

Lenkiamų elementų įtempių ir deformacijų būvio analizė

Andrius Povilaitis

magistrantas,
Vilniaus Gedimino technikos universitetas,
andpov@gmail.

Anotacija. Nagrinėjami saugos ribiniai rodikliai – skersinės jėgos, lenkimo momentai, o taip pat nustatomi tinkamumo ribiniai būvio rodikliai – deviacijos ir įlinkiai. Šiame straipsnyje aptariama, kaip yra sudaromi lenkiamos dviatramės sijos skersinių jėgų ir lenkimo momentų grafikai ir kaip sudaromos deviacijų ir įlinkių kreivės. Pateiktas skaičiuojamojo algoritmo pritaikymo pavyzdys.

Reikšminiai žodžiai: skersinės jėgos, įlinkiai, lenkimo momentai, deviacijos, sija, kreivės.

Įvadas

Dviatramės sijos yra vienas iš pagrindinių ir labiausiai paplitusių elementų sudarančių sudėtingas konstrukcijas. Siekiant nustatyti jų stiprumą ir standumą būtina nagrinėti kiekvieno atskiro jos elemento t.y. dviatramės sijos elgseną ją veikiančių jėgų aplinkoje. Būtina nustatyti saugos ribinių būvių rodiklius – skersines jėgas, lenkimo momentus, o taip pat nustatyti tinkamumo ribinio būvio rodiklius – deviacijas ir įlinkius. Šiame straipsnyje aptariama, kaip yra sudaromi lenkiamos dviatramės sijos skersinių jėgų, lenkimo momentų, deviacijų ir įlinkių funkcijų vaizdai. Pateikiamas skaičiavimo algoritmas, jo taikymo pavyzdys.

Skersinių jėgų, lenkimo momentų, deviacijų ir įlinkių diferencialinis ryšys

Mechanikoje žinomas diferencialinis ryšys tarp lenkimo momento funkcijos ir apkrovos

$$\frac{d^2 M(z)}{dz^2} = q(z). \quad (1)$$

Čia $q(z)$ apkrovos pasiskirstymo funkcija, z – taškas sijos atžvilgiu.

Skersinių jėgų pasiskirstymo funkcija sijos atžvilgiu (2).

$$Q(z) = \int \frac{d^2 M(z)}{dz^2} dz = \int q(z) dz + Q_i. \quad (2)$$

Čia Q_i – skersinės jėgos reikšmė sijos ruožo galiniame pjūvyje.

Lenkimo momentų pasiskirstymo funkcija nediferencialine forma (3).

$$M(z) = \int Q(z) dz = \iint q(z) dz dz + Q_i z + M_i. \quad (3)$$

Čia M_i – lenkimo momento reikšmė duotajame taške.

Apytikslė diferencialinė įlinkių funkcija (4) yra lygi lenkimo momentų pasiskirstymo funkcijai tik su priešingu ženklu

$$EI \frac{d^2 v}{dz^2} = -M(z). \quad (4)$$

Čia EI – lenkiamasis standis.

Deviacijų funkcija gaunama iš funkcijos (4).

$$EI \varphi(z) = EI \frac{dv}{dz} = -\int M(z) dz + C. \quad (5)$$

Nediferencialinė įlinkių kreivės lygtis (6).

$$EI v(z) = \int EI \varphi(z) dz = -\iint M(z) dz + Cz + D. \quad (6)$$

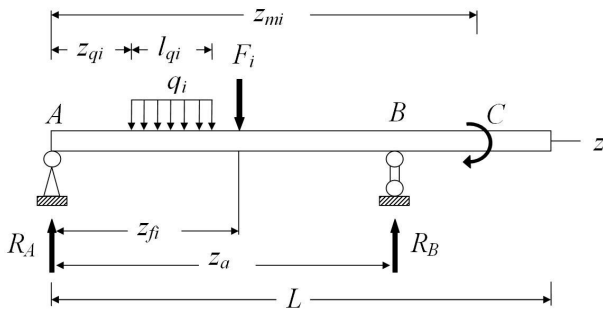
Čia C ir D - integravimo konstantos, kurių reikšmės nustatomos iš kraštinių sąlygų.

Iš (1), (2), (3) ir (4) lygčių seka, jog bendrą sąryšį tarp šių funkcijų galima užrašyti taip:

$$EI \frac{d^4 v(z)}{dz^4} = EI \frac{d^3 \varphi(z)}{dz^3} = \frac{d^2 M(z)}{dz^2} = \frac{dQ(z)}{dz} = q(z). \quad (7)$$

Dviatramės sijos skersinių jėgų ir lenkimo momentų diagramų sudarymo algoritmas

1 pav. pateikiama sprendžiamojo apibendrinto uždavinio schema.



1 pav. Sprendžiamo apibendrinto uždavinio schema

Atraminės reakcijos

Skaičiuojant lenkiamąsias dviatrames sijas, pirmiausia apskaičiuojamos atraminės reakcijos jėgos, o po to skaičiuojamos įrašos.

Iš lenkimo momentų pusiausvyros lygties (8) išreiškiama atraminė reakcija R_B (9), o iš išorinių jėgų pusiausvyros lygties (10) išreiškiama atramos reakcijos jėga R_A (11)

$$\sum M_A = R_B \cdot z_a - \sum_{i=1}^n F_i \cdot z_{fi} - \sum_{i=1}^n q_i \cdot l_{qi} \cdot \left(z_{qi} + \frac{l_{qi}}{2} \right) + \sum_{i=1}^n C_i = 0, \quad (8)$$

$$R_B = \frac{\sum_{i=1}^n F_i \cdot z_{fi} + \sum_{i=1}^n q_i \cdot l_{qi} \cdot \left(z_{qi} + \frac{l_{qi}}{2} \right) - \sum_{i=1}^n C_i}{z_a}. \quad (9)$$

$$\sum F_y = R_A - \sum_{i=1}^n F_i + \sum_{i=1}^n q_i \cdot l_{qi} + R_B = 0, \quad (10)$$

$$R_A = \sum_{i=1}^n F_i + \sum_{i=1}^n q_i \cdot l_{qi} + R_B. \quad (11)$$

Skersinių jėgų funkcijos

Skersinių jėgų pasiskirstymo funkcija išreiškiama iš išorinių jėgų pusiausvyros lygties (12)

$$\sum F_y = R_A - V_F(z) - V_q(z) - V(z) + R_B(z) = 0, \quad (12)$$

$$V(z) = R_A - V_F(z) - V_q(z) + V_{RB}(z) = 0. \quad (13)$$

Skersinė jėga nuo sutelktos apkrovos:

$$V_F(z) = \sum_{i=1}^n F_i, \text{ kai } z \geq z_{fi}. \quad (14)$$

Skersinė jėga nuo paskirstytos apkrovos:

$$V_q(z) = \sum_{i=1}^n q_i \cdot z \cdot l_{qi}, \text{ kai } z \geq z_{qi} \text{ ir } z \geq (z_{qi} + l_{qi}),$$

$$V_q(z) = \sum_{i=1}^n q_i \cdot z \cdot (z - z_{qi}),$$

$$\text{kai } z \geq z_{qi} \text{ ir } z < (z_{qi} + l_{qi}). \quad (15)$$

Skersinė jėga nuo atramos reakcijos jėgų.

$$V_{RB}(z) = R_B, \text{ kai } z > z_a, \quad (16)$$

$$V_{RB}(z) = 0, \text{ kai } z \leq z_a.$$

Lenkimo momentų funkcijos

Iš lenkimo momentų pusiausvyros lygties (17) išreiškiama lenkimo momentų įrašų funkcija (18).

$$\sum M_y = M(z) + M_F(z) + M_q(z) + M_c(z) - R_A \cdot z - M_{RB}(z) = 0, \quad (17)$$

$$M(z) = -M_F(z) - M_q(z) - M_c(z) + R_A \cdot z + M_{RB}(z) = 0. \quad (18)$$

Lenkimo momentų įrašų lygtis nuo paskirstytos apkrovos

$$M_q(z) = \sum_{i=1}^n q_i \cdot z \cdot l_{qi} \cdot \left[z - (z_{qi} + l_{qi}) \right] + \frac{l_{qi}^2}{2}, \text{ kai } z \geq z_{qi} \text{ ir } z \geq (z_{qi} + l_{qi}), \quad (19)$$

$$M_q(z) = \sum_{i=1}^n q_i \cdot z \cdot (z - z_{qi}) \cdot \frac{(z - z_{qi})}{2}, \text{ kai } z \geq z_{qi} \text{ ir } z < (z_{qi} + l_{qi}).$$

Lenkimo momentų įrašų lygtis nuo sutelktos išorinės jėgos

$$M_F(z) = \sum_{i=1}^n F_i \cdot (z - z_{fi}) \cdot z, \text{ kai } z \geq z_{fi} \quad (20)$$

Lenkimo momentų įrašų lygtis nuo išorinių momentų

$$M_c(z) = \sum_{i=1}^n C_i, \text{ kai } z \geq z_{mi}. \quad (21)$$

Lenkimo momentų įrašų lygtis nuo atramos reakcijos jėgų

$$M_{RB}(z) = R_B \cdot (z - z_a), \text{ kai } z > z_a, \quad (22)$$

$$M_{RB}(z) = 0, \text{ kai } z \leq z_a.$$

Deviacijų ir įlinkių funkcijos

Šiame skyriuje aptariama kaip yra sudaromi deviacijų ir įlinkių skaičiuojamieji algoritmai. Panaudojus superpozicijos principą kiekvienai sija veikiančiai jėgai ieškoma lenkimo momentų lygtis darant prielaidą, kad sija veikia tik viena jėga, apskaičiuojamos reakcinės jėgos ir iš lenkimo momentų pusiausvyros lygties randama lenkimo momento lygtis. Integruojant šias lygtis viena kartą gaunamos deviacijų kreivės funkcijos, o integruojant du kartus gaunamos atitinkamų jėgų įlinkių kreivės funkcijos nediferencialine forma. Nustatomos kraštinės sąlygos. Gautos atitinkamos lygtys sumuojamos.

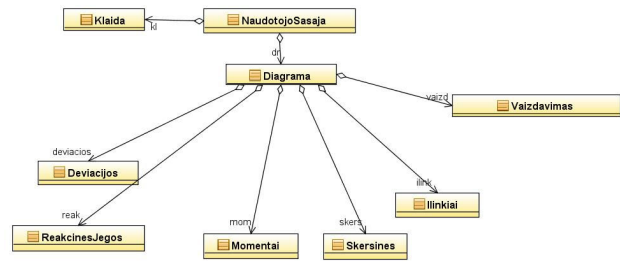
Iš bendros lenkimo momentų įrašų lygties (17) pagal superpozicijos principą kiekvieną lenkimo momentų įrašų funkciją galima sudaryti atskirai (23).

$$M_i(z) = R_{Ai} \cdot z - M_{Fi}(z) - M_{qi}(z) - M_{Ci}(z) - M_{RBi}(z). \quad (23)$$

Pagal (5) ir (6) lygtis bendroms deviacijų ir įlinkių kreivės funkcijoms nustatomos kraštinės sąlygos. Sudaroma lygčių sistema ir randamos C ir D integravimo konstantos.

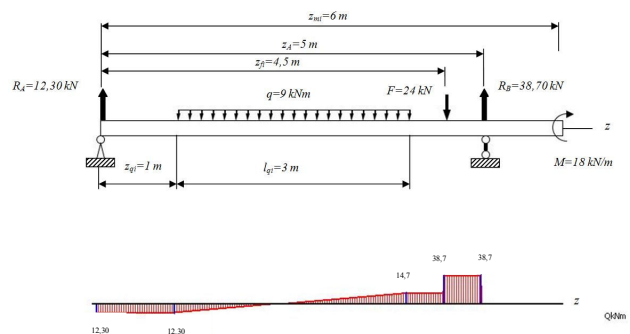
Taikomosios programos algoritmo realizacija

2 pav. pateikta sukurtos taikomosios programos klasių diagrama. Joje aiškiai išskirstyti klasių funkcionalumai t.y. kiekviena klasė yra atsakinga tik už jai priskirtas funkcijas. Joje matomi taikomosios programos procesai. Klasė „NaudotojoSasaja“ pateikia naudotojo sąsają naudotojui ir yra atsakinga už taikomosios programos komponentų išdėstymą ekrane. Klasė „Klaida“ nurodo kaip ir kur suklydo naudotojas formuojant uždavinio schemą. Klasės: „Deviacijos“, „ReakcinesJegos“, „Momentai“, „Skersines“, „Ilinkiai“ yra atsakingos už skersinių jėgų, lenkimo momentų, deviacijų ir įlinkių reikšmių skaičiavimus. Visi šių klasių rezultatai saugomi vienmačiuose masyvuose. Klasė „Vaizdavimas“ yra atsakinga už vaizdinius elementus t.y. uždavinio schemos ir diagramų – grafikų vaizdinių komponentų sukūrimą. Klasė „Diagrama“ pateikia uždavinio schemą, diagramas-grafikus.



2 pav. Taikomosios programos klasių diagrama

3 pav. pateikta skersinių jėgų diagrama. Aukščiau pateikta sprendžiamo uždavinio schema, o po ja skersinių jėgų diagrama.



3 pav. Skersinių jėgų skaičiuojamojo algoritmo pritaikymas

Išvados

1. Sudaryti algoritmai gana efektyviai apskaičiuoja sijos įtempių ir deformacijų būvių kintamųjų reikšmes ir sudaro funkcijų grafikus - diagramas.
2. Sudarant skaičiuojamuosius algoritmus būtina nurodyti sprendžiamo uždavinio ribas, priešingu atveju skaičiuojamasis algoritmas bus sudėtingas.
3. Skaičiuojamasis algoritmas sudarytas taip, kad kiekvienas jo žingsnis būtų izoliuotas, dėl to algoritmas bus lengviau plečiamas ir bus išvengta klaidų.
4. Galimybė sukurta programa spręsti įvairiausių lenkiamųjų sistemų uždavinius ir gaunamų rezultatų vaizdumas padės studijuojantiems geriau suvokti dalyko esmę ir geriau įsisavinti temą.
5. Programa gali būti pritaikoma projektavimo praktikoje, nes yra nesudėtinga, patogi naudotojo sąsaja, platus sprendžiamų uždavinių spektras.

Literatūra

Diliūnas S. Jutas A. Medžiagų mechanikos uždaviniai. Kaunas: Technologija, (2007)

Ulozas R. V., Statybos inžinerijos studijų programos (Dėstytojo/Mokytojo) pagalbininkas I dalis. Šiauliai (2007)

Žiliukas A., Medžiagų mechanika : vadovėlis aukštųjų mokyklų studentams, Kaunas: Technologija, (2004)

STRAIN-STRESS STATE ANALYSIS OF BEAMS

A. Povilaitis

Summary

Security issue of bounding parameters for simply supported beam was reviewed in the article. It is described how these parameters are pictured on shear force and bending moment diagrams. Also the way how deflection and deviation curves are defined is described. These curves are used to describe parameters of fitness for the simply supported beam. The algorithm for shear force diagram making is realized by java programming language

AMONIAKO GAMYBOS METANAVIMO PROCESO MODELIAVIMAS

Giedrius Oberauskas¹

¹ doktorantas, Kauno technologijos universitetas
el. p. ¹GiedriusO@gmail.com

Anotacija. Straipsnyje pateikti ir išanalizuoti amoniako gamybos metanavimo proceso modeliavimo principai, sudarytos masės balanso ir cheminių reakcijų greičių lygtys. Sukurta imitacinio modeliavimo programa *Matlab* programinėje terpėje. Gauti modeliavimo rezultatai palyginti su eksperimentiniais duomenimis ir pateikta rezultatų analizė.

Reikšminiai žodžiai: reakcijų greičiai, masės balansas, metanavimas, amoniako gamyba

Išvadas

Šiame straipsnyje pristatytas amoniako gamybos metanavimo (angl. *methanation*) proceso modeliavimas, kuris jungia masės balanso sistemą atskiriems komponentams ir mechanistinius absoliutinių reakcijų greičių (angl. *mechanistic*) dalinius modelius (angl. *sub-models*).

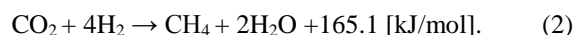
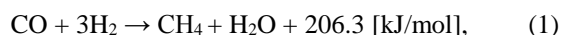
Esant sudėtingoms reakcijoms procesuose, tokiuose kaip biotechnologiniai ir cheminiai procesai (Galvanauskas V. 2006; Simutis R. 2004; Schubert J. 1994) ne visada galima aprašyti procesus, naudojant tik mechanistinius modelius. Todėl reikia naudoti metodus, kurie užpildo specialiųjų chemijos ar inžinerijos žinių spragas. Tolimesni žinių praplėtimai, kurie paremti metodais, naudojamais chemijos inžinerijoje, yra masės balansų sistemos. Pastaruoju metu spausdintoje literatūroje keletas pavyzdžių rodo, kad hibridinis dirbtinių neuroninių tinklų (DNT), mechanistinės kinetikos ir masės balanso lygčių naudojimas kartu gali turėti žymių privalumų (Galvanauskas V. 2006; Simutis R. 2004; Schubert J. 1994; Levišauskas D. 2005). Vis dėlto chemijoje dirbama su dideliu kiekiu ne realiame laike gaunamų (angl. *off-line*) arba beveik ne realiame laike gaunamų (angl. *quasi-off-line*) duomenų tipais ir todėl tokių tipų duomenys yra papildomi naudojant interpoliavimo metodus. Deja, tai gali sukelti dirbtinius trikdžius, be to, kokybė taip pat priklauso nuo interpoliavimo metodų tikslumo. Kaip bebūtų, chemijos technologijoje labiausiai kreipiamas dėmesys į cheminių reakcijų greičius, kadangi jie apibrėžia cheminių procesų mainų pobūdį.

Chemijos inžinerijoje reikalingi patikimi ir veiksmingi procesų modeliavimo metodai, kai dirbame su sudėtingais cheminiais procesais, kurie negali būti modeliuojami reikalaujama tikslumu, naudojant tik paprastas inžinierines koreliacijas ir mechanistinius modelius.

Amoniako gamybos proceso aprašas

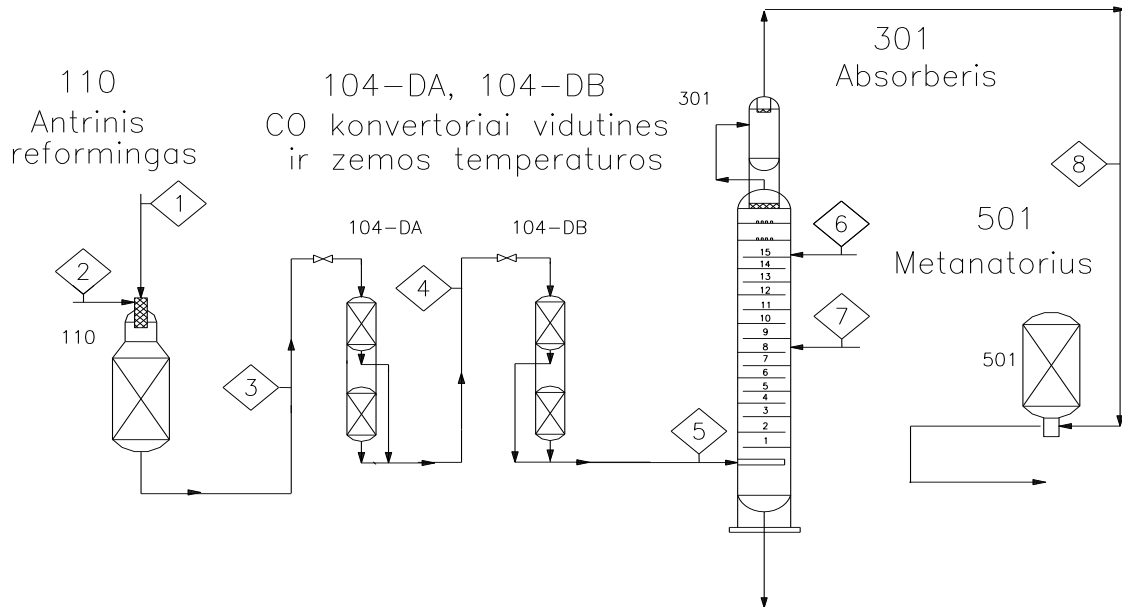
Amoniako gamyba sudaryta iš keleto gamybos stadijų, tokių kaip pirminis ir antrinis reformingai, CO konversija, CO₂ konversija, metanavimas, amoniako sintezė. Technologinė schema pateikta 1 pav. Šiame straipsnyje trumpai pristatytas ir išnagrinėtas metanavimo procesas.

Remiantis amoniako gamybos technologija, maži CO ir CO₂ kiekiai, esantys sintezės dujose, yra toksiški amoniako sintezės procesui, todėl metanavimo proceso metu, reaguojant su vandeniliu, metanas susidaro pagal chemines reakcijos lygtis (1) ir (2):



Amoniako gamybos metanavimo proceso masės balanso lygtys

Cheminių komponentų masės balanso lygtys yra pagrįstos masės tvermės dėsnium, kuris reiškia, kad komponentų koncentracijos kitimas gali vykti dėl cheminės reakcijos, kurios metu cheminės reakcijos komponentai jungiasi vienas su kitu į sudėtingesnę produkto formą arba skyla į paprastesnius junginius ar



1 pav. Amoniako gamybos proceso technologinė schema

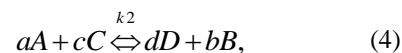
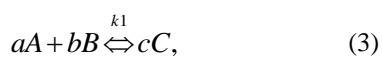
elementus. Be to, komponentų kiekiai gali keistis paduodant atitinkamą komponentą į reaktorių ir atitinkamai iš reaktoriaus pašalinant dalį jo turinio. Diferencialinė medžiagos C_{iout} masės balanso lygtis:

$$\frac{d(C_{iout})}{dt} = q_i C_i + C_{in} \frac{F_{in}}{V} - C_{iout} \frac{F_{in}}{V}, \quad (2)$$

čia: C_{iout} – i -ojo komponento modeliuojama koncentracija, [mol/l]; C_{in} – i -ojo komponento koncentracija esanti sraute, [mol/l], $q_i C_i$ – i -asis absoliutinis reakcijos greitis, [mol/(ls)]; V – įrenginio tūris kuriame vyksta cheminė reakcija, [l]; F_{in} – srautas į reakciją, [l/s].

