HRkp[2]=DBDM1->IrasaiT->FieldName("HR_100")->AsInteger;
        A->HRkp[3]=DBDM1->IrasaiT->FieldName("HR_150")->AsInteger;
        A->HRkp[4]=DBDM1->IrasaiT->FieldName("HR_200")->AsInteger;
        A->HRkp[5]=DBDM1->IrasaiT->FieldName("HR_250")->AsInteger;
        A->HRkp[6]=DBDM1->IrasaiT->FieldName("HR_300")->AsInteger;
        A->HRap[0]=DBDM1->IrasaiT->FieldName("HR_1")->AsInteger;
        A->HRap[1]=DBDM1->IrasaiT->FieldName("HR_2")->AsInteger;
        A->HRap[2]=DBDM1->IrasaiT->FieldName("HR_3")->AsInteger;
        A->HRap[3]=DBDM1->IrasaiT->FieldName("HR_4")->AsInteger;
        A->HRap[4]=DBDM1->IrasaiT->FieldName("HR_5")->AsInteger;

        A->Skp[0]=DBDM1->IrasaiT->FieldName("S_0")->AsInteger;
        A->Skp[1]=DBDM1->IrasaiT->FieldName("S_50")->AsInteger;
        A->Skp[2]=DBDM1->IrasaiT->FieldName("S_100")->AsInteger;
        A->Skp[3]=DBDM1->IrasaiT->FieldName("S_150")->AsInteger;
        A->Skp[4]=DBDM1->IrasaiT->FieldName("S_200")->AsInteger;
        A->Skp[5]=DBDM1->IrasaiT->FieldName("S_250")->AsInteger;
        A->Skp[6]=DBDM1->IrasaiT->FieldName("S_300")->AsInteger;
        A->Sap[0]=DBDM1->IrasaiT->FieldName("S_1")->AsInteger;
        A->Sap[1]=DBDM1->IrasaiT->FieldName("S_2")->AsInteger;
        A->Sap[2]=DBDM1->IrasaiT->FieldName("S_3")->AsInteger;
    }
}

```

```

A->Sap[3]=DBDM1->IrasaiT->FieldByName("S_4")->AsInteger;
A->Sap[4]=DBDM1->IrasaiT->FieldByName("S_5")->AsInteger;

A->Dkp[0]=DBDM1->IrasaiT->FieldByName("D_0")->AsInteger;
A->Dkp[1]=DBDM1->IrasaiT->FieldByName("D_50")->AsInteger;
A->Dkp[2]=DBDM1->IrasaiT->FieldByName("D_100")->AsInteger;
A->Dkp[3]=DBDM1->IrasaiT->FieldByName("D_150")->AsInteger;
A->Dkp[4]=DBDM1->IrasaiT->FieldByName("D_200")->AsInteger;
A->Dkp[5]=DBDM1->IrasaiT->FieldByName("D_250")->AsInteger;
A->Dkp[6]=DBDM1->IrasaiT->FieldByName("D_300")->AsInteger;
A->Dap[0]=DBDM1->IrasaiT->FieldByName("D_1")->AsInteger;
A->Dap[1]=DBDM1->IrasaiT->FieldByName("D_2")->AsInteger;
A->Dap[2]=DBDM1->IrasaiT->FieldByName("D_3")->AsInteger;
A->Dap[3]=DBDM1->IrasaiT->FieldByName("D_4")->AsInteger;
A->Dap[4]=DBDM1->IrasaiT->FieldByName("D_5")->AsInteger;

A->JTkp[0]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_0")->AsString);
A->JTkp[1]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_50")->AsString);
A->JTkp[2]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_100")->AsString);
A->JTkp[3]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_150")->AsString);
A->JTkp[4]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_200")->AsString);
A->JTkp[5]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_250")->AsString);
A->JTkp[6]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_300")->AsString);
A->JTap[0]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_1")->AsString);
A->JTap[1]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_2")->AsString);
A->JTap[2]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_3")->AsString);
A->JTap[3]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_4")->AsString);
A->JTap[4]=StrToFloat(DBDM1->IrasaiT->FieldByName("JT_5")->AsString);

return A;
}
}
//-----
Tyrimas* TPagK1:: DuomenysPilni(TObject *Sender, TyrimasPr * A )
{
int ii;
Tyrimas *B;
if (A!=NULL){
B=new Tyrimas;
if (B==NULL) { ShowMessage ("Trūksta atminties"); return NULL; }
else {
B->tyrimas=A->tyrimasp;
B->amzius=A->amziusp;
B->svoris=A->svoris;
B->ugis=A->ugisp;
B->Nmax=A->Nmaxp;
B->zing=A->zingp;
B->sk=A->skp;
B->poz=A->pozp;

B->HRk[0]=A->HRkp[0];
B->HRk[1]=A->HRkp[1];
B->HRk[2]=A->HRkp[2];
B->HRk[3]=A->HRkp[3];
B->HRk[4]=A->HRkp[4];
B->HRk[5]=A->HRkp[5];
B->HRk[6]=A->HRkp[6];
B->HRa[0]=A->HRap[0];
B->HRa[1]=A->HRap[1];
B->HRa[2]=A->HRap[2];
B->HRa[3]=A->HRap[3];
B->HRa[4]=A->HRap[4];

B->Sk[0]=A->Skp[0];
B->Sk[1]=A->Skp[1];
B->Sk[2]=A->Skp[2];
B->Sk[3]=A->Skp[3];
B->Sk[4]=A->Skp[4];
B->Sk[5]=A->Skp[5];
B->Sk[6]=A->Skp[6];
B->Sa[0]=A->Sap[0];
B->Sa[1]=A->Sap[1];
B->Sa[2]=A->Sap[2];
B->Sa[3]=A->Sap[3];
B->Sa[4]=A->Sap[4];

B->Dk[0]=A->Dkp[0];

```



```

B->Dk[1]=A->Dkp[1];
B->Dk[2]=A->Dkp[2];
B->Dk[3]=A->Dkp[3];
B->Dk[4]=A->Dkp[4];
B->Dk[5]=A->Dkp[5];
B->Dk[6]=A->Dkp[6];
B->Da[0]=A->Dap[0];
B->Da[1]=A->Dap[1];
B->Da[2]=A->Dap[2];
B->Da[3]=A->Dap[3];
B->Da[4]=A->Dap[4];

B->JTk[0]=A->JTkp[0];
B->JTk[1]=A->JTkp[1];
B->JTk[2]=A->JTkp[2];
B->JTk[3]=A->JTkp[3];
B->JTk[4]=A->JTkp[4];
B->JTk[5]=A->JTkp[5];
B->JTk[6]=A->JTkp[6];
B->JTa[0]=A->JTap[0];
B->JTa[1]=A->JTap[1];
B->JTa[2]=A->JTap[2];
B->JTa[3]=A->JTap[3];
B->JTa[4]=A->JTap[4];

// RR //
for (ii=0;ii<7; ii++)
{
    if (A->HRkp[ii]!=999)
        B->RRk[ii]=(float) (60/float (A->HRkp[ii]));
    else B->RRk[ii]=999;
}
for (ii=0;ii<5; ii++) {
    if (A->HRap[ii]!=999)
        B->RRa[ii]=(float) (60/float (A->HRap[ii]));
    else B->RRa[ii]=999; }

// JT/RR //
for (ii=0;ii<7; ii++) {
    if (B->JTk[ii]!=999)
        B->JTRRk[ii]=(float) (B->JTk[ii]/B->RRk[ii]);
    else B->JTRRk[ii]=999; }
for (ii=0;ii<5; ii++) {
    if (B->JTa[ii]!=999)
        B->JTRRa[ii]=(float) (B->JTa[ii]/B->RRa[ii]);
    else B->JTRRa[ii]=999; }

// S-D //
for (ii=0;ii<7; ii++) {
    if (B->Sk[ii]!=999)
        B->SDk[ii]=B->Sk[ii]-B->Dk[ii];
    else B->SDk[ii]=999; }
for (ii=0;ii<5; ii++) {
    if (B->Sa[ii]!=999)
        B->SDa[ii]=B->Sa[ii]-B->Da[ii];
    else B->SDa[ii]=999; }