Pagal (1) ir (2) lygtis ir duomenis iš amoniako gamybos technologinio reglamento yra sudaromos masės balanso lygtys visiems komponentams. Argonas ir azotas yra inertinės dujos, todėl jos nedaro tiesioginės įtakos metanavimo proceso masės balanso lygčių sistemai.

Sudėtinėms reakcijoms dalyvaujančių komponentų koncentracijų paieškos procedūra prasideda tariant, kad reakcijos vyko be pertraukos. Tarpinės koncentracijos žymimos indeksais. Gaunama galutinė koncentracija iš tam tikros reakcijos su pradine koncentracija ir taip pereinama prie kitų reakcijų. Galutinė vertė užrašoma be indekso (Perry R.H. 1997). Absoliutinių reakcijų greičių lygtys ieškomiems komponentams gali būti surastos taip:



čia: a, b, c , ir d cheminių lygčių empiriniai koeficientai; A, B, C , ir D – cheminių lygčių komponentai; k_1 ir k_2 – reakcijų greičių konstantos.

Absoliutinis reakcijos greitis komponentams (3) ir (4) cheminėse lygtyse yra:

$$r_A = -ak_1 A^a B^b - ak_2 A^a C^c, \quad (5)$$

$$r_B = -bk_1 A^a B^b + bk_2 A^a C^c, \quad (6)$$

$$r_C = ck_1 A^a C^c - ck_2 A^a C^c, \quad (7)$$

$$r_D = dk_2 A^a C^c, \quad (8)$$

čia: k_1 ir k_2 – reakcijų greičių konstantos (3) ir (4) cheminėms lygtims.

Lygtys (5-8) pateiktos komponentams A, B, C ir D kaip laiko funkcijos. Diferencialinės lygtys taip pat gali būti užrašytos ir sprendžiamos tiesiogiai ieškomiems komponentams kaip laiko funkcijos, tokiu būdu išvengiant stochiometrinių balansų, nors jie yra susiję su diferencialinėmis lygtimis (Perry R.H. 1997).

Reakcijų greičių konstanta tiesiogiai priklauso nuo proceso temperatūros ir todėl lemia absoliutinį reakcijos greitį. Arėnijaus lygtis, kuri naudojama ieškant absoliutinio reakcijos greičio, taip pat įvertina proceso absoliutinės temperatūros poveikį cheminėms reakcijoms (3) ir (4):

$$k = k_0 \exp\left(-\frac{E}{RT}\right) = K \exp\left(a - \frac{b}{T}\right), \quad (9)$$

čia: a ir b – cheminių lygčių empiriniai koeficientai; K ir k_0 – proporcingumo koeficientai; E – aktyvacijos energija, [J]; R – universalioji dujų konstanta, [J/(K mol)].

Paprastai cheminių procesų absoliutiniai reakcijų greičiai nėra pastovūs ir priklauso nuo įvairių veiksnių, tokių kaip proceso temperatūra, slėgis, katalizatoriaus aktyvumas ir t. t. Proporcionalumo koeficientas yra reikalingas tam, kad būtų galima įvertinti ne tik proceso temperatūrą, bet ir proceso slėgį, katalizatoriaus aktyvumą bei kitus svarbius parametrus, kurie lemia absoliutinį reakcijos greitį.

Modeliuojant amoniako gamybos metanavimo proceso masės balansus remiamasi amoniako gamybos technologinio reglamento duomenimis. Duomenų matavimo vienetai, tokie kaip koncentracija, srautas ir tūris nesutampa su matavimo vienetais, kurie naudojami modeliuojant. Pvz., masės balansuose koncentracija yra [mol/l], o amoniako gamybos technologiniame reglamente ji išreikšta procentais tūrio dalies, tokiais pačiais matavimo vienetais matuojamos koncentracijos ir technologiniame procese. Dėl išvardytų priežasčių koncentracijų matavimo vienetai yra perskaičiuojami į reikiamus matavimo vienetus, įvertinant proceso slėgį ir temperatūrą (žr. 1 lentelę):

$$V = \frac{nRT}{p}, \quad (16)$$

čia: n – medžiagos kiekis, [mol]; T – proceso temperatūra, [K]; p – proceso absoliutinis slėgis, [Pa].

1 lentelė. Metanavimo proceso duomenys

Parametrai	Reikšmės
Temperatūra, [°C]	295,00
Slėgis, [kPa]	2518,3
Srautas F_i , [l/s]	49120
C_{CO_2i} , [mol/l]	1,59e-4
C_{COi} , [mol/l]	1,27e-3
C_{H_2i} , [mol/l]	0,3901
C_{CH_4i} , [mol/l]	1,43e-3
$C_{H_2O_i}$, [mol/l]	2,97e-3

Amoniako gamybos metanavimo proceso masės balansų ir reakcijos greičių lygtys atskiriems cheminiams elementams yra sudaromos remiantis anksčiau

apibūdintais metodais. Masės balanso lygtys metanavimo procesui atskiriems komponentams pagal chemines reakcijos lygtys (1) ir (2) yra tokios:

$$\frac{d(C_{CO_2out})}{dt} = r_{CO_2} + (C_{CO_2in} - C_{CO_2out}) \frac{F_1}{V_1}, \quad (17)$$

$$\frac{d(C_{COout})}{dt} = r_{CO} + (C_{COin} - C_{COout}) \frac{F_1}{V_1}, \quad (18)$$

$$\frac{d(C_{H_2out})}{dt} = r_{H_2} + (C_{H_2in} - C_{H_2out}) \frac{F_1}{V_1}, \quad (19)$$

$$\frac{d(C_{CH_4out})}{dt} = r_{CH_4} + (C_{CH_4in} - C_{CH_4out}) \frac{F_1}{V_1}, \quad (20)$$

$$\frac{d(C_{H_2Oout})}{dt} = r_{H_2O} + (C_{H_2Oin} - C_{H_2Oout}) \frac{F_1}{V_1}, \quad (21)$$

čia: F_1 – garo/dujų srautas į metanatorių, [l/s]; V_1 – m, [l]; C_{CO_2in} , C_{COin} , C_{H_2in} , C_{CH_4in} , C_{H_2Oin} – komponentų koncentracijos esančios garo/dujų sraute (F_1) į metanatorių, [mol/l]. C_{CO_2} , C_{CO} , C_{H_2} , C_{CH_4} , C_{H_2O} – modeliuojamos komponentų koncentracijos dujų/garo sraute po metanavimo proceso, [mol/l];

Absoliutinės reakcijų greičių lygtys komponentams:

$$r_{CO_2} = -k_2 C_{CO_2} C_{H_2}^4, \quad (22)$$

$$r_{CO} = -k_1 C_{CO} C_{H_2}^3, \quad (23)$$

$$r_{H_2} = -3k_1 C_{CO} C_{H_2}^3 - 4k_2 C_{CO_2} C_{H_2}^4, \quad (24)$$

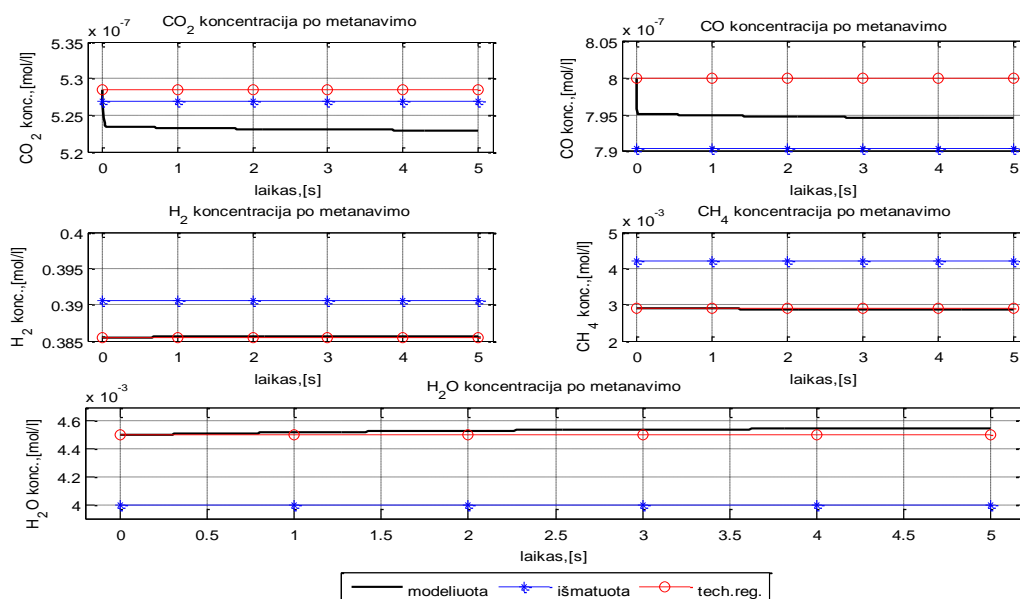
$$r_{CH_4} = k_1 C_{CO} C_{H_2}^3 + k_2 C_{CO_2} C_{H_2}^4, \quad (25)$$

$$r_{H_2O} = k_1 C_{CO} C_{H_2}^3 + 2k_2 C_{CO_2} C_{H_2}^4, \quad (26)$$

čia: k_1 – cheminės lygties (1) reakcijos konstanta; k_2 – cheminės lygties (2) reakcijos konstanta. r_{CO_2} , r_{CO} , r_{H_2} , r_{CH_4} , r_{H_2O} – komponentų koncentracijų absoliutiniai reakcijų greičiai metanatoriuje, [mol/(ls)]; C_{CO_2} , C_{CO} , C_{H_2} , C_{CH_4} , C_{H_2O} – modeliuojamos komponentų koncentracijos dujų ir garo sraute po metanavimo proceso, [mol/l];

Išvados

1. Sukurtas kompiuterinis imitacinis amoniako gamybos metanavimo proceso matematinis modelis *Matlab/Simulink* programos aplinkoje remiantis amoniako gamybos technologiniu reglamentu.



2 pav. Amoniaکو gamybos proceso technologinė schema

2. Modeliavimo rezultatai palyginti su realiais gamyboje išmatuotais duomenimis.
3. Modelio rezultatams papildomai buvo įvestas proporcingumo koeficientas tam, kad įvertintų vykstančios cheminės reakcijos šalutinius poveikius, tokius kaip katalizatoriaus aktyvumą, proceso slėgį ir kitus svarbius parametrus.
4. Kadangi modelis sudarytas remiantis amoniako gamybos technologiniu reglamentu ir technologiniame reglamente koncentracijų reikšmės pateiktos procentais tūrio masės dalies, todėl programinė perskaičiavimo procedūra lemia papildomas modeliavimo rezultatų paklaidas.
5. Vietoj absoliutinių reakcijos greičio lygčių bus naudojami dirbiniai neuroniniai tinklai (DNT), kurie bus apmokomi iš gamybos gautais eksperimentiniais duomenimis.

Padėkos

Dėkoju dr. Vytautui Galvanauskui už pagalbą rengiant straipsnį.

Literatūra

- Galvanauskas V., Georgieva P. and Fejo de Azevedo S. Dynamic Optimisation of Industrial Sugar Crystallization Process based on a Hybrid (Mechanistic + ANN) Model. Proceedings of IEEE World Congress on Computational Intelligence, IJCNN 2006, July 16-21, 2006, Vancouver, Canada, Vol. 6. p. 5035-5042.
- Galvanauskas V., Simutis R. and Lübbert A. Hybrid process models for process optimisation, monitoring

and control. *Bioprocess and Biosystems Engineering*, 2004, 26(6). p. 393-400.

Schubert J., Simutis R., Dors M., Havlik I. and Lübbert A. Bioprocess optimization and control: Application of hybrid modelling, *J. Biotechn.*, 1994, 35. p. 51-68.

Galvanauskas V., Simutis R. and Levišauskas D. Application of evolutionary computing for hybrid model based optimization of biochemical processes. *WSEAS Transactions on Information Science and Applications*, Issue 5, Vol. 2, May 2005. p. 507-511.

Perry R.H., Green D.W. and Moloney J.O. *Chemical Engineers' Handbook*. 7th edition, McGraw-Hill Professional, 1997. 2640 p.

MODELLING OF AMMONIA PRODUCTION METHANATION PROCESS

Giedrius Oberauskas

Summary

In this paper a detailed first principles mathematical model of industrial ammonia production methanation process is elaborated. Mass balance equations and chemical reaction rate equations of the process are presented and analyzed. Computer simulation tests were performed using software tools created in Matlab program environment. The modelling results were compared with the experimental data and discussed.

MOKSLINIO ŽURNALO TINKLAPIO KŪRIMAS

Ilma Ivanauskaitė

Magistrantė

Vilniaus Gedimino technikos universitetas

Anotacija. Nagrinėjamas mokslinio žurnalo tinklapio kūrimas, paieškos algoritmai naudojami MySQL duomenų bazėse. Analizuojamas šių algoritmų efektyvumas. Atliktas kuriamo tinklapio palyginimas su kitų mokslinių žurnalų tinklapiais.

Reikšminiai žodžiai: MySQL, indeksas, B-medis, Turbo Boyer-Moore algoritmas.

Įvadas

„Mathematical Modelling and Analysis“ žurnalas – tarptautinis žurnalas apie matematikos taikymus, skaitinius metodus ir diferencialines lygtis. „Mathematical Modelling and Analysis“ žurnalo tinklalapyje publikuojami rūpestingai atrinkti aukštos kokybės straipsniai, pristatantys naujus ir svarbius rezultatus visose matematinio modeliavimo ir analizės srityse.

Straipsniams publikuoti yra sukurta nemažai programinės įrangos, pavyzdžiui Article System, ePublishing Toolkit, Open Journal System ir kt. Tačiau kuriant „Mathematical Modelling and Analysis“ žurnalo tinklapį buvo nuspręsta naudoti geriausias straipsnių publikavimo įrankių savybes ir sukurti kompaktišką bei mobilią tinklapį, kurį bet kuriuo metu bus galima atnaujinti pagal naujai atsiradusius ar pasikeitusius reikalavimus. Tinklapis buvo sukurtas 2001 metais, tačiau per laiką atsirado poreikis įvesti naujas funkcijas, tokias kaip kelių kalbų palaikymas, vartotojo sąsajos atnaujinimas bei galimybė atlikti paiešką per visus žurnalo tomus pagal įvairius kriterijus. Kuriant „Mathematical Modelling and Analysis“ žurnalo tinklapį naudojama MySQL duomenų bazė, PHP, JavaScript programavimo kalbos.

Prieš tobulinant „Mathematical Modelling and Analysis“ žurnalo tinklapį buvo atlikta analizė ir palyginimas su kitų žurnalų tinklapiais. Atliekant analizę daugiau dėmesio buvo kreipiama į tinklapių funkcionalumą, lyginamos visiems vartotojams matomos tinklapių savybės ir neanalizuojami straipsnių pateikimo ar recenzavimo žingsniai.

Tinklapyje yra išskirti keli vartotojų tipai, kurie priklauso nuo vartotojui suteiktų teisių. Viena dalis yra

visiems vartotojams viešai prieinama dalis, kurioje vartotojas gali pamatyti žurnale publikuotus ar publikavimui pateiktus straipsnius bei atlikti paiešką per visus žurnalo tomus pagal įvairius kriterijus. Antra tinklapio dalis, tai registruotiems vartotojams prieinama dalis. Šioje dalyje vartotojai gali naudotis tomis pačiomis galimybėmis kaip ir viešai prieinamoje tinklapio dalyje, o taip pat ir papildomomis galimybėmis, tokiomis kaip vartotojo informacijos peržiūra ar redagavimas, galimybė pateikti straipsnį ar peržiūrėti straipsnių pateikimo eigą ir kt. Kadangi buvo tobulintos tik šios dvi tinklapio dalys, kitų dalių, tokių kaip straipsnių recenzentų bei redkolegijos dalys, neapžvelgsiu. Straipsnių paieškai sukurta forma, kurioje vartotojas gali pasirinkti įvairius kriterijus ir atlikti paiešką visuose žurnalo tomuose. Paieškai atlikti naudojama B-medžio struktūra bei kai kuriais atvejais naudojamas Turbo Boyer-Moore paieškos algoritmas.

Mokslinių žurnalų tinklapių palyginimas

Norint žinoti, kokio funkcionalumo trūksta „Mathematical Modelling and Analysis“ žurnalo tinklapiui, buvo atliktas žurnalo tinklapio palyginimas su Vilniaus Gedimino technikos universiteto (VGTU) žurnalų tinklapiais, su kitų universitetų ar institutų žurnalų tinklapiais bei su kitų šalių mokslinių žurnalų tinklapiais.

Lyginant „Mathematical Modelling and Analysis“ žurnalo (1 lentelė) tinklapį su kitais Vilniaus Gedimino technikos universiteto žurnalų tinklalapiais, buvo išanalizuota trylika mokslinių žurnalų: 1-„Mathematical Modelling and Analysis“ (VGTU), 2-„Journal of civil Engineering and management international research and achievements“, 3-„Geodezija ir kartografija“, 4-

Redaguojamų straipsnių būsenos peržiūra	x											
Recenzentų peržiūra	x	x	x		x							
Puslapis įvairiomis kalbomis		x		x	x	x	x	x	x	x		
Santraukų peržiūra	x			x				x	x	x	x	x

Tai reiškia, kad daugumai nėra labai svarbi žurnalo internete pateikimo forma ir apsiribojama bendru funkcionalumu. Duomenų bazes ir su jomis susijusį funkcionalumą turi tik nedaugelis tinklalapių. Kaip matome visi tinklalapiai, kurie turi duomenų bazes, turi ir straipsnių paiešką per visus tomus. Iš visų kitų funkcijų, vėl gi, matome, kad „Mathematical Modelling and Analysis“ yra pirmaujantis, nei viename iš nagrinėtų tinklalapių nėra nei forumo, nei registruotų vartotojų, nei paieškos pagal pasirinktus kriterijus ir kt. funkcionalumo.

straipsnių parsisiuntimas yra mokamas arba dalinai mokamas. Pastebėta, kad kitų šalių žurnalų tinklalapiuose nemaža dalis funkcionalumo yra įkelta į registruoto vartotojo dalį, t.y. neregistruotas vartotojas gali tik peržiūrėti publikuotus straipsnius ir bendrą informaciją apie straipsnį. Skirtingai nei „Mathematical Modelling and Analysis“ žurnalo tinklapis, kitų šalių žurnalų tinklalapiuose neregistruoti vartotojai neturi galimybės peržiūrėti ruošiamų straipsnių, straipsnių recenzentų ar pateikto straipsnio būsenos kitimo, t.y. „Mathematical Modelling and

3 lentelė. „Mathematical Modelling and Analysis“ žurnalo tinklapių palyginimas su kitų šalių mokslinių žurnalų tinklalapiais

	1	2	3	4	5	6	7	8	9	10
Neturi savo puslapio (bendras žurnalų tinklapis)					www.versita.com					
Yra duomenų bazė	x				x	www.ams.org				x
Registruoti vartotojai	x				x	www.ams.org				x
Forumas	x				x	www.ams.org				x
Galimybė parsisiųsti straipsnius (nemokamai)	x		x		x		x	x	x	+-
Galima straipsnių paieška per visus tomus					x	www.ams.org		x		x
Paieška pagal pasirinktus kriterijus	x					www.ams.org				x
Paieška tik pagal autorius				x	x	x		x	x	
Rengiamų straipsnių peržiūra (pačių straipsnių)	x									
Rengiamų straipsnių pavadinimai, autoriai	x								x	x
Žurnalo kūrėjų informacija	x	x	x	x	x	x	x	x	x	x
Redaguojamų straipsnių būsenos peržiūra	x									
Recenzentų peržiūra	x	x		x						
Puslapis įvairiomis kalbomis			x				x		x	
Santraukų peržiūra	x	x	x	x	x	x	x	x	x	x
Autorių informacijos peržiūra	x	x	x	x	x	x	x	x	x	x

Lyginant tinklapius (3 lentelė), buvo analizuojami tokių žurnalų tinklalapiai: 1-„Mathematical Modelling and Analysis“ (VGTU), 2-„The ACM Journal of Experimental Algorithmics“, 3-„Algebra Montpellier Announcements“, 4-„Algebra and Geometric Topology“, 5-„Central European Journal of Mathematics“, 6-„Conformal Geometry and Dynamics“, 7-„Differential Equations and Control Processes“, 8-„Electronic Journal of Combinatorics“, 9-„Electronic Journal of Linear Algebra“, 10-„ESAIM: COCV (Controle Optimisation et Calcul des Variations)“.

Atliekant analizę, buvo pastebėta, kad labai didelė dalis žurnalų yra talpinami bendruose žurnalo publikavimo tinklalapiuose. Ir daugelyje tokių tinklalapių esančių

Analysis“ žurnalas laikosi atviros politikos, o daugelis kitų šalių žurnalų tinklalapių didžiąją dalį funkcionalumo yra įkėlę į registruoto vartotojo ar redkolegijos dalis.