// (S-D)/S //
for (ii=0;ii<7; ii++) {
    if (B->SDk[ii]!=999)
        B->SDSk[ii]=(float) (float (B->SDk[ii])/float (B->Sk[ii]));
    else B->SDSk[ii]=999; }
for (ii=0;ii<5; ii++) {
    if (B->SDa[ii]!=999)
        B->SDSa[ii]=(float) (float (B->SDa[ii])/float (B->Sa[ii]));
    else B->SDSa[ii]=999; }

// DS //
for (ii=0;ii<7; ii++) {
    if (B->Sk[ii]!=999)
        B->DSk[ii]=(float) B->Sk[ii]* (float) B->HRk[ii]*0.01;
    else B->DSk[ii]=999; }
for (ii=0;ii<5; ii++) {
    if (B->Sa[ii]!=999)
        B->DSa[ii]=(float) B->Sa[ii]* (float) B->HRa[ii]*0.01;
    else B->DSa[ii]=999; }

```

```

        return B;
    }
    } // jei A netucia
}

//-----
void TPagK1:: Skaiciavimai(TObject *Sender) {
    Tpr=Duomenys(Sender);
    T=DuomenysPilni(Sender,Tpr );
    C=DuomenysCompl(Sender,T);
    Ti=DuomenysI(Sender,T );
}
//-----
void __fastcall TPagK1::Button7Click(TObject *Sender)
{
    AnalK1 = new TAnalK1(this);
    AnalK1->ShowModal();
    delete AnalK1;
    AnalK1= NULL;
}
//-----
void TPagK1:: KL(Tyrimas *T, float *k, float *l){
    int sn=0;
    int shr=0;
    int shrn=0;
    int sns=0;

    for (int j=0; j<T->sk; j++) {
        if (T->zing==25) {
            sn=sn+Max_M[j];
            sns=sns+ Max_M[j]*Max_M[j];}
        else {
            sn=sn+Max_V[j];
            sns=sns+Max_V[j]*Max_V[j]; }
    }
    for (int j=0; j<T->sk; j++) {
        shr=shr+T->HRk[j];
    }
    for (int j=0; j<T->sk; j++) {
        if (T->zing==25)
            shrn=shrn+T->HRk[j]*Max_M[j];
        else
            shrn=shrn+T->HRk[j]*Max_V[j];
    }
    *k= float(sn*shr-(T->sk)*shrn)/float(sn*sns- (T->sk)*sns);
    *l= float(shrn*sn-shr*sns)/float(sn*sn- (T->sk)*sns);
}
//-----
Compl* TPagK1:: DuomenysCompl(TObject *Sender, Tyrimas *T ) {
    Compl *B;
    if (T!=NULL){
        B=new Compl;
        if (B==NULL) { ShowMessage ("Atminties klaida"); return NULL; }
        else {
            B->tyr=T->tyrimas;
            KL(T, &k, &l);
            B->K=k;
            B->L=l;
            B->Nvid=(B->L/B->K)/(T->svoris);
            B->Nis=float(T->Nmax)/float(T->svoris);
            B->N170=(170-B->L)/B->K;
            B->Nvis=(B->Nvid+B->Nis);
            B->Nk=B->Nis/B->Nvis;
            B->PWCl70=B->N170/float(T->svoris);
            PPI_PRI(T, &ppi, &pri);
            B->PPI=ppi;
            B->PRI=pri;
            SDcorr(T, &sdcorr);
            B->SDcorr=sdcorr;
            Sv(T, &sv, &dr2, &djt2, &dN2);
            B->dr2=dr2;
            B->djt=djt2;
            B->dN=dN2;
            B->Sv=DBDM1->IrasaiT->FieldByName("Sv")->AsInteger;
            MN(T, &m, &n);
            B->M=m;

```

```

    B->N=n;
    AB(T, &a, &b);
    B->A=a;
    B->B=b;
    //-----paklaidos-----
    float sp=0;
    float p=0;
    for (int i=0; i<T->sk; i++) {
        if (T->zing==25)
            p=float(abs(T->HRk[i]-(B->K*Max_M[i]+B->L)))/float(B->K*Max_M[i]+B->L);
        else
            p=float(abs(T->HRk[i]-(B->K*Max_V[i]+B->L)))/float(B->K*Max_V[i]+B->L);
        sp=sp+p;
    }
    B->HR_N=sp/T->sk;

    float spp=0;
    if (B->M*T->JTk[0]-B->N!=0)
        spp=float(abs(T->DSk[0] - (B->M*T->JTk[0]-B->N)))/float(B->M*T->JTk[0]-B->N);
    else spp=0;
    if (B->M*T->JTk[1]-B->N!=0)
        spp=spp+float(abs(T->DSk[1] - (B->M*T->JTk[1]-B->N)))/float(B->M*T->JTk[1]+B->N);
    for (int i=1; i<T->sk; i++)
        spp=spp+(float(abs(T->DSk[i]- ((B->A)/T->JTk[i]+B->B)))/(float((B->A)/T->JTk[i]+B->B)));

    B->DS_JT=spp/float(T->sk+1);

    //-----

    return B; }
}
//-----
void TPagK1:: PPI_PRI(Tyrimas *T, float *ppi, float *pri){
    float ss=0;
    //float sn;
    float shr=0;
    float n=0;

    for (int i=0; (i<T->sk & T->Sk[i+1]!=999); i++){
        if ((T->Sk[i+1]-T->Sk[i])>0) {
            n++;
            if (T->zing==25)
                ss=ss+float(T->Sk[i+1]-T->Sk[i])/float(Max_M[i+1]-Max_M[i]);
            else
                ss=ss+float(T->Sk[i+1]-T->Sk[i])/float(Max_V[i+1]-Max_V[i]);
        }
        if ((T->HRk[i+1]-T->HRk[i])>0) & (T->Sk[i+1]-T->Sk[i])>0)
            shr=shr+float(T->Sk[i+1]-T->Sk[i])/float(T->HRk[i+1]-T->HRk[i]);
    }

    if (ss!=0 & n!=0)
        *ppi=(ss/n)*10;
    else *ppi=0;
    if (shr!=0 & n!=0)
        *pri=(shr/n);
    else *pri=0;
}
//-----
void TPagK1:: SDcorr(Tyrimas *T, float *sdcorr){
    int ss=0;
    int sd=0;
    int n=0;
    float vs;
    float vd;
    int sansd=0;
    float vsansd;
    int s2=0;
    int d2=0;
    float vs2;
    float vd2;

    for (int i=0; i<T->sk; i++)
        {n++;
        ss=ss+T->Sk[i];

```

```

        sd=sd+T->Dk[i];
    }
    for (int i=0; (i<5 & T->Sa[i]!=999); i++)
    { n++;
      ss=ss+T->Sa[i];
      sd=sd+T->Da[i];
    }
    vs=float(ss)/float(n);
    vd=float(sd)/float(n);

    for (int i=0; i<T->sk; i++)
        sansd=sansd+T->Sk[i]*T->Dk[i];
    for (int i=0; (i<5 & T->Sa[i]!=999); i++)
        sansd=sansd+T->Sa[i]*T->Da[i];
    vsansd=float(sansd)/float(n);

    for (int i=0; i<T->sk; i++)
        s2=s2+T->Sk[i]*T->Sk[i];
    for (int i=0; (i<5 & T->Sa[i]!=999); i++)
        s2=s2+T->Sa[i]*T->Sa[i];
    vs2=float(s2)/float(n);

    for (int i=0; i<T->sk; i++)
        d2=d2+T->Dk[i]*T->Dk[i];
    for (int i=0; (i<5 & T->Da[i]!=999); i++)
        d2=d2+T->Da[i]*T->Da[i];
    vd2=float(d2)/float(n);

    if ((vs2-vs*vs)>0 & (vd2-vd*vd)>0)
        *sdcorr= (vsansd-vs*vd)/((sqrt(vs2-vs*vs))*(sqrt(vd2-vd*vd)));
}
//-----
void TPagK1:: Sv(Tyrimas *T, int *sv, float * drr2, float * djt2, float *dN2){
    float drr1;
    float djt1;