Lyginant „Mathematical Modelling and Analysis“ žurnalo tinklapį su kitų šalių žurnalų tinklalapiais, kurie turi individualius tinklapius (nesinaudoja bendru žurnalų publikavimu) pastebėta, kad juose visiems vartotojams prieinama tik pagrindinė žurnalo informacija, tokia kaip žurnalo redakcijos informacija, autorių informacija ir kt. Skirtingai nuo lietuviškų tinklalapių, kitų šalių tinklalapiuose yra įgyvendintos straipsnių paieškos sistemos. Tačiau daugelyje tinklalapių yra tik paprasčiausios paieškos sistemos, tokios kaip paieška pagal autorių, o paieškos pagal įvairius kriterijus dažniausiai yra tik bendruose žurnalų tinklalapiuose.

Indeksavimas

Indeksai naudojami greitai surasti eilutes su specifiniais stulpeliais. Be indeksų MySQL paiešką pradeda nuo pirmos eilutės ir iš eilės skaito visas lentelės eilutes, kol randa reikiamą. Kuo didesnė lentelė, tuo ilgiau bus atliekama paieška. Jei lentelė turi indeksą stulpeliui, pagal kurį vykdoma paieška, MySQL gali greitai nustatyti padėtį, kad paieška būtų atlikta duomenų failo viduryje, išvengiant viso failo skaitymo. Jei lentelė turi 1000 įrašų, tai su indeksais paieška trunka bent 100 kartų greičiau, nei skaitymas iš eilės (MySQL 6.0 Reference Manual 2008). Jei reikia apdoroti didžiąją dalį lentelės įrašų, greičiau yra skaityti duomenų failą iš eilės, nei naudoti indeksus. Dauguma MySQL indeksų (primary key, unique, index ir fulltext) yra laikomi B-medžio struktūroje. Išimtis yra specialaus tipo indeksams, kurie saugomi R-medžio struktūroje. Šie, specialaus tipo indeksai, skirti indeksuoti erdviniams laukams, t.y. kuriuose saugoma erdvinė informacija.

B-medis

Viena paprasčiausių ir praktikoje dažnai naudojamų medžio tipo struktūrų yra binarinis arba dvejetainis medis. Bet dažnai reikia medžio, kurio kiekviena viršūnė turėtų daugiau negu dvi šakas. Tokius medžius dar vadina daugiašakiais medžiais. Pagrindinė problema, su kuria susiduria duomenų bankų kūrėjai yra algoritmo efektyvumas paieškos greičio bei naudojamos atminties atžvilgiu. Daugiašakiai medžiai idealūs tokių uždavinių sprendimui, nes jei mes kreipiamės į vieną elementą, kuris saugojamas atmintyje, tai be jokių papildomų sąnaudų mes galime kreiptis ir į tam tikrą grupę elementų. Tai reiškia, kad medis suskaidytas į dalinius medžius, ir daliniai medžiai ir sudaro elementų grupes, kurios prieinamos vienu metu. Vienas daugiašakių medžių pavzdžių yra B-medžiai.

B-medžio struktūra naudojama, kai duomenys saugomi išorinėse laikmenose, o pagrindinis paieškos laikas sugaištamas kopijuojant informaciją iš lėtosios kompiuterio atminties į greitąją vidinę atmintį (Čiegis 2007). Formaliai B-medžiuose yra dviejų tipų viršūnės: vidinės ir išorinės. Tačiau jų realizacija panaikina šį skirtumą. Jei medis yra mažas, jo visos vidinės viršūnės gali tilpti vidinėje atmintyje. Šiuo atveju nuorodos į išorines viršūnes bus diskų ir jų puslapių adresai. Jei medis didelis ir vidinėje atmintyje netelpa, kiekvienai vidinei viršūnei (išskyrus medžio šaknį) galima priskirti atskirą puslapį, turintį savą struktūrą. Įrašų adresavimas bus kompleksinis: disko numeris ir puslapio numeris. Saugant diske viršūnes,

joms skirtuose puslapiuose yra saugomi ir visi jungtims reikalingi adresai, tačiau puslapiuose, saugančiuose įrašus, jungčių saugoti nereikia. Medžio šaknį verta laikyti vidinėje atmintyje, nes pradedant paiešką visada reikia formuoti reikalingus adresus. Aišku, B-medžio atminties kiekis priklauso nuo puslapių, įrašų dydžio ir nuorodų kiekio.

B-medžio savybės:

- kiekvienas puslapis, išskyrus šakninį, turi nuo n iki $2n$ viršūnių, o skaičius n yra vadinamas medžio eile (medžio šaknyje gali būti ir mažiau nei n elementų);
- puslapis yra arba lapas, arba turi $(m + 1)$ vaiką, čia m puslapio elementų skaičius; visų lapų lygiai yra vienodi;
- kiekviename puslapyje elementai išdėstomi didėjimo tvarka, tada paieškai B medyje, sudarytame iš N elementų, užtenka $\log_n N$ kreipinių.

Paieška B-medyje: Nagrinėkime B medžio atskirą fragmentą: puslapį $K = (k_1, k_2, \dots, k_m)$, kurio elementus k_j vadinsime raktais (angl. key) ir jo palikuonis, t.y. puslapius (angl. page) P_0, P_1, \dots, P_m . Kiekvieno puslapio elementai išdėstyti didėjimo tvarka, todėl teisingi raktų įverčiai:

$$k_1 < k_2 < \dots < k_m, \quad (1)$$

Svarbiausia B medžių savybė yra eiliškumas tarp puslapio K raktų ir jo palikuonių puslapiuose saugomų elementų:

$$P_0 < k_1 < P_1 < k_2 < P_2 < \dots < P_{m-1} < k_m < P_m, \quad (2)$$

Todėl vykdant elemento paiešką B medyje ir tikrinant K elementus, galimos tokios situacijos:

- radome raktą tarp K elementų, t.y. $x = k_j$, tada paiešką baigiame;
- teisingos nelygybės $k_j < x < k_{j+1}$, $1 \leq j < m$, tada paiešką tęsiame P_j puslapyje;
- teisinga nelygybė $x < k_1$, tada paiešką tęsiame P_0 puslapyje;
- teisinga nelygybė $x > k_m$, tada paiešką tęsiame P_m puslapyje (Čiegis 2007).

Turbo Boyer-Moore algoritmas

Boyer-Moore ypatingai efektyvus teksto paieškos algoritmas. Jis sukurtas Bobo Boyer ir J. Strother Moore 1977 metais. Boyer-Moore algoritmu nereikia apdoroti kiekvieno ieškomo teksto simbolio, nes kai kuriuos iš jų jis paprasčiausiai praleidžia. Bendrai kalbant, algoritmas greitėja ilgėjant ieškomam tekstui. Toks efektyvumas

pasiekiamas dėl to, kad po kiekvieno nesėkmingo bandymo ieškant atitikimų tarp ieškomo teksto ir teksto fragmento, yra gaunama informacija, kuri padeda išsiaiškinti kiek įmanoma daugiau pozicijų, kuriose ieškomo teksto tikrai negali būti.

Algoritmo veikimas: Boyer-Moore algoritmas lygina fragmentą su tekstu iš dešinės į kairę (pati paieška vykdoma iš kairės į dešinę). Jei tekstinis simbolis, kuris yra lyginamas su pačiu dešiniausiu fragmento simboliu, neatitinka fragmento, tada fragmentas „peršoka“ m pozicijų už šio simbolio.

Boyer-Moore algoritmo idėja gali būti pagrįsta tokiu pavyzdžiu: jei fragmento 10100111 8-a, 7-a ir 6-a pozicijos sutampa su sekos simboliais, bet 5-a pozicija nesutampa, iš karto fragmento analizė perkeliama per 7 pozicijas ir tikrinama nuo 15-os pozicijos. Šitokį sprendimą galima pagrįsti tuo, kad fragmento 6, 7, 8 pozicijos (t.y. fragmento dalis 111) niekur daugiau fragmente nesikartoja, todėl paties fragmento pirmose septyniose sekos pozicijose negali būti. Verta pastebėti, kad turbūt pagrindinis skiriamasis šio algoritmo bruožas yra tai, kad jo veiksmams valdyti yra naudojamas simbolių nesutapimo požymis.

Boyer-Moore paieškos algoritmas visada naudoja ne daugiau kaip $(M+N)$ simbolių lyginimų. Jam reikia apie N/M lyginimų, jei alfabetas nėra labai mažas, o fragmentas nėra labai ilgas.

Pastaroji savybė, aišku, yra euristinė. Ji pagrįsta pastebėjimu, kad jei simbolių alfabete yra daug, o fragmentas ne toks jau ilgas, daugelio simbolių arba nebus fragmente iš viso, arba jie pasikartos ne daugiau kaip vieną kartą (Juozapavičius 1997).

Išvados

1. Atlikus „Mathematical Modelling and Analysis“ žurnalo tinklapių palyginimą su kitų žurnalų tinklapiais matome, kad tinklapis yra vienas iš pirmaujančių Lietuvoje ir neatsiliekančių nuo užsienio bendrų žurnalų tinklapių. Tinklapis turi daugumą žurnalo tinklapiui reikalingų funkcijų. Pati svarbiausia tinklapių savybė - galimybė atlikti straipsnių paiešką per visus žurnalo tomus pagal įvairius kriterijus.

2. Kadangi „Mathematical Modelling and Analysis“ žurnalo duomenų bazė nėra labai didelė ir atsiradus naujam tomui bazė pasipildė tik 10-15-a įrašų (per metus išleidžiami keturi tomai, tai bazė pasipildys 40-60-a įrašų), todėl B-medžio paieškos algoritmas pilnai tinka atliekant straipsnių paiešką per visus tomus, B-medžio struktūroje saugant indeksuotus įrašus. Taip pat, kai B-

medis nėra didelis jo visos viršūnės gali tilpti vidinėje atmintyje ir yra nustatyta, kad darbo metu po įterpimų ir trynimų B-medžio užpildoma yra 50% (Juozapavičius 1997), taigi nelieka bereikalingų kreipimųsi į neužpildytas rodykles turinčius elementus.

3. Išnagrinėta, kad atliekant paiešką pagal žodžio fragmentą, MySQL duomenų bazėse naudojamas Boyer-Moore paieškos algoritmas, kuris neapdoruoja kiekvieno ieškomo teksto simbolio, o ieško pagal teksto šabloną, taigi straipsnių paieška ne pagal indeksuotus laukus taip pat yra labai efektyvi.

Padėkos

Dėkoju prof. habil. dr. Raimondui Čiegiui už pagalbą rengiant straipsnį.

Literatūra

- Boyer R.S. 1977. A Fast String Searching Algorithm. Communications of the ACM, vol. 20(10), New York.
- Bocek T., Hunt E., Stiller B., 2007. Fast Similarity Search in Large Dictionaries. Switzerland.
- Čiegis R., 2007. Duomenų struktūros, algoritmai ir jų analizė. Vilnius: Technika. 324 p. ISBN 978-9955-28-109-2.
- Juozapavičius A. 1997. Duomenų struktūros ir algoritmai [žiūrėta 2009 m. balandžio 28 d.]. Prieiga per internetą <<http://mif.vu.lt/cs2/lt/kursai/algorithm/files/dstruct.pdf>>.
- MySQL 6.0 Reference Manual [žiūrėta 2009 m. Balandžio 10 d.]. Prieiga per internetą: <<http://dev.mysql.com/doc/refman/6.0/en/index.html>>.

SCIENTIFIC JOURNAL'S WEB SITE DEVELOPMENT

I. Ivanauskaitė

Summary

„Mathematical Modelling and Analysis“ is an international journal on mathematical applications, numerical analysis and differential equations.

WEB site was created in 2001 year, but in the course of time arose a need of additional functions, such as several languages support, users interface upgrade and the ability to search articles through all journal's volumes.

Before refining „Mathematical Modelling and Analysis“ journal there were made the analysis and comparison with other journals WEB sites. In the analysis phase we paid a lot attention on WEB sites functionality, properties comparison and do not on articles presentation or recension steps.

Implementing search page it was found out that MySQL uses B-tree structure and in some cases Turbo Boyer-Moore algorithm for searching.

MATEMATIKOS MOKSLINIŲ STRAIPSNIŲ INFORMACINĖ ANALIZĖ

Jelena Lavcel

Vilniaus Gedimino technikos universitetas,
jelena.lavcel@gmail.com

Anotacija. Nagrinėjama, kaip atlikti matematikos srities mokslinių straipsnių informacinę analizę pagal temas, nustatyti ar du straipsniai yra tos pačios tematikos, išmatuoti panašumo lygį. Tokie uždaviniai atsiranda, kai reikia nustatyti tam tikros mokslinės publikacijos sritį, tarkime, atlikti mokslinio straipsnio recenzavimą. Vienas, taikytinų šiam uždaviniui spręsti įrankių, yra automatinis indeksavimas, kai kompiuteris ieško tekste raktinių žodžių. Dar vienas praktinis aspektas - sudaryti straipsnio charakteristiką, pavyzdžiui, raktinių žodžių rinkinį, kuris leistų ieškoti panašių straipsnių Internete arba duomenų bazėje.

Reikšminiai žodžiai: indeksavimas, tezasauras, entropija, Hemingo atstumas, raktiniai žodžiai.

Įvadas

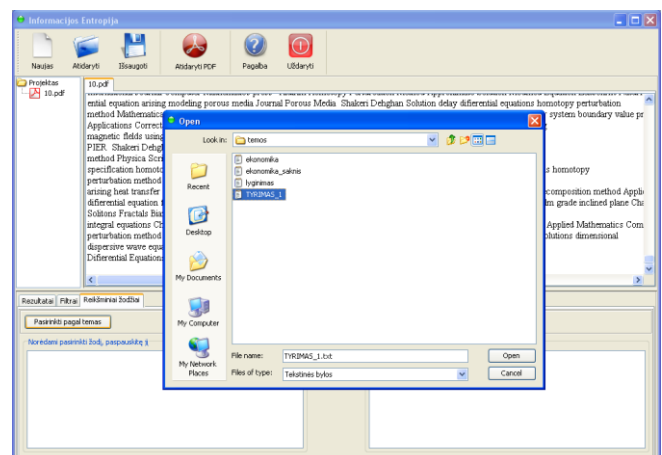
Šio darbo tikslas - atlikti matematikos srities mokslinių straipsnių informacinę analizę pagal temas, nustatyti ar du straipsniai yra tos pačios tematikos, išmatuoti panašumo lygį. Tokie uždaviniai atsiranda, pavyzdžiui, kai reikia nustatyti tam tikros mokslinės publikacijos sritį, tarkime, kad atlikti mokslinio straipsnio recenzavimą. Vienas, taikytinų šiam uždaviniui spręsti įrankių, yra automatinis indeksavimas, kai kompiuteris ieško tekste raktinių žodžių. Literatūroje nagrinėjami automatinio referavimo uždaviniai, kai programa sudaro santrauką. Dar vienas praktinis aspektas - sudaryti straipsnio charakteristiką, pavyzdžiui, raktinių žodžių rinkinį, kuris leistų ieškoti panašių straipsnių Internete arba duomenų bazėje.

Sukuriame pakankamai didelį raktinių žodžių žodyną (tezasaurą, angl. *thesaurus*), kuris sudaromas iš mokslo srities terminų, jame neturėtų būti bendrinės kalbos žodžių. Tačiau, priklausomai nuo mokslinių straipsnių specifikos, tie patys žodžiai gali būti tiek terminai, tiek ir specifinės kalbos bendriniai žodžiai, pavyzdžiui, žodis „analysis“ gali būti tiek matematikos terminas, tiek bendrinės kalbos žodis, priklausomai nuo konteksto. Vienas iš šio tyrimo tikslų - ištirti metodikos „stabilumą“, kai tezasauras sudarytas nelabai kvalifikuotai, t.y. „grubiai“. Šio žodyno sudaryme dalyvavo studentai, jie matematinius terminus, pasikartojančius kiekviename iš 22 straipsnių, kurių tematika apibūdinama raktiniais žodžiais: „homotopy perturbation method“. Žodyną sudaro 165 terminai.

Buvo skaičiuojami šių terminų ir jų variacijų dažniai. Tam panaudojome programas, kurias sukūrė Vilniaus Gedimino technikos universiteto studentai Lina Kulbytė, Romas Rachmanovas, Andrius Povilaitis, Violeta Kalinauskaitė ir Vytautas Sunklodas. Šios programos leidžia lyginti straipsnių informatyvumą, ir pabandyti įvertinti jų „artumo“ laipsnį.

Tyrimas

Prieš pradėdant dirbti su programa, reikia turėti, atskiruose TXT failuose surinktus ir surašytus pagal temas raktinius žodžius. Patalpiname TXT failus kataloge „Temos“, kadangi paspaudus mygtuką „Pasirinkti pagal temas“, programa atidarys būtent šį katalogą.



1 pav. Darbo su programos pavyzdys

Pasirinkus kurį nors TXT failą ir jį atidarius, kairiajame lange matome visus žodžius, kurie buvo tame faile. Norėdami pasirinkti žodį, reikia spustelėti ant jo arba paspausti mygtuką „Visus >>“ ir žodžius automatiškai perkels į dešiniąją lauką. Žodžiai esantys dešiniajame lauke bus nagrinėjami, tai yra tekstas bus analizuojamas pagal juos.

TXT failuose galima surašyti pilnus žodžius arba tik jų šaknis. Tai priklauso, ar norime analizuoti pagal tam tikrus tikslus žodžius, ar pagal įvairias žodžio variacijas.

Rezultatų lange yra pateikiami entropijų, tikimybių bei dviejų tekstų panašumo koeficientų skaičiavimo rezultatai.

Pasirinkus analizuojamus tekstus, kurie turi būti PDF formato ir raktinius žodžius, pagal kuriuos skaičiuojame rezultatus, skaičiuosime jų panašumo koeficientus.

Taip pat programa skaičiuoja, tiek vieno, tiek kito teksto žodžių dažnį, raktinių žodžių dažnių sumą ir bendrą sistemos entropiją.

Iš suskaičiuotų x_k dažnių sudarome kiekvienam straipsniui vektorių $\vec{x} = (x_1, x_2, \dots, x_n)$. Gauname diskretųjį informacijos šaltinį ir apskaičiuojame entropiją (matuojama *natais*). Gautus duomenys normuojame: t.y. sudarytą vektorių padaliname iš rastų žodžių sumos $\sum_{k=1}^n |x_k| = d$ ir skaičiuojame straipsnių „atstumus“ vieną iki kito pasinaudojus Hamingo atstumų formule

$$\|\vec{x} - \vec{y}\| = \sum_{k=1}^n |x_k - y_k|, \quad (1)$$

čia $\vec{x} = (x_1, x_2, \dots, x_n)$ - vieno straipsnio santykinų dažnių vektorių; $\vec{y} = (y_1, y_2, \dots, y_n)$ - kito straipsnio santykinų dažnių vektorių.

Kitas tyrimo žingsnis yra entropijos skaičiavimas. Apskaičiuojame santykinus dažnius

$$p_k = \frac{d_k}{d}, \quad (2)$$

čia: d_k - žodžių dažniai; d - dažnių suma, kurią skaičiuojama pritaikius formulę $d = \sum_{k=1}^n d_k$.

Toliau apskaičiuojame entropiją pagal formulę

$$E = -\sum_{k=1}^n p_k \cdot \ln p_k, \quad (3)$$

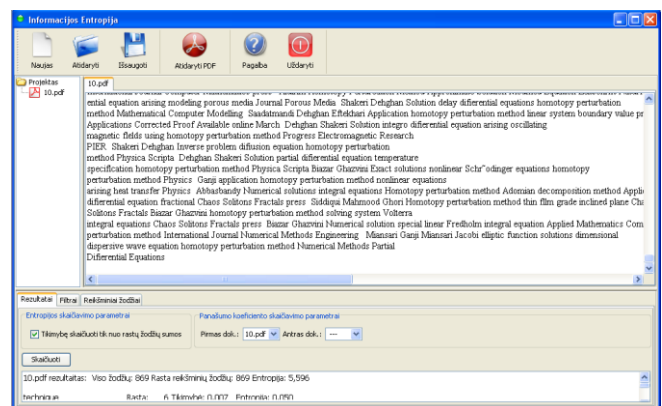
čia: p_k - santykinis dažnis; E - entropija.

Jei žodžių beveik nesutikome, gausiname mažą entropiją. Entropijos maksimumas galimas, kai turime $\frac{1}{n}$.

Jei imame kitus straipsnius, kurių vektorių toks pat, tai jie yra identiški, tai yra gauname tiek pat informacijos. Mūsų atveju entropija yra rodiklis, kai gauname informaciją apie straipsnį.

Eksperimentas

Tyrimui paėmėme 22 straipsnius PDF formato, iš kurių studentai išrinko raktinius žodžius, kuriuos surinkome TXT faile ir patalpinome kataloge „Temos“. Pasinaudojame programa ir išfiltravome tekstą, tai yra formules, tarpus, skaičius, vieno, dviejų ir trijų simbolių žodžius, neperskaitytus simbolius pakeitėme į atitinkamus. Suradome kiekvieno straipsnio terminų ir jų variacijų skaičių ir dažnius nuo rastų žodžių sumos.



2 pav. Rezultatų vaizdavimas programoje

Duomenys patalpinome į Excel failą (1 lentelė), apskaičiuojame santykinį dažnį pagal (2) formulę, sudarėme kiekvienam straipsniui vektorių, sudarytus iš dažnių, normavome juos ir apskaičiuojame kiekvieno straipsnio entropiją, pasinaudojus (3) formule.

1 lentelė. Dalis duomenų lentelės iš Excel failo

tensor						12					4			4	1	11							
term	1		4		1	15		2	7	6	9	2	2	2	5	15	3	2	6	2	1	10	
theoretical						1			1	2					1	1		1	1				
theory	4	2			1	11		1	1	4		1	7	2	3	10			3			1	
thermocline														2									
topology								1	2						2	1		1	2		1	1	
trial										1													
type					1			2	4						1	3	1			1	3		1
unaffected																							
unity	2		1					1	1	1				2			1		2	1	1		
unknown				2	1	1		1	1	2					2	1					2		
value	4		1			15	5	4	1	3	5	2	7	8	2	15	1	16	1	28	2	7	
variable	4	3				3		1	2	4			2		6	3	21	1					2
variation		2		1	9		1	13	1	40	3	2	2	3	22		1	1				4	4
various			2	2		1	2	1		2		1			3	1	1	2	1	1			8
vector	3											1			1								
zero			3			7		1	1	1	2	2	4	5			9	1	7	3	1	1	
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	
d =	218	205	266	110	134	551	413	376	338	862	431	221	302	368	615	552	598	405	364	473	399	357	
entropija (straipsnio) =	<i>3,3537</i>	<i>3,2667</i>	<i>3,4918</i>	<i>3,5006</i>	<i>3,5665</i>	<i>3,7375</i>	<i>3,3798</i>	<i>3,8227</i>	<i>3,8257</i>	<i>3,8586</i>	<i>3,6172</i>	<i>3,2553</i>	<i>3,6346</i>	<i>3,5237</i>	<i>4,0336</i>	<i>3,7765</i>	<i>3,5072</i>	<i>3,7581</i>	<i>3,7193</i>	<i>3,5052</i>	<i>3,6694</i>	<i>3,7122</i>	
	66%	64%	68%	69%	70%	73%	66%	75%	75%	76%	71%	64%	71%	69%	79%	74%	69%	74%	73%	69%	72%	73%	

Kitas tyrimo žingsnis normuoti vektorių ilgus ir apskaičiuoti straipsnių „atstumus“ vieną nuo kito, pasi-
naudojus (1) formule. Gauname 2 lentelę

2 lentelė. Atstumų tarp straipsnių lentelė

	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22	kitos tematikos	filosofija	geografija	
1	0,88	1,09	1,24	1,36	1,12	1,06	1,14	1,1	1,2	1,2	1,3	1,1	1,2	1,2	1,1	1,1	1,1	1,0	1,1	1,1	1,1	1,8	1,6	1,7	
2		1,03	1,08	1,36	1,22	0,93	0,99	0,89	0,9	1,1	1,1	1,1	1,0	1,1	1,2	1,0	0,9	1,1	1,1	0,9	1,1	1,8	1,6	1,7	
3			0,79	1,03	1,34	0,64	0,69	0,71	0,8	0,7	0,9	0,9	0,7	0,8	1,4	0,8	0,7	0,5	0,8	0,6	0,9	1,8	1,5	1,5	
4				0,82	1,39	0,92	0,83	0,96	0,9	1,0	0,9	1,1	1,1	0,9	1,4	1,0	1,1	0,9	1,1	0,9	0,9	1,8	1,6	1,5	
5					1,52	1,18	0,94	1,11	1,0	1,2	1,1	1,2	1,2	0,9	1,5	1,2	1,3	1,0	1,2	1,1	1,0	1,8	1,6	1,7	
6						1,36	1,24	1,26	1,3	1,3	1,2	1,1	1,2	1,3	0,1	1,4	1,1	1,2	1,3	1,2	1,3	1,7	1,5	1,4	
7							0,65	0,7	0,7	0,8	1,0	0,9	0,9	0,9	1,4	0,5	0,8	0,9	0,7	0,6	1,0	1,8	1,5	1,6	
8								0,63	0,6	0,8	0,9	0,9	0,8	0,5	1,3	0,8	0,7	0,8	0,9	0,5	0,9	1,8	1,5	1,5	
9									0,6	0,8	1,0	0,9	0,8	0,7	1,3	0,6	0,8	0,8	0,8	0,5	1,0	1,7	1,5	1,6	
10										0,8	0,9	1,0	0,9	0,5	1,3	0,6	0,9	0,9	0,8	0,7	0,9	1,8	1,5	1,6	
11											1,0	1,1	0,7	0,9	1,3	0,9	0,8	0,8	0,8	0,8	0,7	1,9	1,5	1,5	
12												1,0	1,0	1,0	1,2	1,0	1,1	1,0	1,0	0,9	0,9	1,8	1,6	1,6	
13													0,9	1,0	1,1	0,9	0,9	0,9	0,9	0,9	1,0	1,8	1,6	1,6	
14														1,0	1,2	0,9	0,5	0,8	0,7	0,8	0,9	1,8	1,6	1,6	
15															1,2	0,9	0,9	0,9	1,0	0,7	0,9	1,8	1,4	1,5	
16																1,4	1,1	1,2	1,3	1,3	1,3	1,7	1,5	1,4	
17																		0,9	0,9	0,8	0,7	1,0	1,8	1,6	1,7
18																			0,7	0,7	0,7	0,9	1,8	1,5	1,5
19																			0,9	0,7	0,9	1,8	1,5	1,4	
20																				0,8	1,0	1,7	1,5	1,5	
21																					1,0	1,7	1,4	1,6	
22																						1,8	1,5	1,5	
kitos tematikos																							1,7	1,6	
filosofija																								1,2	

Matome, kad mažiausias atstumas (lygus 0,1) tarp 6 ir 16 straipsnių. Abu straipsniai yra to paties autoriaus, vienas parašytas 2007 m. rugpjūčio 24 d., o kitas yra pataisytas ir paruoštas spausdinimui 2008 m. vasario 5 d. Taip pat pastebime, kad didžiausias atstumas yra tarp 5 ir 6 straipsnių ir lygus 1,5.

Kontrolei paėmėme 3 straipsnius: matematikos iš kitos temos, filosofijos ir geografijos. Atlikome tuos pačius tyrimus ir gavome pateiktus 3 lentelėje rezultatus. Skaičiai įdomūs tuo, kad hipotezė pasitvirtino. Iš lentelės matome, kad matematikos ir geografijos straipsnių entropija sumažėjo (3 lentelė), o atstumai iki kitų straipsnių padidėjo (2 lentelė). Išimtis yra filosofijos srities straipsnis, kurio entropija yra $\approx 70\%$ nuo maksimalios reikšmės,

bet atstumai iki kitų straipsnių lieka dideli (2 lentelė). Kol kas norėjome išbandyti metodiką ir gilesnė analizė yra būsimųjų tyrimų objektas.

Pastebime, kad žodyne yra tokių žodžių, kurie matematikoje yra terminai, o kitur gali būti naudojami, kaip bendrinės kalbos žodžiai.

Iš žodyno išrenkame dažniausiai pasikartojančius žodžius. Sudarėme TXT failą ir patalpiname kataloge „Temos“. Pasinaudojame programa ir suradome kiekvieno straipsnio terminų ir jų variacijų skaičių, ir dažnius. Duomenys patalpiname į Excel failą, skaičiavome santykinį dažnį nuo rastų žodžių sumos, sudarėme kiekvienam straipsniui vektorius, sudarytus iš dažnių, apskaičiavome entropiją.

3 lentelė. Dažniausiai pasikartojančių žodžių Excel duomenų lentelės fragmentas

nonlinear	9	1	22	10	11	3	22	25	16	30	23	26	14	12	39	3	34	8	29	15	25	28
numerical	4		4	1	5	3	10	3	1	22	4	6	2		10	3	3	3	4	14	4	10
obtain	3	13	10	6	3	35	2	9	8	32	18	12	6	11	22	32	6	4	10	6	13	12
parameter	3		6			4	1	5	2	3	19	3	1	15	5	3	6	17	10	6	7	1
part		1		1		3	10	4	1	10				1	6	3	24	3	1	1	3	2
perturbation	17	14	21	4		3	35	14	21	38	25	2	2	41	13	2	41	27	21	38	26	1
polymer						39			1	1					1	38						
problem	1	2	8	2	2	10	4	5	1	11	16	4	6	28	10	10	7	21	17	32		16
result	11	3	5	3	1	35	11	10	5	1	3	5	4	3	1	34	5	5	5	7	8	4
science			6	2	1		9	4	3	7	2		10		10	1	2	4	13	2	4	2
series	24	21	3			1	2	3	2	2	3		1	4	1	1		3	4	1	2	8
solution	10	12	15	7	5	15	27	19	22	56	23	31	25	27	28	13	42	24	14	35	13	30
solv -	3	2	6	1	4	12	11	7	7	13	5	7	12	8	14	12	29	9	6	13	12	4
technique			7			5	5	4	6	7	1		12	5		6	9	7	6	7	1	
term	1		4		1	15		2	7	6	9	2	2	2	5	15	3	2	6	2	1	10
value	4		1			15	5	4	1	3	5	2	7	8	2	15	1	16	1	28	2	7
variation		2		1	9		5	13	1	40	3	2	2	3	22		1	1			4	4
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
d =	158	167	202	77	86	340	334	275	233	639	322	166	212	287	407	335	440	292	249	378	295	246
entropija =	2,7980	2,8509	2,8317	2,8593	2,7957	2,6687	2,8849	3,0530	2,9533	2,9606	2,8717	2,6263	2,8110	2,8138	3,0631	2,6727	2,7708	2,9707	2,9477	2,8497	2,9610	2,9829
	80%	82%	81%	82%	80%	76%	83%	87%	84%	85%	82%	75%	80%	80%	88%	76%	79%	85%	84%	82%	85%	85%

Toliau normavome vektorių ilgius ir apskaičiavome straispnių „atstumus“ vieną nuo kito. Gavome tokius rezultatus:

4 lentelė. Dažniausiai pasikartojančių žodžių atstumai iki straispnių

	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	22
1		0,7	0,9	1,1	1,2	1	0,9	1	0,9	1	1	1,1	1	1	1,1	1,1	0,9	1	0,9	1	0,9	1
2			0,9	0,8	1,2	1,1	0,8	0,8	0,7	0,7	0,9	0,9	0,9	0,8	0,8	1,2	0,8	0,8	0,9	0,9	0,7	0,9
3				0,6	0,9	1,2	0,5	0,5	0,5	0,6	0,6	0,7	0,7	0,6	0,5	1,2	0,5	0,6	0,3	0,7	0,5	0,7
4					0,7	1,3	0,7	0,6	0,7	0,7	0,8	0,7	0,8	0,9	0,6	1,3	0,8	0,9	0,7	0,9	0,7	0,7
5						1,4	1	0,7	1	0,8	1	0,8	1	1,1	0,6	1,4	1	1,1	0,9	1,1	0,9	0,8
6							1,2	1,2	1,2	1,2	1,3	1,2	1	1,2	1,2	0,1	1,3	1	1,1	1,2	1,2	1,3
7								0,4	0,4	0,5	0,7	0,8	0,7	0,7	0,7	0,6	1,3	0,4	0,7	0,6	0,6	0,4
8									0,4	0,4	0,6	0,7	0,7	0,7	0,7	0,3	1,2	0,6	0,6	0,5	0,8	0,4
9										0,4	0,6	0,8	0,7	0,7	0,7	0,5	1,2	0,3	0,7	0,6	0,7	0,3
10											0,7	0,7	0,8	0,8	0,3	1,2	0,5	0,8	0,7	0,7	0,6	0,8
11												0,8	0,9	0,6	0,8	1,3	0,7	0,7	0,6	0,7	0,7	0,6
12													0,8	0,9	0,7	1,1	0,8	1	0,8	0,8	0,8	0,7
13														0,8	0,7	1	0,7	0,7	0,7	0,7	0,7	0,8
14															0,9	1,3	0,8	0,4	0,6	0,5	0,7	0,8
15																1,2	0,6	0,8	0,6	0,9	0,5	0,7
16																	1,3	1,1	1,1	1,3	1,2	1,3
17																		0,8	0,7	0,7	0,5	0,9
18																			0,5	0,5	0,7	0,8
19																				0,8	0,5	0,7
20																					0,8	0,9
21																						0,9
22																						

Laikome, kad žodynas sudarytas patenkinamai. Tuomet kitas klausimas, kaip sukurti kriterijų lyginti du

straipsnius. Ar jie artimi, ar skiriasi pagal problematiką, stilių, autorius ir pan. G.Salton siūlė įvairias metrikas.

Mes patikriname:

- simetriškumą
- trikampio nelygybę
- parodysimė, kad maksimumas lygus 2.

Tarkime, kad turime trikampio nelygybę:

$$\sum_{k=1}^n |x_k - y_k| \leq \sum_{k=1}^n |x_k| + \sum_{k=1}^n |y_k| \quad (4)$$

čia: $x_k = (x_1, x_2, \dots, x_n)$; $y_k = (y_1, y_2, \dots, y_n)$

Kadangi $0 \leq x_k \leq 1$ ir $0 \leq y_k \leq 1$, tai $\sum_{k=1}^n |x_k| = 1$ ir

$\sum_{k=1}^n |y_k| = 1$. Gauname, kad (4) nelygybė turės pavidalą:

$$\sum_{k=1}^n |x_k - y_k| \leq 1 + 1 = 2$$

Pavyzdžiui, paimame $x(1, 0, \dots, 0)$ ir $y(0, 1, \dots, 0)$.

Tuomet, įstačius į (4) nelygybę, gauname:

$$\sum_{k=1}^n |x_k - y_k| \leq \sum_{k=1}^n |x_k| + \sum_{k=1}^n |y_k| = 1 + 1 = 2$$

Įrodėme, kad maksimalus ilgis tarp straipsnių yra lygus 2.

Atstumas visur yra nedidelis, išskyrus keletą vietų, kurie yra didesni. Metodika yra verta dėmesio ir galime ją bandyti tobulinti.

Išvados

1. Norėdami tobulinti metodiką, reikia daugiau dėmesio skirti žodyno sudarymui.

2. Reikia atsižvelgti į žodžių kombinacijas, praplešti žodyną. Tas pats žodis gali būti nagrinėjamas kaip terminas, kitur kaip bendrinės kalbos specialusis žodis. Žodyno sudarymas reikalauja tolimesnių tyrimų.

3. Turime atlikti eksperimentus su kitomis metrikomis, skirtumu tarp straipsnių matas. Tai irgi yra tolimesnių eksperimentų objektas.

Padėkos

Dėkoju prof. dr. Aleksandrui Krylovui už pagalbą rengiant straipsnį.

Literatūra

Солтон, ДЖ. 1979. Динамические библиотечно-информационные системы. Москва [Salton, G. 1975. Dynamic informatin and library processing. New Jersey]

Сэлтон, Г. 1973. Автоматическая обработка, хранение и поиск информации. Москва. [Salton, G. 1968. Automatic information organization and retrieval. London.]

Luhn, H.P. 1957. A Statistical Approach to mechanized encoding and searching of library information. IBM Journal of Research and Development 1, No. 4, October 1957, 309-317.

Luhn, H.P. 1961. The automatic derivation of information retrieval encodements for machine readable texts, in infor- mation retrieval and machine translation. Part 2, ed. A.Kent, New York: Interscience Publishers.

Salton, G. 1989. A comparison of book indexing methods. TR89-1033, August 1989, New York.

Gudivada, V.N.; Raghavan, V.V.; Grosky, W.I.; Kasanagottu, R. 1997. Information retrieval on the World Wide Web. Vo- lume 1, Issue 5, Sep/Oct 1997, 58 – 68.

ARTICLE TITLE IN SUMMARY LANGUAGE

J. Lavcel

Summary

Examining, how it is possible to do the informative analysis of mathematical text on themes, to define whether there are two articles of one theme to define the degree of coincidence. Such tasks appear, when it is necessary to define the theme of publication, for example, to do a review. One of methods of decision is automatic indexation, when a computer searches keywords in a text. Another task - to do description of text, which would allow to find the look articles like in the Internet or in a database.

ŽENKLŲ KALBOS KURTIESIEMS ANIMAVIMAS

Justas Jocius¹, Edvard Michnevič²

Vilniaus Gedimino technikos universitetas,

el. p. ¹ justas.jocius@fm.vgtu.lt; el. p. ² edvard.michnevic@fm.vgtu.lt;

Anotacija. Straipsnyje aptariama lietuvių kalba kurtiesiems. Nagrinėjami žmogaus rankos biomechanikos principai ir jos virtualaus modelio animavimo galimybės. Aptarti keli rankos modeliavimo būdai, bei pasirinktas labiausiai tinkantis. Apžvelgiamas virtualaus rankos modelio formavimas ir animavimas AutoDesk Maya programine įranga. Aptariamas gautų rezultatų naudojimas, kaip mokomoji medžiaga norintiems išmokyti kurčiųjų kalbą.

Reikšminiai žodžiai: Kurčiųjų kalba, rankos biomechanika, kinematika, animacija.

Įvadas

Ranka yra svarbi žmogaus kūno dalis. Ji kasdieniniame gyvenime atlieka tokius veiksmus, kaip daiktų paėmimas, komunikavimas ir begalę kitų. Panašias funkcijas žmogaus ranka atlieka ir virtualioje aplinkoje. Šioje aplinkoje pagrindinis uždavinys rankos modeliavimą ir animaciją padaryti kuo tikroviškesniais ir išgauti kuo tikslesnius judesius. Virtualus rankos modelis naudojamas tiek komunikavimui, tiek kompiuteriniuose žaidimuose, mediciniame modeliavime ir t. t. Kaip bebūtų, rankos modeliavimas reikalauja didelio žinių „bagažo“, tai yra modeliavimo priemonių išmanymo, rankos kompleksinės sudėties žinojimo. Rankos skeleto judėjime galime išskirti daug laisvės laipsnių, kas parodo viso modelio sudėtingumą. Kitas virtualios rankos formavimo žingsnis yra jos dalių judėjimo taisyklių ir parametrų nustatymas. Šias žinias apie rankos judėjimą, galime pavadinti paprasčiausiai rankos anatomija arba rankos biomechanikos principais, kuriuos ir aptarsime šiame straipsnyje.

Ženklų kalbos modeliavimas ir animavimas

Ženklų kalba (arba gestų kalba) [1, 2] – kalba, kuri suvokiama vizualiai ir perduodama rankų bei kūno gestais. Gestai žymi raides, sąvokas, o nacionalinės gestų kalbos turi mažai ką bendro su tose šalyse vartojamomis žodinėmis kalbomis.

Ženklų kalba nėra tarptautinė. Ji nacionalinė, nes yra gyva ir nuolat kintanti. Tačiau gestų kalbos gali būti giminiškos arba labai tolimos. Pavyzdžiui, lietuvių gestų kalboje yra daug sąsajų su rusų, tačiau ji labai skiriasi nuo kinų gestų kalbos. Egzistuoja ir tarptautinė gestų kalba, ji

reikalinga kurtiesiems bendraujant įvairiuose tarptautiniuose renginiuose.

Šiame darbe mums yra aktualiausias dvi sąvokos: *daktikologija* – pirštų (daktilinė) kalba, t. y., bendravimas vartojant pirštų ženklus ir *daktilinė abėcėlė* – tam tikra tvarka išdėstytų pirštų ženklų visuma. Daugumai daktilinių abėcėlių būdinga, kad ženklas atitinka tam tikros raidės grafinį vaizdą. Daktilinė kalbos forma neleidžia suvokti visą žodį ar sakinį vienu metu, kalbant daktikologija matomi atskiri, erdvėje rodomi ženklai iš kurių mintyse reikia sudėti žodį. Daktilinio kalbėjimo tempas 2,5 karto lėtesnis negu sakinio kalbėjimo. Daugelio šalių daktilines abėcėles galima rasti internete[3].

Lietuvių daktilinę abėcėlę sudaro 32 ženklai. Daugelis jų panašūs į lietuvių kalbos spausdintines raides

Žemiau pateiktame paveikslyje (1 pav.) galime pamatyti lietuvių, daktilinę abėcėlę [1, 2].

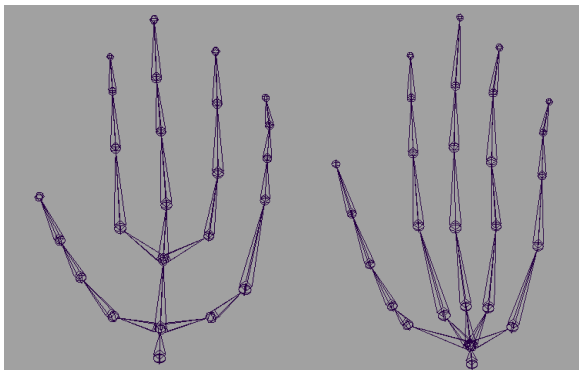


1 pav. Lietuvių kalbos daktilinė abėcėlė

Žmogaus rankos skeleto anatomija

Toliau aptarsime žmogaus rankos skeleto anatomiją, kad susipažinti su judančiomis rankos jungtimis.

Įvairių virtualių rankos modelių galime nemokamai parsisiųsti iš internetinių svetainių, arba įsigyti iš 3D grafikos kūrimo kompanijų. Dažniausiai rankos modelis vaizduojamas kaip nepertraukimas daugiakampių tinklis, kuris gali būti importuotas į skirtingus 3D programinės įrangos naudojamus formatus. Ir kaip tik modelis importuojamas, tada jam jau galima kurti ir integruoti skeleto modelį. Paskutinis žingsnis – to modelio animavimas. Dažniausiai rankos animacijoje naudojami dviejų tipų modeliai [4,5]: 24 jungčių modelis ir 26 jungčių modelis, kurie pavaizduoti paveikslėlyje (2 pav.):



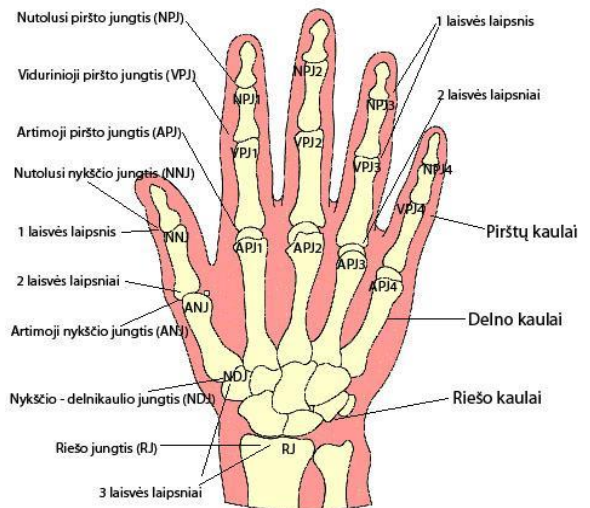
2 pav. Skeleto 24 ir 26 jungčių modeliai

Pirmojo modelio privalumas yra tas, kad turi papildomą delno jungtį, kuri leidžia geriau perteikti plaštakos judėjimą. O antrasis modelis neturi papildomos riešo jungties, tačiau turi atskiras nuo riešo smulkiųjų riešo kaulų jungtis ir papildomą nykščio jungtį. Antrasis modelis leidžia mums išgauti žymiai natūralesnį viršutinės rankų dalies (pirštų) judėjimą ir ypač nykščio. Kadangi mus aktualesnis yra pirštų judėjimo realistiškumas, tad mes ir aptarsime plačiau kaip sudaromas antrasis rankos modelis.