    float sn=0;
    float nn;

    drr1=((T->RRk[0])-(T->RRk[T->sk-1]))/0.727;
    djt1=((T->JTk[0]/12)-(T->JTk[T->sk-1]/12))/0.26;

    for (int i=0; (i<T->sk); i++){
        if (T->zing==25)
            sn=sn+Max_M[i];
        else
            sn=sn+Max_V[i];
    }
    int n=T->sk-1;

    if (T->zing==25)
        {nn=sn/(n*Nm);
        *sv=(1/mk)*sqrt(drr1*drr1+djt1*djt1+ nn*nn)*100;
        }
    else
        {nn=sn/(n*Nv);
        *sv=(1/vk)*sqrt(drr1*drr1+djt1*djt1+ nn*nn)*100; }

    *drr2=drr1;
    *djt2=djt1;
    *dN2=nn;
}
//-----
void TPagK1:: MN(Tyrimas *T, float *m, float *n){
    if ((T->JTk[1]-T->JTk[0])!=0) {
        *m=float (T->DSk[1]-T->DSk[0])/float (T->JTk[1]-T->JTk[0]);
        *n=float (T->JTk[0]*T->DSk[1]- T->JTk[1]*T->DSk[0])/float (T->JTk[1]-T->JTk[0]);
    }
    else {
        *m=0;
        *n=0; }
}
//-----
void TPagK1:: AB(Tyrimas *T, float *a, float *b){
    float sdsjt=0;

```

```

float sds=0;
float sjt=0;
float sjtjt=0;

for (int i=1; i<T->sk; i++) {
    sdsjt=sdsjt+ float (T->DSk[i]/(T->JTk[i]);
    sds=sds+T->DSk[i];
    sjt=sjt+1/T->JTk[i];
    sjtjt=sjtjt+1/(T->JTk[i]*T->JTk[i]);
}
float n=T->sk-1;
*a=(n*sdsjt-sds*sjt)/(n*sjtjt-sjt*sjt);
*b=(sjtjt*sds-sjt*sdsjt)/(n*sjtjt-sjt*sjt);
}
//-----

void __fastcall TPagK1::Button3Click(TObject *Sender)
{
if (Application->MessageBox ( "Ar tikrai norite ištrinti šio paciento duomenis?"
,"Ispėjimas",MB_YESNO+MB_ICONINFORMATION+MB_DEFBUTTON1)==IDYES) {
DBDM1->PacientaiT->Delete();
DBDM1->PacientaiT->Refresh();
DBDM1->PacientaiT->Close();
DBDM1->PacientaiT->Open(); }
}
//-----
Tyrimasi* TPagK1:: DuomenysI(TObject *Sender, Tyrimas * A )
{
int ii;
Tyrimasi *B;
if (A!=NULL){
B=new Tyrimasi;
if (B==NULL) { ShowMessage ("Trūksta atminties"); return NULL; }
else {

B->Nmaxi=A->Nmax;
B->zingi=A->zing;
B->ski=A->sk-1;

for (ii=0;ii<6; ii++)
{
if (A->HRk[ii+1]!=999)
{if (A->zing==25)
B->HRki[ii]=float (A->HRk[ii+1]-A->HRk[ii])/float (Max_M[ii+1]-Max_M[ii]);
else
B->HRki[ii]=float (A->HRk[ii+1]-A->HRk[ii])/float (Max_V[ii+1]-Max_V[ii]);
}
else B->HRki[ii]=999;
}

for (ii=0;ii<6; ii++)
{
if (A->Dk[ii+1]!=999)
{if (A->zing==25)
B->Dki[ii]=float (A->Dk[ii+1]-A->Dk[ii])/float (Max_M[ii+1]-Max_M[ii]);
else
B->Dki[ii]=float (A->Dk[ii+1]-A->Dk[ii])/float (Max_V[ii+1]-Max_V[ii]);
}
else B->Dki[ii]=999;
}