Žmogaus rankos skeletas susideda iš 27 kaulų [6-9]. Šie kaulai gali būti padalinti į tris grupes:

- 8 riešo kaulai;
- 5 delno kaulai;
- 14 pirštų kaulų.

Savaime suprantama, kad riešo kaulus galime rasti rieše, delnakaulius plaštakoje ir pirštikaulius pirštuose. Jungtys (sąnariai) tarp šių kaulų juda tiek, kiek turi laisvės laipsnių. Laisvės laipsnių skaičius jungtyje yra lygus ašių sukimosi krypties skaičiui. Toliau pateiktame paveikslėlyje (3 pav.) matome prieš tai minėtus kaulus, jų jungtis ir laisvės laipsnius jungtyse:



3 pav. Rankos modelis

Kaip matome paveikslėlyje, artimąja piršto jungtimi (APJ) bus vadinamas pirmas piršto sąnarys (krumplys). Kol dar nenutolome nuo rankos biologinio modelio reikėtų aptarti terminus kurie naudojami literatūros šaltiniuose [7, 8] apibūdinant rankos judėjimą:

Fleksija ir Ekstencija (toliau - f/e). Fleksija, tai jungties lenkimas į priekį, arba atgal, o ekstencija (ištiesimas) tai atgalinis veiksmas fleksijai norint atstyti jungtį į pradinę padėtį.

Abdukcija ir Adukcija (toliau - a/a). Abdukcija, tai jungties lenkimas į šonus (kairę ir dešinę puses), o adukcija (ištiesimas) tai atgalinis veiksmas abdukcijai norint atstyti jungtį į pradinę padėtį. Tokiais pat terminais nusakomas visų pirštų judėjimas įskaitant ir nykštį.

Visi pirštai išskyrus nykštį turi tokią pačią esminę struktūrą [9, 10]:

- Nutolusi piršto jungtis (NPJ) turi 1 laisvės laipsnį;
- Vidurinioji piršto jungtis (VPJ) taip pat turi 1 laisvės laipsnį;
- Apatinė piršto jungtis (APJ) turi 2 laisvės laipsnius.

Kiekviena iš šių jungčių gali būti tiek lenkiama (fleksija), tiek tiesiama (ekstencija). Tačiau tik apatinė piršto jungtis (APJ) dar gali būti ir lenkiama (adukcija) į šonus ir tiesiama (abdukcija) atgal. Reikėtų pažymėti, kad visas rankos judėjimas vykdomas judinant plaštakos, pirštų kaulus ir riešo kaulus.

Nykščio struktūra yra kitokia nei kitų pirštų, pradedant nuo to, kas nykštys neturi vidurinėsios piršto jungties:

- Riešo – delnakaulio (RDJ) jungtis turi 3 laisvės laipsnius;
- APJ – 2 laisvės laipsnius;
- Ir NPJ turi 1 laisvės laipsnį.

RDJ jungtis turi kaip ir APJ gali būti lenkiama į priekį (f/e), atgal ir į šonus (a/a), tačiau dar papildomai RDJ jungtis gali sukstis aplink delnakaulį. NPJ jungtis kaip ir kituose pirštuose gali būti lenkiama tik į priekį ir atgal.

Ir paskutinė riešo jungtis (RJ) kaip ir RDJ jungtis turi 3 laisvės laipsnius ir gali judėti visomis kryptimis.

Viso rankos modelis turi 26 laisvės laipsnius. Toliau panagrinėsime šį skaičių kaip modelio apribojimą.

Rankos modeliavimas koordinacinių sistemoje pagal Denavitą-Hartenburgą (Denavit-Hartenburg)

Denavitas ir Hartenburgas [11-13] išstobulino modelio sistemos judėjimą koordinacinių ašyje į veiksmų seką kinematinėje ryšių grandinėje. Po to kai šis metodas buvo patvirtintas, tapo įmanoma naudoti standartines transformacijas judinant modelį koordinacinių sistemoje. Šios transformacijos aprašyti yra naudojami tik keturi parametrai (lentelė):

Lentelė. Piršto jungčių transformacijos matricos parametrai

Parametro aprašymas	Žymėjimo simbolis
Jungties kampas	α
Sujungimo ilgis	l
Jungties diametras	d
Jungties sukimosi kampas	θ

Šie keturi parametrai naudojami transformacijos matricoje A (1 ir 2 formulės), kuri sudaroma taip:

$$A = \text{Rot}(z, \theta) \times \text{Trans}(0,0,d) \times \text{Rot}(x, \alpha); \quad (1)$$

Čia: A – transformacijos matrica; Rot(z,θ) – rotacija z ašyje, θ kampu; Trans(0,0,d) – jungties pločio d transformacija; Rot(x,α) – rotacija x ašyje, α kampu;

$$A = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \times \cos(\alpha) & \sin(\theta) \times \sin(\alpha) & l \times \cos(\theta) \\ \sin(\theta) & \cos(\theta) \times \cos(\alpha) & -\cos(\theta) \times \sin(\alpha) & l \times \sin(\theta) \\ 0 & \sin(\alpha) & \cos(\alpha) & d \\ 0 & 0 & 0 & 1 \end{bmatrix}; \quad (2)$$

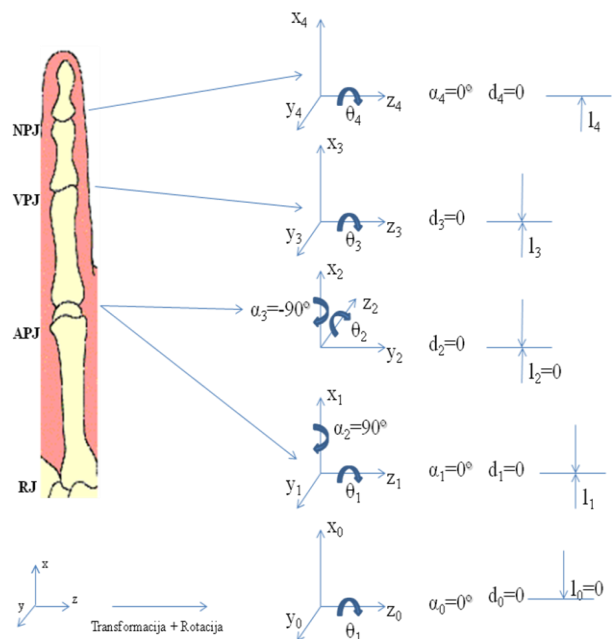
čia: A – piršto jungčių koordinacinių sistemoje transformacijos matrica; l – piršto sujungimo (kaulo) ilgis;

Kiekvienos naujos modelio dalies sistema koordinacinių ašyje bus susieta su ankstesne (4 pav.), kol matrica bus užpildyta visais keturiais parametrais tai modelio sistemos daliai. Šių parametų parinkimas turi būti atliktas pagal Denavito-Hartenburgo taisyklę. Pagrindiniai šio pasirinkimo proceso reikalavimai kurie padės sudaryti visą modelį bus aprašyti toliau. Daugiau apie galima pasiskaityti šiame straipsnyje [11].

- Z ašis turėtų būti aprašoma kaip sukimosi ašis;
- X ašis turėtų būti statmena Z ašiai;
- Y ašis turėtų būti priskirta kaip sankirta tarp X ir Z ašių.

Visoms rankos modelio dalims koordinacinių sistemoje tik vienas parametras bus modifikuojamas po modelio sukūrimo [12]. Tai bus sąryšio sukimosi kampas θ. Iš to seka, kad modelio jungtims su daugiau nei vienu laisvės laipsniu reikalingos sudėtingesnės koordinacinių sistemos. Pirmas toks koordinacinių rinkinys padalintos jungties (turima omenyje, dalį kuri arčiau riešo) turės 1 parametą arba sąryšio ilgį lygų 0. Tiktai paskutinė sistemos dalis susieta su padalinimo sąryšiu turės ryšio ilgį, kuris atspindės tikrą pasirinkto piršto ilgį. Pavyzdžiui artimoji piršto jungtis (APJ) yra modeliuojama iš dviejų dalių koordinacinių sistemų. Pirmą turi l=0, o antrojoje l yra lygus pirštikaulio tarp APJ ir VPJ jungčių ilgiui pasirinktame piršte.

Paveikslėlyje (4 pav.) matome kaip piršto jungtys paskirstytos koordinacinių sistemoje. Kaip ir ankščiau buvo minėta visi pirštai turi tokią pat esminę struktūrą, taigi ši sistema yra tinkama visiems pirštams.



4 pav. Piršto modeliavimas koordinacinių sistemoje

Kaip matome iš paveikslėlio (4 pav.), koordinacinių sistemoje jungtys yra susietos riešo jungtimi (RJ) ir transformuojamos pašalinės koordinacinių sistemos su pervedimais ir rotacijomis.

Koordinacinių sistemų paskirstymas nykščio jungtims yra toks pats. Tačiau dėl RDJ esančių 3 laisvės laipsnių visa sistema tampa truputi sudėtingesnė. Iš viso reikia 5 koordinacinių sistemų, kad sumodeliuoti visą rankos modelį. Ir sujungti viską riešo jungtyje.

Rankos modelio realizavimas

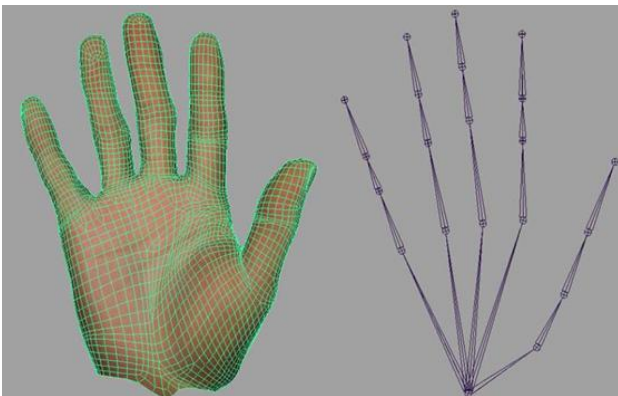
Jungčių judėjimo priklausomybės ir ribojimai

Rankos jungtys turi įvairių vidinių ir išorinių ribojimų ir juos galime išskirstyti į tris pagrindines grupes [7, 10, 13]:

- Vidiniai jungčių ribojimai
- Piršto jungčių tarpusavio priklausomybės
- Jungčių tarp pirštų priklausomybės ir ribojimai

Nurodytuose literatūros šaltiniuose buvo nagrinėjami labai skirtingi ribojimų parametrai, jų reikšmės, bei priklausomybės. Tačiau ne visi jie buvo tinkami mūsų rankos modeliui, todėl mes kiek įmanoma pamėginome patys suderinti ir pasirinkti individualius rankos ribojimus, kurie mūsų manymu suteikė daugiausiai realistiškumo rankos modeliui. Visą tai mes atlikome eksperimentų pagalba.

Mūsų naudojamą skeletą sudaro 21 jungtis, įskaitant pirštų galų pagalvėles ir riešo jungtį. Skirtingai, negu pradžioje aptartuose modeliuose nusprendėme atsakyti kiekvienam pirštui atskirų riešų jungčių, kadangi tai būtų netikslinga. Atskiros riešo jungtys nepriduos mūsų rankos modeliui labai pastebimo modelio judėjimo pagerėjimo, tačiau gerokai pasunkintų paties rankos modelio modeliavimą, ir animacijos procesą. Rankos skeleto modelis pateiktas paveikslėlyje (5 pav.):



5 pav. Maya rankos ir skeleto modeliai

Vidiniai jungčių ribojimai

Žmogaus rankos jungtys turi apribotas tam tikras vidines judesio galimybes. Tačiau, kad apibrėžti tuos judesio ribojimus reikia atsižvelgti ir į išorinius faktorius, tokius kaip išorinės jėgos, kurie gali įtakoti judesio ribų peržengimus. Be to kiekvieno žmogaus rankos dalių judėjimas yra individualus, pavyzdžiui vieni gali nykštį atlenkti daugiau kiti mažiau ir t. t. Mes nagrinėsime tikrai apytiksles (vidutines) rankos judėjimo ribojimų reikšmes, neveikiant jokios išorinės jėgos. Tai yra esant tik vidinėms rankos judėjimo galimybėms. Normalios žmogaus rankos judėjimas yra vaizduojamas jungčių kampų ribojimais modelyje. Šie ribojimai yra tik paprasčiausios ribinės kampų sukimosi reikšmės. Tad mes ir apibrėžime mūsų manymų mums labiausiai tinkančias reikšmes. Riešo jungties (RJ) judėjimo ribojimai:

$$-90^{\circ} < RJ (f/e) < 80^{\circ}$$

$$-15^{\circ} < RJ (a/a) < 30^{\circ}$$

$$-30^{\circ} < RJ \text{ sukimas (rotacija)} < 120^{\circ}$$

Riešo – delnkaulio jungties (RDJ) judėjimo ribojimai:

$$-90^{\circ} < RDJ (f/e) < 80^{\circ},$$

$$-15^{\circ} < RDJ (a/a) < 30^{\circ},$$

$$-30^{\circ} < RDJ \text{ sukimas (rotacija)} < 120^{\circ}.$$

Artimųjų pirštų jungčių (APJ), išskyrus nykštį, ribojimai:

$$-10^{\circ} < APJ (f/e) < 90^{\circ},$$

$$-15^{\circ} < APJ (a/a) < 30^{\circ}.$$

Viduriniųjų pirštų jungčių (VPJ), išskyrus nykštį, ribojimai:

$$0^{\circ} < VPJ (f/e) < 100^{\circ}.$$

Nutulusiųjų pirštų jungčių (NPJ), išskyrus nykštį, ribojimai:

$$0^{\circ} < NPJ (f/e) < 90^{\circ}.$$

Artimosios nykščio jungties (ANJ) ribojimai:

$$-10^{\circ} < ANJ (f/e) < 20^{\circ},$$

$$0^{\circ} < ANJ (a/a) < 80^{\circ}.$$

Nutuliosios nykščio jungties (NNJ) ribojimai:

$$10^{\circ} < NNJ (f/e) < 90^{\circ}.$$

Dar kartą reikėtų paminėti, kad neįmanoma nustatyti tikslių rankos judesio ribojimų, kurie tikėtų kiekvieno žmogaus rankai. Tad šie laipsniai yra vidutinės ribojimų reikšmės.

Piršto jungčių tarpusavio priklausomybės

Šis ribojimų – priklausomybių tipas nusako piršto vidinių jungčių tarpusavio priklausomybes, tai yra, kada lenkiant pirštą ties viena jungtimi jis automatiškai lenkiasi ir ties kita. Pavyzdžiui lenkiant pirštą ties viduriniąją piršto jungtimi (VPJ) pirštas automatiškai lenkiasi nuto-

lusioje piršto jungtyje (NPJ), arba atvirkščiai. Visa tai galime užrašyti taip:

$$VPJ (f/e) = \frac{2}{3} NPJ,$$

$$\text{arba } NPJ (f/e) = \frac{1}{3} VPJ.$$

Nykščio ribojimus galime užrašyti taip:

$$ANJ (f/e) = \frac{2}{3} NNJ,$$

$$\text{arba } NNJ (f/e) = \frac{1}{3} ANJ.$$

Tokios priklausomybės yra todėl, kad atliekant lenkimo veiksmą naudojama ta pati raumenų grupė ir judesiai natūraliai vyksta kartu. Čia susikerta rankos anatomijos studijos su realios rankos studijoms tarp šių jungčių.

Dar vienas svarbus vidinių pirštų jungčių priklausomybės niuansas yra tas, kad APJ jungtyje, kuo didesniu kampu naudojama atliekamas (f/e) veiksmas, proporcingai, tuo siauresnis tampa (a/a) judėjimo intervalas. Ir atvirkščiai.

Kaip bebūtų šie ribojimai neįtakoja laisvės laipsnių skaičiaus, o tik sumažina jungties judėjimo galimybes.

Jungčių tarp pirštų priklausomybės ir ribojimai

Sudėtinga, netgi beveik neįmanoma judinti pirštų nepriklausomai vienas nuo kito, kada jie yra tvirtai ištiesti ir jų neveikia jokios išorinės jėgos. Kaip pavyzdį galime paimti vidurinį (3 pirštas)(didįjį) pirštą, jį lenkiant matome, kad tose pačiose jungtyse lenkiasi ir bevardis (4 pirštas). Ir tokios priklausomybės galioja tarp daugelio pirštų, ir ne tik tarp tų pačių, bet ir skirtingų jungčių. Kad būtų aiškiau pavaizduosime kaip ir prieš tai buvusias priklausomybes:

$$\text{Mažylio (5 piršto) APJ (f/e) = } \frac{1}{3} \text{ Bevardžio (4 piršto)}$$

to) APJ (f/e), tai galioja visoms šių dviejų pirštų jungtims.

$$\text{Bevardžio (4 piršto) APJ (f/e) = } \frac{1}{3} \text{ Didžiojo (3 piršto)}$$

to) APJ (f/e), tai galioja visoms šių dviejų pirštų jungtims.

$$\text{Didžiojo (3 piršto) APJ (f/e) = } \frac{1}{3} \text{ Smiliaus (2 piršto)}$$

APJ (f/e) ir $\frac{1}{2}$ Bevardžio (4 piršto), tai galioja visoms šių dviejų pirštų jungtims.

$$\text{Smiliaus (4 piršto) APJ (f/e) = } \frac{1}{3} \text{ Didžiojo (3 piršto)}$$

APJ (f/e), tai galioja visoms šių dviejų pirštų jungtims.

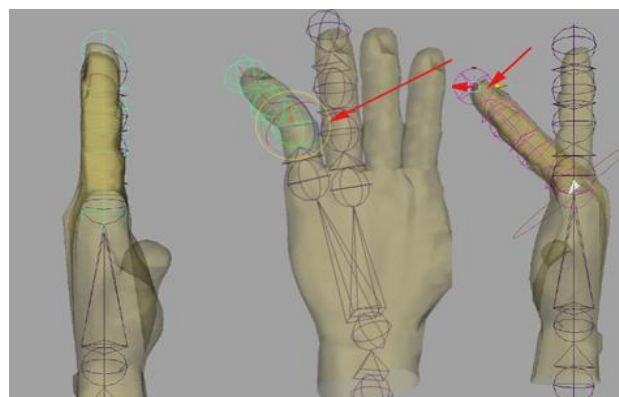
Reikėtų atkreipti dėmesį, kad šie skaičiai nebuvo patvirtinti jokiais moksliniais tyrimais, tai tik prielaidos. Tačiau jos padaro rankos animavimą gerokai tikroviškesnį ir galime tikroviškiau realizuoti kurčiųjų kalbos animaciją.

Rankos modelio formavimas ir animavimas su Autodesk Maya

Mūsų atveju mes naudosime Autodesk Maya programą. Kodėl pasirinkome Maya? Tai galinga programinė įranga, kurioje integruoti 3D modeliavimo, animacijos, efektų kūrimo ir vizualizavimo sprendimai. Ši programa skirta filmų ir video, žaidimų kūrėjams, profesionaliems dizaineriams ir studentams, siekiantiems sukurti patrauklius, tarsi tikrus skaitmeninius vaizdus. Mūsų atveju Maya pagalba mes galime realizuoti 3D rankos modelį (suformuoti jį iš nuotraukos). Maya [14,15] įrankių pagalba sumodeliuojame rankos skeletą, kurį integruojame į prieš tai sudarytą rankos modelį. Dar vienas didelis šios programinės įrangos paketo privalumas, kad galime modifikuoti ir paviršių tekstūros deformacijas kurias įtakoja modelio judėjimas. Tai suteikia modeliui realistiškumo.

Dėka išsamios rankos anatomijos, galime nustatyti jungčių pozicijas, kryptį ašyse, bei apriboti judėjimo intervalus. Sekančiame paveikslėlyje (5 pav.) galime pamatyti rankos ir rankos skeleto modelius autodesk Maya programoje.

Integravus skeletą į rankos modelį, sekantis žingsnis yra rankos judesių apribojimas. Judesių ribojimai nustatomi pasirenkant jungčių judėjimo intervalų ribinius parametrus Maya programoje ir tada mūsų modelio jungtys nebegali išeiti iš realios rankos judėjimo ribų. Apriboję modelį, toliau jį animuojame. Animacija vyksta laiko juostoje modeliuojant jungčių judėjimą laiko intervaluose. Judesio rezultatus matome paveikslėlyje (6 pav.):



6 pav. Animacija su Maya

Paskutinis žingsnis yra lietuvių kalbos daktilinės abėcėlės animavimas. Maya paketo pagalba sukurti visi

daktilinės abėcėlės ženklai ir perėjimus iš kiekvieno gesto, į kiekvieną kitą. Visi animacijos failai eksportuoti į video failus, kurie bus naudojami kuriant mokomąją programą daktilinės abėcėlės mokymuisi.