for (ii=0;ii<6; ii++)
{
if (A->Sk[ii+1]!=999)
{if (A->zing==25)
B->Ski[ii]=float (A->Sk[ii+1]-A->Sk[ii])/float (Max_M[ii+1]-Max_M[ii]);
else
B->Ski[ii]=float (A->Sk[ii+1]-A->Sk[ii])/float (Max_V[ii+1]-Max_V[ii]);
}
else B->Ski[ii]=999;
}

for (ii=0;ii<6; ii++)
{

```

```

if (A->JTk[ii+1]!=999)
  {if (A->zing==25)
    B->JTKi[ii]=float(A->JTk[ii+1]-A->JTk[ii])/float(Max_M[ii+1]-Max_M[ii]);
  else
    B->JTKi[ii]=float(A->JTk[ii+1]-A->JTk[ii])/float(Max_V[ii+1]-Max_V[ii]);
  }
else B->JTKi[ii]=999;
}

for (ii=0;ii<6; ii++)
{
if (A->RRk[ii+1]!=999)
  {if (A->zing==25)
    B->RRki[ii]=float(A->RRk[ii+1]-A->RRk[ii])/float(Max_M[ii+1]-Max_M[ii]);
  else
    B->RRki[ii]=float(A->RRk[ii+1]-A->RRk[ii])/float(Max_V[ii+1]-Max_V[ii]);
  }
else B->RRki[ii]=999;
}

for (ii=0;ii<6; ii++)
{
if (A->JTRRk[ii+1]!=999)
  {if (A->zing==25)
    B->JTRRki[ii]=float(A->JTRRk[ii+1]-A->JTRRk[ii])/float(Max_M[ii+1]-Max_M[ii]);
  else
    B->JTRRki[ii]=float(A->JTRRk[ii+1]-A->JTRRk[ii])/float(Max_V[ii+1]-Max_V[ii]);
  }
else B->JTRRki[ii]=999;
}

for (ii=0;ii<6; ii++)
{
if (A->SDk[ii+1]!=999)
  {if (A->zing==25)
    B->SDki[ii]=float(A->SDk[ii+1]-A->SDk[ii])/float(Max_M[ii+1]-Max_M[ii]);
  else
    B->SDki[ii]=float(A->SDk[ii+1]-A->SDk[ii])/float(Max_V[ii+1]-Max_V[ii]);
  }
else B->SDki[ii]=999;
}

for (ii=0;ii<6; ii++)
{
if (A->SDSk[ii+1]!=999)
  {if (A->zing==25)
    B->SDSki[ii]=float(A->SDSk[ii+1]-A->SDSk[ii])/float(Max_M[ii+1]-Max_M[ii]);
  else
    B->SDSki[ii]=float(A->SDSk[ii+1]-A->SDSk[ii])/float(Max_V[ii+1]-Max_V[ii]);
  }
else B->SDSki[ii]=999;
}

for (ii=0;ii<6; ii++)
{
if (A->DSk[ii+1]!=999)
  {if (A->zing==25)
    B->DSki[ii]=float(A->DSk[ii+1]-A->DSk[ii])/float(Max_M[ii+1]-Max_M[ii]);
  else
    B->DSki[ii]=float(A->DSk[ii+1]-A->DSk[ii])/float(Max_V[ii+1]-Max_V[ii]);
  }
else B->DSki[ii]=999;
}

return B;
}

```

```

    } // jei A netucia
}

//-----
void TAnalK1::Analyze(TObject *Sender) {
    Memol->Lines->Clear();
    AnsiString eilute, e, ee, eee, eeee;
    int i;
    Memol->Lines->Add(" Sveikatos vertinimas pagal Sv:");
    if ( PagK1->C->Sv>=0 & PagK1->C->Sv<40)
        Memol->Lines->Add("Žemas suminis sveikatos vertinimas. Yra ligos požymių. Būtini detalesni tyrimai");
    if ( PagK1->C->Sv>=40 & PagK1->C->Sv<50)
        Memol->Lines->Add("Sveikatos būklė ribinė.Derėtų susirūpinti");
    if ( PagK1->C->Sv>=50 & PagK1->C->Sv<70)
        Memol->Lines->Add("Sveikatos būklė vidutinė(normali)");
    if ( PagK1->C->Sv>=70 & PagK1->C->Sv<80)
        Memol->Lines->Add("Sveikata gera");
    if ( PagK1->C->Sv>=80)
        Memol->Lines->Add("Sveikatos būklė puiki.");
    Memol->Lines->Add("Fizinis krūvis adekvatus. Sveikata pagerėjo");
    Memol->Lines->Add("");