Išvados

1. Aptarus ir išanalizavus žmogaus rankos modelio formavimo būdus, bei jos judėjimo biomechaniką, pasirinkome ir pritaikėme mums labiausiai tinkamą modelį, bei pakoregavome to modelio jungčių skaičių. Parinkome mūsų modeliui labiausiai tinkančias ribines reikšmes.

2. Apžvelgę esamą programinę įrangą, nusprendėme, kad mums labiausiai tiktų Autodesk Maya programa, kadangi ji turi visus mums reikiamus įrankius skirtus rankos modelio formavimui ir animavimui. Be to Maya programoje galima modifikuoti odos (tekstūros) deformacijas, kas suteikia modeliui daugiau realistiškumo.

3. Maya programa gavome rankos animaciją, vaizduojančią kurčiųjų ženklų kalbos animaciją nuo kiekvienos raidės, į kiekvieną kitą. Gauti rezultatai yra geros kokybės ir yra tinkami mokomosios programos kurčiųjų kalbos mokymuisi kūrimui. Galutinis programos vartotojas klaviatūroje galės pasirinkti norimą raidę ir pamatys tam tikrą animacijos fragmentą iki jo pasirinktos raidės.

Literatūra

1. <http://lt.wikipedia.org/wiki/Ženklų_kalba>. [žiūrėta 2009 gruodžio 12 d.].
2. <<http://www.kurtieji.com>>. [žiūrėta 2008 gruodžio 12 d.].
3. <<http://members.surfeu.fi/deaf/links.html>>. [žiūrėta 2008 gruodžio 12 d.].
4. Nicoletta Adamo-Villani, Gerardo Beni. 2004. *A new method of hand gesture configuration and animation*. Department of Computer Graphics Technology, Purdue University, West Lafayette, IN, USA . Volume 7, Number 3, pp. 367-386. [žiūrėta 2009 sausio 16 d.]. Prieiga per internetą: <http://www2.tech.purdue.edu/cg/facstaff/nadamovillani/AdamoINF_2004.pdf>
5. By Jieun Lee and Myung-Soo Kim. 2007. *Human hand adaptation using sweeps: generating animatable hand models*. Comp. Anim. Virtual Worlds 2007; 18: 505–516.
6. Xiaopeng Yang, Jangwoon Park, Kihyo Jung, Heecheon You. 2008. *Development and Evaluation of a 25-Degree of Freedom Hand Kinematic Model*. Department of Industrial Engineering and Management, Pohang University of Science and Technology, Pohang 790-784, South Korea. [žiūrėta 2008 spalio 10 d.]. Prieiga per internetą: <http://center.postech.ac.kr/03_publication/DC/08_FL_ESK_HandModel.pdf>.
7. Irene Albrecht, Jörg Haber, and Hans-Peter Seidel. 2003. *Construction and Animation of Anatomically Based Human Hand Models*. MPI Informatik, Saarbrücken, Germany. [žiūrėta 2009 vasario 6 d.]. Prieiga per internetą: <<http://www.mpi-inf.mpg.de/~albrecht/SCA2003.pdf>>
8. George ElKoura and Karan Singh. 2003. *Handrix: Animating the Human Hand*. Department of Computer Science, Uni-

versity of Toronto, Toronto, Canada. [žiūrėta 2008 kovo 12 d.]. Prieiga per internetą: <http://www.dgp.toronto.edu/~gelkoura/noback/scapas_ca03.pdf>

9. <http://www.h-anim.org/>
10. Doina Dragulescu, Véronique Perdereau, Michel Drouin, Loredana Ungureanu and Karoly Menyhardt. 2007. *3D active workspace of human hand anatomical model*. Bio-Medical Engineering OnLine. [žiūrėta 2009 balandžio 4 d.]. Prieiga per internetą: <<http://www.biomedical-engineeringonline.com/content/pdf/1475-925X-6-15.pdf>>
11. Forward kinematics: the Denavit – Hartenberg convection. Chapter3. [žiūrėta 2009 vasario 12 d.]. Prieiga per internetą: <http://www4.cs.umanitoba.ca/~jacky/Teaching/Course_s/COMP_7950-HumanoidRobotics/ReadingList/spong_kinematics.pdf>
12. Kai-Nan An, William P. Cooney, Ronald L. Linscheid, E. Y. Chao E. Y. Chao. *Biomechanics of the Hand: A Basic Research Study (Hardcover)* World Scientific, 1989.
13. Huan Du, Edoardo Charbon. 2007. 3D hand model fitting for virtual keyboard system. [žiūrėta 2009 vasario 12 d.]. Ecole Polytechnique Fédérale de Lausanne 1015 Lausanne, Switzerland. Prieiga per internetą: <<http://ieeexplore.ieee.org/iel5/4569859/4581395/04581429.pdf?arnumber=4581429>>
14. <<http://www.highend3d.com/maya/tutorials/>>. [žiūrėta 2009 sausio 10 d.].
15. <http://www.3dlinks.com/tutorials_maya.cfm>. [žiūrėta 2009 sausio 10 d.].

J.Jocius

Summary

This article is about animation of sign language for deaf. At the beginning we are talking about sign language, its history and development stages. We review various virtual hand models, their movement biomechanics. Using all this information we determine restrictions of hand joints movement. The last step is modeling and animation of sign language gestures in virtual hand model with AutoDesk Maya.

IŠTIRPUSIO DEGUONIES KONCENTRACIJOS MATEMATINIO MODELIO TYRIMAS

Kęstutis Jonelis

Doktorantas

Kauno technologijos universitetas, Procesų valdymo katedra

el. p. kestui@mail.lt

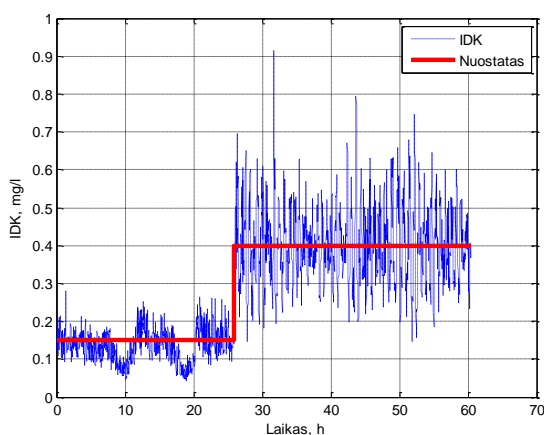
Anotacija. Šiame straipsnyje aprašomas matematinio modelio identifikavimas ir rezultatų patikra (validacija) ištirpusio deguonies koncentracijos dinamikos elgsenai modeliuoti pramoniniame nutekamųjų vandenų biologinio valymo procese. Identifikavimo procedūra atlikta remiantis Kauno vandenvalos įmonės technologinių parametru stebėjimo archyviniais duomenimis. Detaliai aprašytas technologinis procesas, pateikti skaičiavimai ir aptarti sunkumai, kilę atliekant darbą.

Reikšminiai žodžiai: Matematinis modelis, identifikavimas, ištirpusio deguonies koncentracija, biologinis nutekamųjų vandenų procesas.

Įvadas

Tikslus ir optimalus ištirpusio deguonies koncentracijos (IDK) valdymas yra labai svarbus technologinis uždavinys, nes orapūtės suvartoja labai daug elektros energijos ir todėl jas būtina efektyviai valdyti. Tačiau automatinis IDK valdymas yra sudėtingas uždavinys, nes procesas yra netiesinis ir veikiamas daugybės trikdžių. Todėl tradiciniai tiesiniai reguliatoriai (PI, PID) su pastoviais derinimo parametrais negali kokybiškai valdyti proceso, kuomet pakinta technologinės sąlygos ar pradeda veikti dideli trikdžiai (D. Levišauskas 1995).

Pastaruoju metu yra sukurta visa eilė metodų, padedančių spręsti netiesiškumo ir trikdžių problemas remiantis proceso būsenos modeliu, naudojamų nuspėti proceso dinamiką (Bastin et al 1990, Lindberg, 1998, Yoo et al, 2001)



1 pav. IDK valdymo sistemos darbas su tradicinius PI reguliatoriumi esant nestabiliam procesui (duomenys iš pramoninės vandenvalos įmonės).

IDK valdymo problema yra svarbi Kauno vandenvalos įmonėje, nes įprasta nuostato automatinio valdymo sistema procesą valdo prastai (1 pav.).

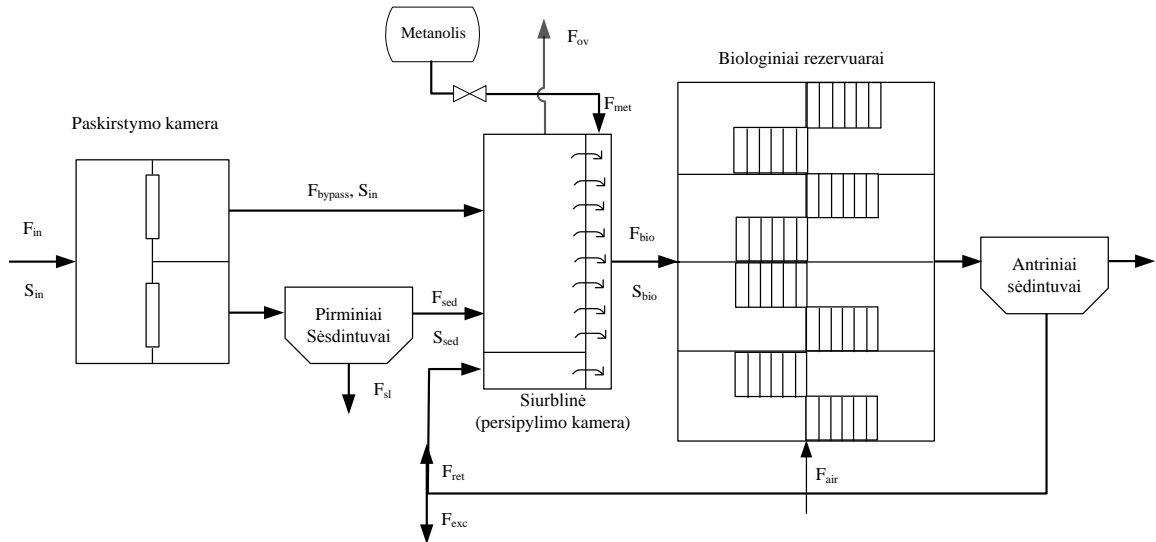
Šiame darbe yra identifikuotas IDK būsenos modelis proceso dinamikai modeliuoti biologiniame nuotekų valymo procese. Matematinis modelis identifikuotas remiantis proceso technologinių parametru stebėjimų duomenimis ir bus naudojamas kurti adaptyvias valdymo sistemas.

Technologinis procesas

Proceso technologinė schema yra pateikta 2 – am paveikslėlyje. Atitekančių nuotekų srautas F_{in} , kuriame substrato koncentracija žymima S_{in} pro mechaninius filtrus patenka į surinkimo kamerą. Priklausomai nuo apvadinės sklendės padėties, srautas padalijamas į 2 dalis: viena teka į pirminius sėsdintuvus (F_{sed}), kita – tiesiai į persipylimo stotį, iš kur galiausiai patenka į biologinio valymo rezervuarus (F_{bypass}). Srautų padalijimas įvertinamas koeficientu k_{sed} , kuris parodo, kuri srauto dalis % teka per pirminius sėsdintuvus ir priklauso nuo motorizuotos apvadinės sklendės vartų padėties (0% - uždaryta, 100% - pilnai atidaryta). Ši priklausomybė yra įvertinama interpoliuojant 3 taškus, pateikiamų surinkimo kameros dokumentacijoje (1 lentelė).

1 lentelė. Apvadinės sklendės vartų padėtis ir k_{sed} taškai

Apvadinės sklendės vartų padėtis g, %	k_{sed} , %
0	100
28.8	50
100	0



2 pav. Biologinio nuotekų valymo proceso technologinė schema

Priklausomybė aproksimuota panaudojant kvadratinį polinomą:

$$k_{sed} = 0.01034 \cdot g^2 - 2.034 \cdot g + 10. \quad (1)$$

Nuotekų srautas per pirminius sėdintuvus lygus:

$$F_{sed} = \frac{k_{sed}}{100} F_{in}. \quad (2)$$

Dumblo srautas F_{sl} iš pirminių sėdintuvų yra labai mažas, palyginus su atitekančiu nuotekų srautu (~1%), tai tolimesniuose skaičiavimuose jis nevertinamas. Priklausomai nuo nuotekų srauto per pirminius sėdintuvus, substrato koncentracija ištekančiame iš sėdintuvų vandenyje gali sumažėti nuo 15% (esant didiam srautui) iki 40% (kai srautas mažas). Procentinis substrato koncentracijos sumažėjimas ištekančiame iš sėdintuvo sraute įvertinamas tokia tiesine priklausomybe:

$$k_{dec} = -0.0043 \cdot F_{sed} + 40, \quad (3)$$

kur k_{dec} yra procentinis substrato koncentracijos sumažėjimas ištekančiame sraute, %; F_{sed} yra ištekantis srautas, m^3/h .

Atitinkamai, substrato koncentracija ištekančiose nuotekose iš pirminių sėdintuvų yra lygi:

$$S_{sed} = S_{in} - \frac{k_{dec}}{100} S_{in}. \quad (4)$$

Abu atitekantys srautai yra vėl sumaišomi siurblinėje (persipylimo kameroje). Jeigu suminis srautas yra didesnis nei $5855 m^3/h$, perteklinis srautas F_{ov} nuvedamas į išorę, apeinant biologinius rezervuarus ir antrinius sėdintuvus. Jei reikalinga pagerinti proceso eigą, gali būti pridėta šiek tiek metanolio (F_{met}). Sumaišytų nuotekų srautas F_{bio} , kurio substrato koncentracija S_{bio} , patenka į keturis biologinio valymo rezervuarus. Kadangi metanolis nebu-

vo paduodamas proceso tyrimo metu, substrato koncentracija S_{bio} įtekančio į biologinius rezervuarus nuotekų sraute yra apskaičiuojama pagal tokią formulę:

$$S_{bio} = \frac{s_{in} \left(F_{in} - \frac{k_{sed}}{100} F_{in} \right) + s_{sed} \frac{k_{sed}}{100} F_{in} + s_{ret} F_{ret}}{F_{in} + F_{ret}}, \quad (5)$$

kur F_{ret} ir S_{ret} yra atitinkamai grąžinamo dumblo iš antrinių sėdintuvų srautas ir substrato koncentracija. Perteklinis dumblas F_{exc} yra pašalinamas tolimesniam perdirbimui.

Oro srautas F_{air} paduodamas iš orapūčių stoties į visus 4 biologinius rezervuarus ir yra apskaičiuojamas matuojant orapūčių kompresorių energijos suvartojimą P . Remiantis vandenvalos įmonės technine dokumentacija, bendras oro srautas yra apskaičiuojamas remiantis tokia tiesine priklausomybe:

$$F_{air} = 0.05455 \cdot P - 1200, \quad (6)$$

kur P yra kompresorių suvartota elektros energija.

Šiame darbe buvo naudoti tik vieno aerotanko stebėjimų duomenys. Oro srautas į pasirinktą aerotanką buvo apskaičiuotas remiantis oro sklendžių padėtimis kiekviename aerotanke:

$$F_{air4} = F_{air} \frac{V_4}{V_1 + V_2 + V_3 + V_4}, \quad (7)$$

Kur F_{air4} yra oro srautas pasirinktame aerotanke (4-tas), m^3/h , v_1, v_2, v_3, v_4 aerotankų oro sklendžių padėtys, %.

IDK matematinis modelis biologiniame nuotekų valymo procese

Būsenos modelis biologinio nuotekų valymo procese yra sudarytas remiantis masių balansu ištirpusio deguonies koncentracijos ir substrato (D. Levišauskas 1995, Bastin et al 1990, Lindberg, 1998, Yoo et al 2001, Staniskis et al 1992):

$$\frac{dc}{dt} = k_L a (c_{satur} - c) + (c_{in} - c) \frac{F_{bio}}{V} - OUR, \quad (8)$$

$$\frac{ds}{dt} = - \frac{\mu_{max}}{Y_{x/s}} \frac{s}{K + s} X + \frac{S_{bio} - s}{V} F_{bio}, \quad (9)$$

$$OUR = \frac{\mu_{max}}{Y_{x/o}} \frac{s}{K + s} X, \quad (10)$$

$$k_L a = \alpha N^\beta Q^\gamma, \quad (11)$$

kur c yra ištirpusi deguonies koncentracija, $k_L a$ - deguonies masės perdavimo koeficientas iš dujinės į skystąją fazę, c_{satur} yra įsotinta IDK vertė, c_{in} - IDK atitekančiose nuotekose, F_{bio} įtekančių į aerotanką nuotekų srautas, V yra aerotanko tūris, OUR - deguonies perdavimo koeficientas, s - substrato koncentracija, μ_{max} - maksimalus savitasis augimo greitis, K - prisotinimo koeficientas, $Y_{x/s}$ - našumo rodiklis biomasei/substratui, $Y_{x/o}$ - našumo rodiklis biomasei/deguoniui, X - biomasės koncentracija, S_{bio} - substrato koncentracija įtekančiose į aerotanką nuotekose, N - maišyklės sukimosi greitis, Q - oro srautas, α, β, γ - parametrai, priklausantys nuo technologinių sąlygų.

Modelio parametrų identifikavimas

Kadangi vienintelis būsenos kintamasis c yra tiesiogiai matuojamas (s įvertina modelis), modelio parametrų identifikavimui pasirinkta tikslo funkcija minimizuoti suminę integralinę kvadratinę paklaidą:

$$S = \int_0^T (c_{model} - c_{data})^2 dt \rightarrow \min, \quad (12)$$

kur c_{model} yra modelio prognozuojama vertė, c_{data} yra IDK realaus proceso matavimų duomenys, T - tyrimo laikotarpis. Modelio lygčių integravimui (8), (9), diskretiniai matavimų duomenys technologinių parametrų F_{in} , F_{ret} , Q ir c buvo interpoluoti naudojant Matlab įrankius (tiesinė Lagranžo interpoliacija). Pradinė vertė tiesiogiai nematuojamo būsenos kintamojo $s_{t=0}$ buvo nustatyta proceso eksperto. Įtekančio į aerotanko substrato koncentracija S_{bio} apskaičiuojama iš (5) formulės. Biomasės koncentracija X įvertinama iš tokios tiesinės priklausomybės:

$$X = X_0 + F_{ret} \cdot k, \quad (13)$$

kur X_0 ir k yra identifikuojami parametrai.

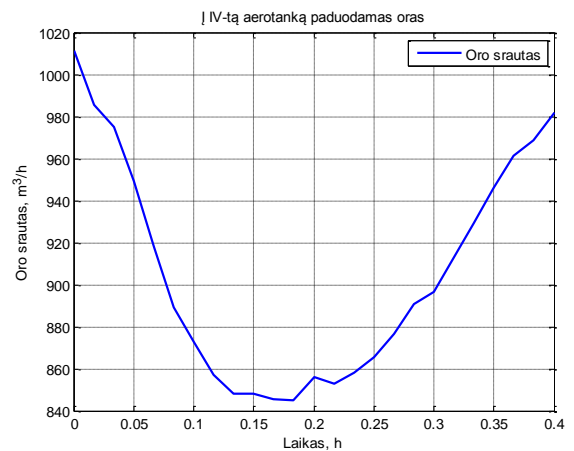
Modelio parametras c_{in} yra tiesiogiai matuojamas, c_{satur} yra fizinė konstanta, V yra konstrukcinis parametras. Kadangi maišymo greitis tiriamajame procese yra pastovus, išraiška αN^β identifikuojama kaip parametras α^* . Pradinės parametrų $\alpha^*, \gamma, \mu_{max}, K, Y_{x/s}, Y_{x/o}$ vertės paimtos iš literatūros (S. Eswaramoorthi et al 2003, Glen T. et al 1998, Repšytė et al 2001). Tikslo funkcijai (12) panaudotas modifikuotas evoliucinio programavimo algoritmas (V.W. Porto et al 1995). Paieškos metu identifikuojami parametrai gali kisti tik griežtai apibrėžtose technologinėse ribose. Mutacijos laipsnis perskaičiuojamas po kiekvienos iteracijos. Dalis vektorių tėvų dirbtinai pakeičiama, siekiant kuo greičiau apeiti lokalius ekstremumus. Paieškos procedūra baigiama, kuomet pasiekiamas užduotas tikslumas. Skaičiavimo programa realizuota panaudojant Matlab įrankius.

Pradinės būsenos kintamųjų ir identifikuotų modelio parametrų vertės pateikiamos 2-oje lentelėje.

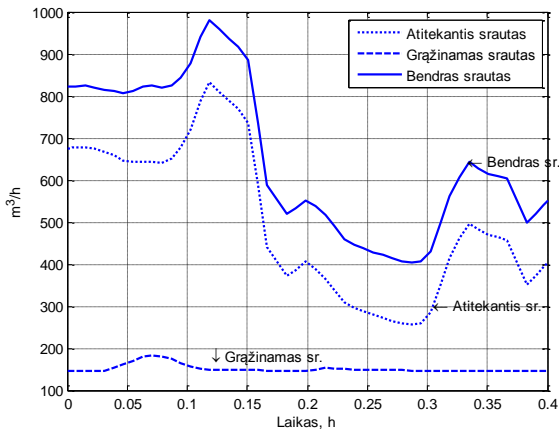
2 lentelė. Pradinės būsenos kintamųjų ir identifikuotos modelio parametrų vertės

Pradinės būsenos kintamųjų vertės		
Kintamasis	Matavimo vienetąs	Vertė
c	mg/l	0.16
s	mg/l	32.28
Identifikuotos modelio parametrų vertės		
α^*	-	5.09
γ	-	0.138
μ_{max}	h ⁻¹	0.10
K	mg/l	35.36
$Y_{x/s}$	mg/mg	2.80
$Y_{x/o}$	mg/mg	0.052
X_0	mg/l	140
k	-	-0.0006

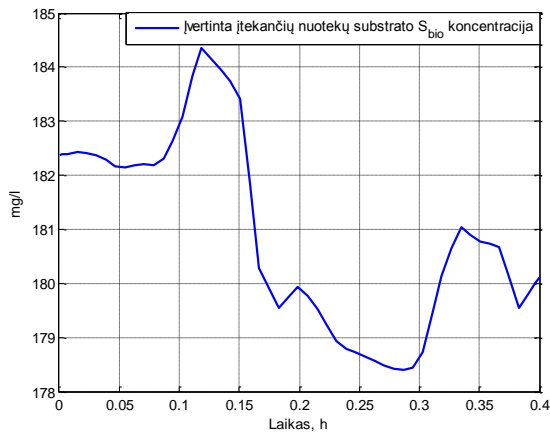
Stebėjimų duomenys ir modelio prognozuojamos IDK vertės pateiktos paveikslėliuose 3-6.



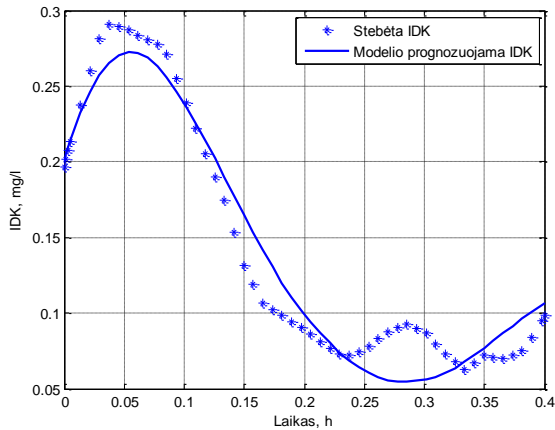
3 pav. Paduodamas oro srautas



4 pav. Atitekantis, grąžinamas ir bendras nuotekų srautai

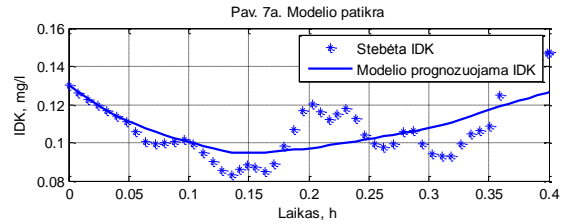


5 pav. Įvertinta substrato koncentracija atitekančiose ir biologiniuose rezervuaruose nuotekose

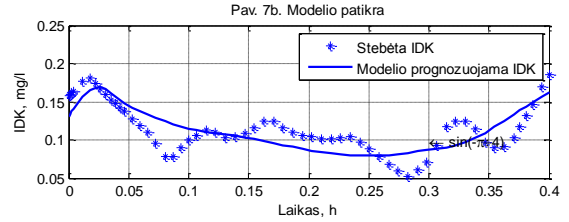


6 pav. Modelio prognozuojama ir reali IDK vertės

7 – tame a ir b paveikslėliuose pateikti modelio patikros (validacijos) rezultatai. Modelio patikrai buvo naudoti kiti stebėjimų duomenys.

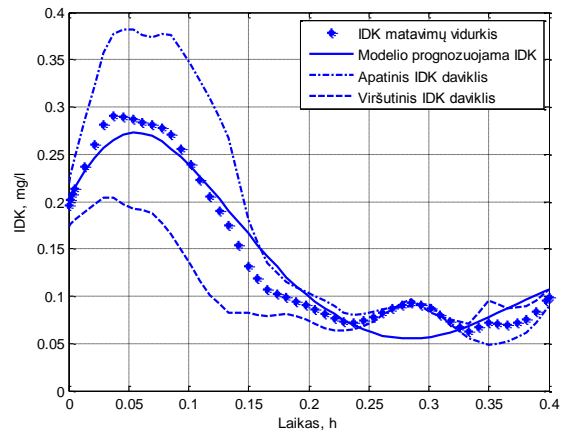


7 pav. Modelio patikros rezultatai



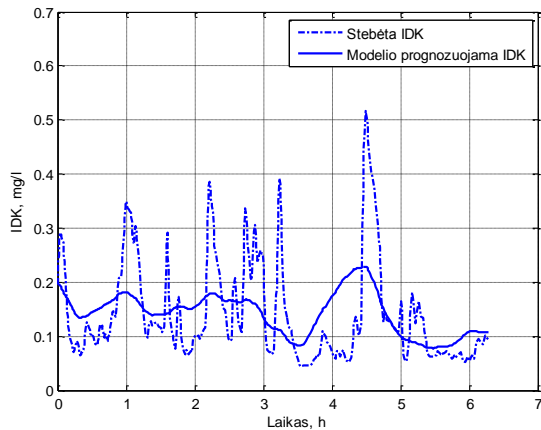
Darbo aptarimas

Vienas sudėtingiausių iššūkių šiame darbe buvo atrinkti informatyvius stebėjimų (matavimų) duomenis. Tiek modelio identifikavimui, tiek patikrai buvo paimti pereinamųjų procesų duomenys, kad būtų galima kuo tiksliau identifikuoti modelį. Matavimai vandenvals įmonėje atliekami kas 15 sekundžių, todėl, siekiant kiek nufiltruoti daviklių parodymus, identifikavimui buvo naudotas minutės stebėjimų vidurkis. Kitas svarbus dalykas - IDK daviklių (matuoklių) parodymai. Kadangi jie išdėstyti skirtingose aerotanko vietose, jų parodymai skiriasi. Šiame darbe buvo naudotas aritmetinis parodymų vidurkis, o kaip iš tikrųjų atrodo matuojama IDK, parodyta 8 pav.



8 pav. Daviklių, vidutinė ir modelio prognozuojama IDK vertės

Identifikavimo rezultatai taip pat priklauso ir nuo imties dydžio – kitaip tariant, kokio ilgio duomenų periodas pasirenkamas. Jei imtis per didelė, identifikuotas modelis veikia kaip filtras (9 pav.).



9 pav. Identifikavimo rezultatai, kuomet imtis per nelyg didelę

Išvados

Šiame darbe buvo sukurtas ištirpusio deguonies koncentracijos matematinis modelis, kuriuo alternatyviai galima tirti IDK elgseną kintant technologinėms sąlygom. Modelio identifikavimo ir patikros rezultatai rodo, kad modelis yra tinkamas prognozuoti proceso dinamiką. Kitas tyrimų etapas bus sukurti realaus laiko adaptyvią valdymo sistemą, kad būtų galima pagerinti IDK valdymo kokybę.

Literatūra

1. Bastin, G. and Dochain, D., "On-Line Estimation and Adaptive Control of Bioreactor," Elsevier, New York, 1990
2. D. Levišauskas. An Algorithm for Adaptive Control of Dissolved Oxygen Concentration in Batch Culture // *Biotechnology Techniques*. 1995, Vol. 9, n.2, p. 85-90.
3. Interneto prieiga <http://www.ensic.inpl-nancy.fr/COSTWWTP>
4. Yoo, C.K., Cho, J.H., Kwak, H. J., Choi, S. K., Chun, H.D. and Lee, I., "Closed-Loop Identification and Control for Dissolved Oxygen Concentration in the Full-Scale Coke Wastewater Treatment Plant," *Wat. Sci. Tech.*, in press, 2001
5. Glen T. Daigger, John A. Buttz. *Upgrading wastewater treatment plants*. Second edition. Lancaster, Pennsylvania, 1998
6. Lindberg, C. F., "Multivariable Modeling and Control of an Activated Sludge Process," *Wat. Sci. Tech.*, 37, 149, 1998.
7. Repšytė Jolanta, Simutis, Rimvydas, *Mathematical model of biological wastewater treatment process* / Jolanta Repšytė, Rimvydas Simutis. // *Informacinės technologijos ir valdymas*. Kaunas: Technologija. ISSN 1392-124X. 2001, nr. 4(21), p. 21-30

8. S. Eswaramoorthi K. Dhanapal and D.D. Chauhan. *Application of membrane bioreactor for textile wastewater treatment: Pilot plant process modeling and scale-up*, New Delhi, India, 2003
9. Staniskis, J., Levisauskas, D., Simutis, R., Viesturs, U., Kristapsons, M. *Automation of Biotechnological Processes: Automatic Monitoring, Optimization and Control*. ISBN 5-7968-0593-3. Riga, Zinatne, 1992, 348 p. (in Russian - Ю. Старишкис, Д. Левишаускас, Р. Симулис, У. Виестурс, М. Кристасонс, Автоматизация биотехнологических процессов: автоматический контроль, оптимизация и управление. ISBN 5-7968-0593-3. Рига, Зинатне, 1992, 348 с.)
10. V.W.Porto, D.B.Fogel, L.J.Fogel. Alternative neural networks training methods. *IEEE Expert*, 1995, 16-22.

GAMS TECHNOLOGIJŲ VERTINIMAS KAI KURIEMS GLOBALIOSIOS OPTIMIZACIJOS UŽDAVINIAMS

Lina Kulbytė

Magistrantas,

Vilniaus Gedimino technikos universitetas,
el. p. lina.kulbyte@gmail.com

Anotacija. Straipsnyje bandomi įvertinti globaliosios optimizacijos paketo GAMS sprendikai sprendžiant kuprinės ir keliaujančio pirklio globaliosios optimizacijos uždavinius. Kuprinės uždaviniui pateikiami visų variantų perrinkimo ir GAMS sistemos mišraus sveikojo programavimo (MIP) metodo gauti sprendimo rezultatai bei jų palyginimas. Taip pat surandamas geriausias, pagal laiką ir sprendimo rezultatą, GAMS sprendikas šiems uždaviniams spręsti.

Reikšminiai žodžiai: GAMS, sprendikas, kuprinės, keliaujančio pirklio uždaviniai, globalusis optimizavimas.

Įvadas

Nuo seniausių laikų optimizavimo problemų gausu praktinėje žmonių veikloje. Optimalumas siejamas su geriausiais galimybių variantais, išvalgiausiomis elgesio strategijomis, kai siekiama tam tikro tikslo – didžiausios naudos ar mažiausių išteklių sąnaudų. Inžinerinėje veikloje optimizavimas yra būtinas dalykas. Visi inžinieriai, projektuojantys naujas technologijas, siekia maksimalaus efektyvumo. Dažniausiai optimizavimo uždaviniai turi kelis sprendinius – yra globaliosios optimizacijos uždaviniai. Šios optimizacijos tikslas – sukurtiems modeliams rasti globalų geriausią sprendinį iš (galimų arba žinomų) daugelio lokalių optimumų. Globalioji optimizacija yra žymiai sudėtingesnė už lokaliąją, tad tokiems uždaviniams reikalingi didesni resursai. Tačiau praktinis globaliosios optimizacijos uždavinių sprendimas yra labai aktualus.

Uždavinio formavimas

Kaip pasinaudoti optimizavimo teorija, kai kokiame nors praktiniame uždavinyje iškyla optimizavimo būtinybė? Norint suformuluoti uždavinį matematiškai, reikia sudaryti taikomojo uždavinio matematinį modelį. Matematinis uždavinio formulavimas labai svarbus optimizavimo etapas, nuo kurio priklauso ir jo išsprendžiamumas, ir gautų rezultatų adekvatumas praktiniam uždaviniui, kurį jis formalizuoja. Pirmiausia, reikia apibrėžti kintamųjų vektorių, bei tikslo funkciją ir leistinąją sritį. Optimizavimo procedūra parenka tokias kintamųjų vertes, su

kuriomis tikslo funkcija įgyja didžiausią arba mažiausią vertę. Todėl uždaviniui išspręsti būtina sudaryti algoritmą tikslo funkcijos ir ribojimus apibrėžiančių funkcijų reikšmėms apskaičiuoti. Remiantis matematinio modelio savybėmis ir optimizacijos teorijos rezultatais pasirenkamas tinkamiausias metodas suformuluotam optimizavimo uždaviniui spręsti.

Bendra Algebrinė Modeliavimo Sistema – GAMS

Esminė pažanga algoritmų ir kompiuterinių kodų skirtų didelėms matematinėms problemoms spręsti, buvo padaryta 1950-aisiais ir 1960-aisiais. Šiais laikais yra daug įvairių modeliavimo ir optimizavimo sistemų:

- AIMMS,
- AMPL,
- MPL,
- MathPro,
- OMNI,
- LINDO,
- GAMS.

Šios sistemos sprendžia skirtingų tipų modelius (tiesinius, netiesinius, stochastinius ir kt.), vienos iš jų pritaikytos spręsti tik tam tikram uždavinių tipui, kitos – apima kelis tipus. Daugiausiai uždavinio tipų apimanti ir viena iš populiariausių bei seniausių optimizavimo sistemų yra GAMS. Ši sistema buvo sėkmingai panaudota ir pramonėje, ir akademinuose uždaviniuose. Nuo 1987-ųjų turi

daugiau nei 10 tūkstančių naudotojų daugiau kaip 100 šalių.

Bendra algebrinė modeliavimo sistema (angl. **General Algebraic Modeling System**) yra aukšto lygio modeliavimo sistema matematinio programavimo ir optimizavimo problemoms spręsti. Ši sistema suprojektuota modeliuoti sudėtingas ir dideles optimizavimo problemas, naudotojui leidžia sukurti didelius modelius, kurie lengvai pritaikomi naujoms situacijoms. Sistema susideda iš kalbos kompiliatoriaus ir stabilaus, integruoto aukšto lygio įrenginio - sprendiko. Kitaip sakant, GAMS yra modernių optimizavimo programų paketas, turintis keliasdešimt optimizavimo varikliukų. Visas šias programas jungia vidinė GAMS programavimo kalba. Ji pirma algebrinė modeliavimo kalba (AML) ir yra formaliai panaši į paprastai vartojamas programavimo kalbas. Modeliai yra apibūdinami glaustomis algebrinėmis formuluotėmis, kurias lengva skaityti ir žmonėms, ir mašinoms. Naudotojas gali susikoncentruoti ties algebra, pakeisti pavyzdinę formuluotę, keisti vieną sprendiką kitu ir net tiesinį modelį paversti į netiesinį.

Uždavinių teoriniai modeliai

GAMS gali spręsti, ir analizuoti mažą ir paprastą bei didelę ir sudėtingą optimizavimo problemą. Straipsnyje GAMS paketą analizuosime spręsdami du gerai žinomus globaliojo optimizavimo uždavinius – kuprinės ir keliaujančio pirklio. Kuprinės ir keliaujančio pirklio uždaviniai atspindi labai daug šiuolaikinių realių uždavinių. Yra nemažai šių uždavinių modifikacijų ir variacijų. Visų šių uždavinių esmė – suskirstyti tam tikrų objektų aibę į atskirus poaibius, atsižvelgiant į tam tikrus reikalavimus, kurie skirtingoms uždavinių variacijoms gali būti skirtingi. Abu šie uždaviniai gali būti sprendžiami kaip mišriojo sveikojo programavimo (MIP) uždaviniai.

Jei nežinomi kintamieji yra visi sveikieji skaičiais, tai problemą vadinama sveikojo skaičiaus programavimu (IP). MIP (ang. mixed integer programming) – mišrusis sveikojo skaičiaus programavimas, kur kai kurie iš kintamųjų yra realieji, o kiti yra sveikieji skaičiai. Matematiškai mišriojo sveikojo skaičiaus programavimo problema atrodo taip:

Minimizuoti (arba maksimizuoti)

$$cx + dy$$

$$Ax + By \geq b$$

$$L \leq x \leq U$$

$$y = \{ 0, 1, 2, \dots \}$$

kur x yra kintamųjų vektorius – realieji skaičiai, o y yra kintamųjų vektorius, kur gali būti tik sveikieji skai-

čiai. $cx + dy$ yra tikslo funkcija, o $Ax + By \geq b$ apibrėžia apribojimų aibę. L ir U yra kintamųjų vektorių apatinės ir viršutinės ribos. $y = \{0, 1, 2, \dots\}$ yra vientisumo reikalavimas sveikiems kintamiesiems y .

Kuprinės uždavinyje žygeivis turi pasirinkti reikalingiausių daiktų su sąlyga, kad bendras jų svoris ir tūris neviršytų leistinų. Tačiau reikalingų daiktų svoris ir tūris viršija leistinuosius. Žinoma žygeivis galėtų rinktis bet kaip, net atsitiktinai, ką krauti į kuprinę, o ką palikti. Bet panagrinėkime racionalaus elgesio atvejį. Tarkime, turime n nedalomų skirtingų objektų (daiktų), kurių vertė (kaina) p_j , svoris w_j ir tūris g_j , kur $j = 1, \dots, n$. Į kuprinę galima sudėti ne daugiau kaip W svorio vienetų ir C tūrio vienetų krovinį. Atsižvelgiant į kuprinės talpą ir maksimalų leistiną svorį, reikia nustatyti kraunamų į kuprinę maksimalios vertės daiktų rinkinį. Kuprinės sąvoka čia yra apibendrinta, vietoje kuprinės gali būti kiti objektai: sunkvežimis, traukinio vagonas, lėktuvas, laivas, sandėlis ar bet kokia talpykla, o vietoje naudojamų apribojimų galimi ilgio, pločio, išteklių kiekio ir t.t. Kuprinės uždavinys taip pat turi daug ekonominių interpretacijų, žinomiausia yra susieta su investicijų paskirstymu tam tikromis sąlygomis.

Sudėtingesnis uždavinys yra keliaujančio pirklio, jis yra atskiras atvejis bendresnio Hamiltono maršrutų radimo uždavinys. Logistikoje labai dažnai yra sprendžiami tokie uždaviniai. Sakykime, turime N miestų, kurie sudaro visiškai kryptingą grafą su C - atkarpos kaina, tad uždavinys yra nesimetris. Pirklys, išėjęs iš pirmo miesto, turi aplankyti visus miestus ir grįžti į pradinį miestą. Į kiekvieną miestą galima patekti tik vieną kartą, t.y. į miestą patekti ir iš jo išeiti galima tik kartą. Reikia rasti mažiausią kainą turintį pirklio maršrutą. Dažnai šis uždavinys yra aprašomas grafais. Miestai sudaro grafų viršūnių aibę, o atstumai tarp miestų – grafo briaunos, kurios surašomos į grafo briaunų gretimumo matricą.

GAMS technologijos vertinimas

Paprasčiausias būdas rasti maksimalios vertės kuprinę yra sugeneruoti visus įmanomus kuprinės formavimo variantus ir iš jų išrinkti patį geriausią variantą. Taip pat galima rasti ir pirklio maršrutą – sugeneruoti visus skirtingus viršūnių išsidėstymo būdus, patikrinti kurie iš jų apibrėžia maršrutą. Šio algoritmo pliusas toks, kad mes visada rasime optimalų rezultatą, tačiau šis metodas turi nemažai minusų. Jis gali būti naudojamas tik santykinai mažiems uždaviniams spręsti.

Pilnam kuprinės variantų perrinkimo metodui (brute force) buvo parašyta programa C kalba. Šis metodas be-

veik visuose praktiniuose uždaviniuose yra labai didelė problema dėl laiko sąnaudų, todėl dažnai tenka taikyti apytikšlius paieškos metodus arba naudoti optimizavimo programas, tokias kaip GAMS. Didžiausias uždavinys visų variantų perrinkimo metodu buvo išspręstas su 16 daiktų. Pasirenkame kelis skirtingus apribojimus uždaviniui su tais pačiais duomenimis ir palyginkime gautus rezultatus su GAMS Baron sprendiko rezultatais.

1 lentelė. Kuprinės uždavinio GAMS ir perrinkimo programos verčių ir laikų palyginimas

W / C (svorio / tūrio)	Laikas (s)		Kuprinės vertė	
	GAMS	Perrin - kimu	GAMS	Perrin - kimu
300/300	00.453	433	507	510
310/310	00.344	738	507	517
320/320	00.328	1262	522	527
330/330	00.343	1973	542	542
340/340	00.328	2905	542	552
350/350	00.344	5021	542	562
360/360	00.344	8226	562	569

1 lentelėje matyti, kad GAMS rezultatai atitinka perrinkimo programos gautus rezultatus. Kai kuriais atvejais GAMS duoda „blogesnius“ rezultatus nei visų variantų perrinkimas. Skaičiavimo laikai šių abiejų būdų gerokai skiriasi. Perrinkimo būdas daug lėtesnis už GAMS sprendikus, tiksliau GAMS visus šiuos uždavinius išsprendžia per milisekundes. Gauti rezultatai patvirtina, kad perrinkimas visada turi duoti globalųjį rezultatą, o tuo tarpu GAMS, nors jo sprendikai yra taikomi globaliajai optimizacijai, ne visada tą globalųjį maksimumą (ar minimumą) randa.

GAMS platforma turi nemažai optimalių „varikliukų“ - sprendikų. Toliau seks šių varikliukų palyginimas. Skaičiavimai buvo atlikti su uždaviniu, kur: N = 16 daiktų, maksimalus leistinas kuprinės svoris – 360, tūris – 360, su visai tinkamais sprendikais pasirinktam MIP optimizavimo metodu.

2 lentelė. Kuprinės uždavinio GAMS sprendikų palyginimas

Sprendikas (MIP)	Rezultatas
BARON	562
BDMLP	562
CoinCbc	567
CoinGlpk	529
CPLEX	562
MOSEK	567
OSL	516
XA	529
XAPAR	529
XPRESS	567

Visų variantų perrinkimo metodu išspręstas šis uždavinys duoda globalų ir vienintelį geriausią sprendinį – maksimali kuprinės vertė 569. GAMS sprendikai duoda globalaus sprendinio artinius – 567, nei vienas sprendikas nerado globalaus optimumo. Galime daryti prielaidą, kad trys geriausi sprendikai sprendinių atžvilgiu – **CoinCbc**, **MOSEK**, **XPRESS**.