    Memol->Lines->Add(" Fizinių galimybių vertinimas pagal ŠSD:");

    if (PagK1->T->poz=="s" ) {
        for (i=0; i<PagK1->T->sk; i++) {
            if (PagK1->T->HRK[i]>220-PagK1->T->amzius)
                if (PagK1->T->zing==50) {
                    eilute= "Fizinių galimybių riba pagal ŠSD (";
                    ee=") ties ";
                    eeee= " W";
                    e=IntToStr(220-PagK1->T->amzius);
                    eee= IntToStr(Max_V[i]);
                    eilute=eilute+e+ee+eee+eeee;
                    Memol->Lines->Add(eilute);
                    return;
                }
            else {
                eilute= "Fizinių galimybių riba pagal ŠSD (";
                ee=") ties ";
                eeee= " W";
                e=IntToStr(220-PagK1->T->amzius);
                eee= IntToStr(Max_M[i]);
                eilute=eilute+e+ee+eee+eeee;
                Memol->Lines->Add(eilute);
                return;
            }
        }
        AnsiString riba=220-PagK1->T->amzius;
        Memol->Lines->Add("Fizinių galimybių riba (" +riba +") pagal ŠSD nepasiekta ");
    }

    if (PagK1->T->poz=="a" || PagK1->T->poz=="n" || PagK1->T->poz=="n2" || PagK1->T->poz=="n3" ||
        PagK1->T->poz=="n4" || PagK1->T->poz=="t" || PagK1->T->poz=="l") {
        for (i=0; i<PagK1->T->sk; i++) {
            if (PagK1->T->HRK[i]>0.85*(220-PagK1->T->amzius))
                if (PagK1->T->zing==50) {
                    eilute= "Fizinių galimybių riba pagal ŠSD (";
                    ee=") ties ";
                    eeee= " W";
                    e=0.85*(220-PagK1->T->amzius);
                    eee= IntToStr(Max_V[i]);
                    eilute=eilute+e+ee+eee+eeee;
                    Memol->Lines->Add(eilute);
                    return;
                }
            else {
                eilute= "Fizinių galimybių riba pagal ŠSD (";
                ee=") ties ";
                eeee= " W";
                e=0.85*(220-PagK1->T->amzius);
                eee= IntToStr(Max_M[i]);
            }
        }
    }
}

```

```

        eilute=eilute+e+ee+eee+eeee;
        Memol->Lines->Add(eilute);
        return;
    }

}
AnsiString riba=0.85*(220-PagK1->T->amzius);
Memol->Lines->Add("Fizinių galimybių riba (" +riba+" )pagal ŠSD nepasiekta" );
}

    if (PagK1->T->poz=="i") {
        for (i=0; i<PagK1->T->sk; i++) {
            if (PagK1->T->HRk[i]>0.75*(220-PagK1->T->amzius))
                if (PagK1->T->zing==50) {
                    eilute= "Fizinių galimybių riba pagal ŠSD (";
                    ee=") ties ";
                    eeee= " Ą";
                    e=0.75*(220-PagK1->T->amzius);
                    eee= IntToStr(Max_V[i]);
                    eilute=eilute+e+ee+eee+eeee;
                    Memol->Lines->Add(eilute);
                    return;
                }
                else {
                    eilute= "Fizinių galimybių riba pagal ŠSD (";
                    ee=") ties ";
                    eeee= " Ą";
                    e=0.75*(220-PagK1->T->amzius);
                    eee= IntToStr(Max_M[i]);
                    eilute=eilute+e+ee+eee+eeee;
                    Memol->Lines->Add(eilute);
                    return;
                }
            }
        }
        AnsiString riba=0.75*(220-PagK1->T->amzius);
        Memol->Lines->Add("Fizinių galimybių riba (" +riba+" )pagal ŠSD nepasiekta" );
    }

}
}
//-----

void TAnalK1::AnalyzeJT(TObject *Sender) {
    AnsiString eilute, e,ee, eee,eeee;
    int i;
    Memol->Lines->Add("");
    Memol->Lines->Add("    Fizinių galimybių vertinimas pagal JT:");

    for (i=0; i<PagK1->T->sk; i++) {
        if (PagK1->T->JTk[i]<1.92 || (i>1 & PagK1->T->JTk[i]>(PagK1->T->JTk[i-1]-0.05*PagK1->T->JTk[i-1])))
            if (PagK1->T->zing==50) {
                eilute= "Fizinių galimybių riba pagal JT (";
                ee=") arba pokyčio apsiribojimas ties ";
                eeee= " Ą";
                e="0.16";
                eee= IntToStr(Max_V[i]);
                eilute=eilute+e+ee+eee+eeee;
                Memol->Lines->Add(eilute);
                return;
            }
            else {
                eilute= "Fizinių galimybių riba pagal JT (";
                ee=") arba pokyčio apsiribojimas ties ";
                eeee= " Ą";
                e="0.16";
                eee= IntToStr(Max_M[i]);
                eilute=eilute+e+ee+eee+eeee;
                Memol->Lines->Add(eilute);
                return;
            }
        }
    }
    Memol->Lines->Add("Fizinių galimybių riba (0.16 s)pagal JT nepasiekta" );
}
}
//-----

```



```

void TAnalK1::AnalyzeS(TObject *Sender) {
    AnsiString eilute, e,ee, eee,eeee;
    int i;
    Memol->Lines->Add("");
    Memol->Lines->Add(" Fizinių galimybių vertinimas pagal S:");

    for (i=0; i<PagK1->T->sk; i++) {
        if ((PagK1->T->Sk[i]>240)|| (i>0 & PagK1->T->Sk[i-1]-PagK1->T->Sk[i]>5))
            if (PagK1->T->zing==50) {
                eilute= "Fizinių galimybių riba pagal S (";
                ee=") arba neigiamas pokytis ties ";
                eeee= " Ą";
                e="240";
                eee= IntToStr(Max_V[i]);
                eilute=eilute+e+ee+eee+eeee;
                Memol->Lines->Add(eilute);
                return;
            }
            else {
                eilute= "Fizinių galimybių riba pagal S (";
                ee=") arba neigiamas pokytis ties ";
                eeee= " Ą";
                e="240";
                eee= IntToStr(Max_M[i]);
                eilute=eilute+e+ee+eee+eeee;
                Memol->Lines->Add(eilute);
                return;
            }
        }
        Memol->Lines->Add("Fizinių galimybių riba (240)pagal S nepasiekta" );
    }
}

//-----
void TAnalK1::AnalyzeD(TObject *Sender) {
    AnsiString eilute, e,ee, eee,eeee;
    int i;
    Memol->Lines->Add(" ");
    Memol->Lines->Add(" Fizinių galimybių vertinimas pagal D:");

    for (i=0; i<PagK1->T->sk; i++) {
        if (PagK1->T->Dk[i]>120)
            if (PagK1->T->zing==50) {
                eilute= "Fizinių galimybių riba pagal D (";
                ee=") ties ";
                eeee= " Ą";
                e="120";
                eee= IntToStr(Max_V[i]);
                eilute=eilute+e+ee+eee+eeee;
                Memol->Lines->Add(eilute);
                return;
            }
            else {
                eilute= "Fizinių galimybių riba pagal D (";
                ee=") ties ";
                eeee= " Ą";
                e="120";
                eee= IntToStr(Max_M[i]);
                eilute=eilute+e+ee+eee+eeee;
                Memol->Lines->Add(eilute);
                return;
            }
        }
        Memol->Lines->Add("Fizinių galimybių riba (120)pagal D nepasiekta" );
    }
}

```