Kaip ir kuprinės uždavinį, taip ir keliaujančio pirklio uždavinį išsprendžiame su GAMS. Skaičiavimai buvo atlikti su uždaviniu, kur: N = 7 miestų, iš kiekvieno miesto galima patekti į kiekvieną (visiškai jungus grafą). Uždavinys yra nesimetrinis ir jis sprendžiamas su visais tinkamais sprendikais pasirinktam MIP optimizavimo metodu.

3 lentelė. Keliaujančio pirklio uždavinio GAMS sprendikų laikų palyginimas

Sprendikas (MIP)	Laikas, ms
BARON	875
BDMLP	281
CoinCbc	453
CoinGlpk	297
CPLEX	657
MOSEK	547
OSL	515
XA	313
XAPAR	250
XPRESS	813

Visų sprendikų sprendinys yra vienodas, skiriasi tik skaičiavimo laikas. Iš 3 lentelės matyti, kad geriausias sprendikas laiko atžvilgiu yra BDMLP, CoinGlpk ir XAPAR.

Norint patvirtinti arba paneigti padarytas prielaidas apie geriausius GAMS sprendikus, reikia išspręsti daugiau įvairaus dydžio uždavinių, kad būtų galima gauti teisingą teiginį, tolimesni tyrimai ir darbai leis tai padaryti.

Išvados

1. GAMS sprendikų rezultatai – ne visada globalūs net ir mažiems uždaviniams, ką rodo jų rezultatų palyginimas su perrinkimu gautais rezultatais, tačiau jų sprendimo laikas yra mažas, tinka spręsti dideliems uždaviniams.

2. Laimėdami laiko – pralaimime tikslumą, tačiau sprendžiant dideles problemas, laikas yra labai svarbus, tad naudotojai dažniausiai renkasi komercines kompiuterines programas, kurios savyje turi optimalius sprendikus.

Literatūra

- Apynis A. Optimizavimo metodai. Vilnius: Vilniaus universiteto leidykla, 2005. 199p.
- Arora Jasbir S. Introduction to optimization. Department of Civil and Environmental Engineering, Department of Mechanical and Industrial Engineering, Center for Computer Aided Design, The University of Iowa, Iowa City, Iowa, U.S.A.
- Čiegis R. Duomenų struktūros, algoritmai ir jų analizė. Vilnius: Technika, 2007. 309p.
- Dzemyda G., Šaltenis V., Tiešis V. Optimizavimo metodai. Vilnius: Mokslo aidai, 2007. 161p.
- G'abor Pataki Teaching Integer Programming Formulations Using the Traveling Salesman Problem. 2003 Society for Industrial and Applied Mathematics
- GAMS pagrindinis puslapis. Interneto prieiga www.gams.com
- Hahsler M., Hornik K. TSP – Infrastructure for the Traveling Salesperson Problem. Journal of Statistical Software. December 2007, Volume 23, Issue 2.
- Kalanta S. Taikomosios optimizacijos pagrindai. Vilnius: Technika, 2007. 474p.
- Meeraus A., Paul van der Eijk, Bussieck M., Dirkse S., Steacy P. McCarl GAMS User Guide. 2008 GAMS Development Corporation
- Tekorius T., Kildišas V. Optimizavimo pagrindai. Kaunas: Technologija, 2007. 71p.

ASSESSMENT OF GAMS TECHNOLOGIES FOR SOME GLOBAL OPTIMIZATION PROBLEMS

L. Kulbytė

Summary

The purpose of this paper is to analyze the 0/1 Knapsack and The traveling salesperson (or, salesman) (TSP) global optimization problems. The experimental part is done with GAMS optimization system - The General Algebraic Modeling System. The knapsack problem is solved with brute force algorithm and with GAMS solvers. The comparison of both of these methods is given and the best solvers by the time and solution are found in the article.

PATIKIMUMO CHARAKTERISTIKŲ TYRIMAS MECHATRONIKOS GAMINIŲ KOKYBEI NUSTATYTI

Magistrantas Martynas Karkauskas

*Vilniaus Gedimino technikos universitetas,
el. p. shaldytuv@gmail.com*

Anotacija. Nagrinėjama gaminių kokybės ir patikimumo naujausi literatūros šaltiniai. Apžvelgiami pagrindiniai metodai, kurie yra naudojami siekiant užtikrinti gaminių kokybę ir patikimumą. Metodai realizuojami pasitelkiant matematinį paketą Maple, bei automatizuojami. Parengta kompiuterinė technologija mechatronikos gaminių patikimumui ir kokybei nustatyti.

Reikšminiai žodžiai: Statistinė kokybės kontrolė, kokybės kontrolės grafikai, patikimumas

Įvadas

Produkcijos kokybės lygio reguliavimas ir gerinimas yra vienas iš svarbiausių kiekvienos šiuolaikinės įmonės uždavinių, kad išleidžiama produkcija būtų konkurencinga pasaulinėje rinkoje.

Vienas iš pagrindinių produkcijos kokybės lygio valdymo būdų yra kokybės kontrolės sistema, apimanti įeinamąją žaliavų ir detalių kontrolę, tarpinę kontrolę, taikomą įvairiais gamybos etapais, ir gatavos produkcijos išleidžiamąją kontrolę. Gaminant sudėtingus elektroninius prietaisus kontrolės sistemos svarba yra gana didelė tiek sąnaudų požiūriu, tiek poveikiu galutiniam rezultatui. Keičiant kontrolės sistemą gana ženkliai galima keisti ir išleidžiamos produkcijos kokybės lygį, ir savikainą. (Kruopis, Vaišvila, Kalnius 2005)

Statistinė kontrolė ir kokybės valdymas

Siekiant užtikrinti gaminio patikimumą yra naudojama statistinė kokybės kontrolė ir kokybės valdymas. Todėl įmonėje turi būti įgyvendinti šie procesai:

- Konstruktorių dalyvavimas, statistinės kokybės kontrolės ir kokybės valdymo, planavime;
- Technologų dalyvavimas, statistinės kokybės kontrolės ir kokybės valdymo, planavime;
- Atrankinės kontrolės naudojimas gaminio priėmimo stadijoje;
- Atrankinės kontrolės metodų personalo apmokymai;
- Technologinių procesų statistinių kontrolės metodų naudojimas priimant gaminį ir jo sudedamąsias dalis;
- Technologinių procesų statistinių kontrolės metodų, naudojamų priėmimo stadijoje, personalo apmokymai;

- Stabilumo kontrolė ir technologinio proceso reguliavimas pagal kokybės rodiklius gaminant gaminį;
- Stabilumo kontrolė ir ypač svarbių gaminio dalių technologinio proceso reguliavimas, bei specifinių technologinių procesų reguliavimas;
- Technologinio proceso duomenų analizė, siekiant nustatyti vykstančius pokyčius;
- Stabilumo kontrolės ir technologinių procesų reguliavimo personalo apmokymai;
- Koreguojančių veiksnių panaudojimas, kai technologinis procesas peržengia leistinąsias ribas;
- Papildomos kontrolės taikymas atliekant koreguojančius veiksmus;
- Tiekėjų produkcijos (žaliavų, pusgaminių) statistinės kokybės kontrolės vertinimas.

Įvykdžius šiuos punktus yra užtikrinama produkcijos kokybė. (Безъязычный, Замятин, Замятин, Семенов 2003)

Kontroliuojami ir nekontroliuojami procesai

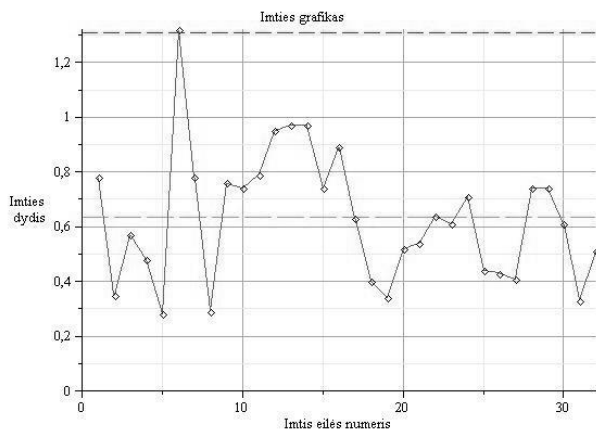
Norint taikyti statistinę kokybės kontrolę, svarbų yra žinoti ar procesas yra kontroliuojamas, nes nuo to priklausos veiksnių eiga taikoma statistinėje kontrolėje. Taigi kontroliuojamu procesu vadinsime tokį procesą, kuriuo stebimos reikšmės neperžengia nustatytų ribų. Žinoma, jog procesas, kurio reikšmės peržengia ribas, nebūtinai gali būti nekontroliuojamas, nes tai sugriauna spėjimą, jog stebimi duomenys yra atsitiktiniai, todėl svarbu išanalizuoti šių reikšmių atsiradimo priežastis. (Berk, Kenneth 2005)

Kokybės kontrolės grafikai

Vienas iš labiausiai paplitusių būdų produkcijos kokybei kontroliuoti yra kokybės kontrolė grafikų pagalba. Tai pasiekama sekant produkcijos atitikimą reikalavi-

mams(standartams). Atliekant atrankinę kontrolę produkcijos kokybės matas yra įrašomas į grafiką. Tokiu būdu yra gaunami kokybės kontrolės grafikai, kurių analizė leidžia daryti išvadas apie gamybos procesą, įrengimų ir technologinių procesų tikslumą, bei priemones, kurias reikia taikyti norint užtikrinti kokybės lygį. Kokybės kontrolės grafikai yra sudaryti:

1. Taškai sudarantys kreivę yra kokybės charakteristikos matas, kurie yra paimti iš gamybos procesų skirtingais laiko momentais(duomenys);
2. Vidurio liniją yra brėžiama, suskaičiuojant proceso charakteristikų vidurkį, kuris randamas iš duomenų;
3. Viršutinė ir apatinė kontrolės ribos, kurios nurodo kontroliuojamų procesų ribas.(Коуден 1961)



1 pav. Imties grafikas (*R-chart*). Kaip matome 6 narys yra už kontroliuojamų ribų.

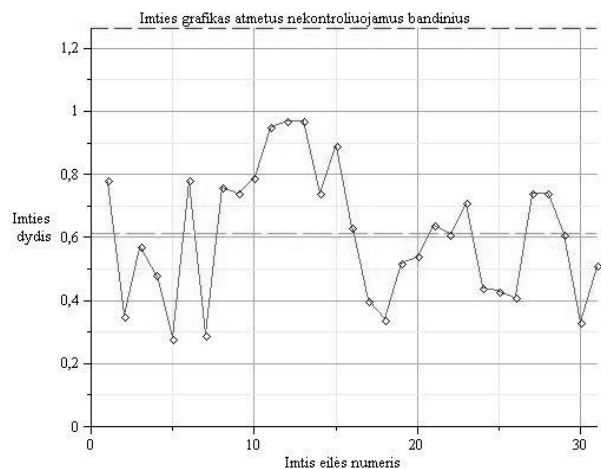
Imties ir vidurkio grafikų panaudojimas kokybės kontrolei

Imties ir vidurkio grafikų metodo žingsniai:

1. Pasirinkti racionalius duomenų pogrupius. Iš kiekvienos gamybos partijos pasirenkamas atitinkamas skaičius tiriamų gaminių skaičius ir jie renkami vienodais laiko intervalais;
2. Sudaryti imties grafiką(*R-chart*);
3. Nustatyti kiekvieno taško, kuris nepatenka į kontroliuojamas ribas, atsiradimo priežastis. Jei tai sukėlė technologinio proceso pokytis ar kiti veiksniai, kurie sukelia nuokrypį iškraipantį tiriamą procesą galima tęsti metodą 4 punktu.
4. Perskaičiuoti imties grafiką, neįtraukiant nekontroliuojamų taškų;

5. Gavus kontroliuojama procesą imties grafike, suskaičiuoti vidurkio grafiką(\bar{x} -chart), atmetant nekontroliuojamus imties grafiko narius;
6. Jei vidurkio grafikas parodo, jog procesas yra nekontroliuojamas, tai reiškia, jog gamyba vyksta nekokybiškai.
7. Toliau tęsiame vidurkio ir imties grafikų stebėjimą įtraukdami naujus gamybos duomenis.(Navidi 2006)

Taigi atlikdami metodo 1,2,3,4 žingsnius iš turimų gamybos proceso duomenų gaunami 1 ir 2 paveikslai, kuriuose apskaičiuojami dydžiai yra randami pagal 1, 2, 3, 4 formules.



2 pav. Imties grafikas atmetus nekontroliuojamus narius. gamybos procesas tampa kontroliuojamu, todėl galime tęsti metodą toliau.

$$\bar{R} = \frac{\sum R_i}{n} \quad (1)$$

\bar{R} - imties dydžių absoliutaus skirtumo vidurkis; R_i - i-tosios imties absoliutus skirtumas; n - imčių skaičius.

$$R_{up} = \bar{R} + 3\bar{\sigma} \quad (2)$$

R_{up} - viršutinis rėžis; $\bar{\sigma}$ - imties standartinių nuokrypių vidurkis.

$$R_{down} = \bar{R} - 3\bar{\sigma} \quad (3)$$

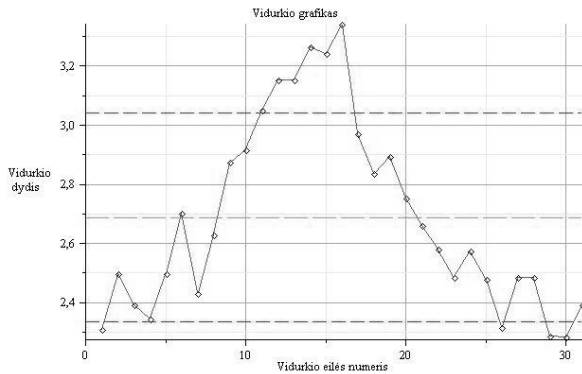
R_{down} - apatinis rėžis.

$$\bar{\sigma} = \frac{\sum \sigma_i}{n} \quad (4)$$

σ_i - i-tasis imties standartinis nuokrypis.

1 lentelė. Gamybos proceso stebėjimo duomenys

Duomenys								
Imties nr.	Imties reikšmės					Vidurkis (\bar{X})	Imties dydis (R)	Dispersija (σ)
1	2,53	2,66	1,88	2,21	2,26	2,308	0,78	0,303
2	2,69	2,38	2,34	2,47	2,61	2,498	0,35	0,149
3	2,67	2,23	2,1	2,43	2,54	2,394	0,57	0,23
4	2,1	2,26	2,51	2,58	2,28	2,346	0,48	0,196
5	2,64	2,42	2,56	2,51	2,36	2,498	0,28	0,111
6	2,64	1,63	2,95	2,12	2,67	2,402	1,32	0,525
7	2,58	2,69	3,01	3,01	2,23	2,704	0,78	0,327
8	2,31	2,39	2,6	2,4	2,46	2,432	0,29	0,108
9	3,03	2,68	2,27	2,54	2,63	2,63	0,76	0,274
10	2,86	3,22	2,72	3,09	2,48	2,874	0,74	0,294
11	2,71	2,8	3,09	2,6	3,39	2,918	0,79	0,32
12	2,95	3,54	2,59	3,31	2,87	3,052	0,95	0,375
13	3,14	2,84	3,77	2,8	3,22	3,154	0,97	0,39
14	2,85	3,29	3,25	3,35	3,59	3,266	0,74	0,267
15	2,82	3,71	3,36	2,95	3,37	3,242	0,89	0,358
16	3,17	3,07	3,14	3,63	3,7	3,342	0,63	0,298
17	2,81	3,21	2,95	3,04	2,85	2,972	0,4	0,16
18	2,99	2,65	2,79	2,8	2,95	2,836	0,34	0,137
19	3,11	2,74	2,59	3,01	3,03	2,896	0,52	0,221
20	2,83	2,74	3,03	2,68	2,49	2,754	0,54	0,198
21	2,76	2,85	2,59	2,23	2,87	2,66	0,64	0,265
22	2,54	2,63	2,32	2,48	2,93	2,58	0,61	0,226
23	2,27	2,54	2,82	2,11	2,69	2,486	0,71	0,293
24	2,4	2,62	2,84	2,5	2,51	2,574	0,44	0,168
25	2,41	2,72	2,29	2,35	2,63	2,48	0,43	0,186
26	2,4	2,33	2,4	2,02	2,43	2,316	0,41	0,169
27	2,56	2,47	2,11	2,43	2,85	2,484	0,74	0,266
28	2,21	2,61	2,59	2,24	2,34	2,398	0,4	0,191
29	2,56	2,26	1,95	2,26	2,4	2,286	0,61	0,225
30	2,42	2,37	2,13	2,09	2,41	2,284	0,33	0,161
31	2,62	2,11	2,47	2,27	2,49	2,392	0,51	0,201
32	2,21	2,15	2,18	2,59	2,61	2,348	0,46	0,231
						$\bar{X} = 2,678$	$\bar{R} = 0,633$	$\bar{\sigma} = 0,225$



3 pav. Vidurkio grafikas

Naudojantis 5, 6, 7, 8 formulėmis sukonstruojame vidurkio grafiką (3 pav).

$$\bar{\bar{X}} = \frac{\sum \bar{X}_i}{n} \quad (5)$$

$\bar{\bar{X}}$ - vidurkių vidurkis; \bar{X}_i - i-tasis vidurkis.

$$\bar{\bar{X}}_{up} = \bar{\bar{X}} + A_2 \bar{R} \quad (6)$$

$\bar{\bar{X}}_{up}$ - vidurkių viršutinis rėžis; A_2 - išskaičiuotinis koeficientas vidurkiams.

$$\bar{\bar{X}}_{down} = \bar{\bar{X}} - A_2 \bar{R} \quad (7)$$

$\bar{\bar{X}}_{down}$ - vidurkių apatinis rėžis.

$$A_2 = \frac{3\sigma_{\bar{X}}}{\bar{R}} \quad (8)$$

$\sigma_{\bar{X}}$ - vidurkių vidutinis kvadratinis nuokrypis.

Panaudojus imties ir vidurkio grafikų metodą, galima nustatyti, ar gamybos procesas išpildo keliamus kokybės reikalavimus. Verta pažymėti, jog grafikų pagalba mes nustatome proceso kontroliuojamumą ir iš to nusprenđžiame apie gamybos kokybę, tačiau negalime nustatyti priežasties kodėl taip įvyksta, todėl yra svarbu nusimanyti ir technologinį gamybos procesą ir bendradarbiaujant kartu su įvairių sričių specialistams yra ištaisomi nuokrypiai ir pasiekama norima kokybė. (Navidi 2006)

Metodo realizacija

Šiuo metu yra ypač aktualu įvairius gamybos procesus automatizuoti, ne išimtis yra ir patikimumo uždavinių sprendimas, todėl yra svarbu pasitelkti šiuolaikines technologijas. Virtualių eksperimentų planavimas ir matematinis modeliavimas yra vienas iš būdų. (Valavičius, 2006)

Metodas buvo realizuotas matematinio paketo Maple pagalba. Buvo lyginama su Microsoft Excel, Matlab ir

Statistica paketų įrankiais, taikomais statistikos uždaviniams spręsti. (Valavičius, Paramonov, Kleinhofs, Labendik 2006)

Lyginamieji bandymai parodė, jog Maple paketo pagalba yra lengviau automatizuojami tokio tipo uždavinių sprendimai. Kitaip tariant, Maple paketo universalumas yra akivaizdus, tačiau jis reikalauja didesnių matematinės statistikos, bei programavimo įgūdžių.

Išvados

1. Parengtas elektroninis metodas, kuris gali būti naudojamas siekiant užtikrinti mechatronikos gaminių kokybę ir patikimumą.

2. Atlikti automatizuoti skaičiavimai pasitelkiant matematinį paketą Maple, gauti ir analizuojami duomenų vizualizavimo grafikai.

3. Parengtas gaminių kokybės metodas pirmą kartą pritaikytas mechatronikos gamybos proceso gaminių kokybei nustatyti pagal pateiktus statistinius duomenis.

4. Tolimesnis darbų tikslas yra sukurti automatizuotą programą, kurioje būtų analizuojami visų gamybos procesų kontrolė ir patikimumas.

Padėkos

Dėkoju doc. Vincui Valavičiui už pagalbą rengiant straipsnį.

Literatūra

- Berk, Kenneth N., . Анализ данных с помощью Microsoft Excel : адаптировано для Office XP / Кеннет Н. Берк, Патрик Лэйри. - Москва: Издательский дом "Вильямс", 2005. - 555 p.
- Kruopis J., Vaišvila A., Kalnius R. Mechatronikos gaminių kokybė. VU Leidykla, Vilnius, 2005.
- Navidi, William.: Statistics for engineers and scientists /William Navidi New York (N.Y.) : McGraw-Hill Higher Education, 2006.. xviii, 869 p.: iliustr. + 1 kompakt. diskas.
- Valavičius V. Matematinis modeliavimas ir eksperimentų planavimas. Vilnius: Technika, 2006. 127 p.
- Valavičius, Vincas; Paramonov, Y.; Kleinhofs, M.A.; Labendik, Y.. New opportunity for aviation transport reliability study offered by MATLAB // Rīgas Tehniskās universitātes zinātniskie raksti. 6. serija. Mašīnzinātne un transports. Dzelzceļa transports. ISSN 1407-8015. 2006. p. 7-17.
- Безъязычный В.Ф., Замятин А.Ю., Замятин В.Ю., Замятин Ю.П., Семенов В.А. Авиадвигателестроение. Качество, сертификация и лицензирование. Москва: Машиностроение.2003. -840psl.
- Кобзарь А. И. Прикладная математическая статистика. Для инженеров и научных работников. Москва: Физматлит. 2006.

Коуден Д. Статистические методы контроля качества. Москва: Физматлит. 1961.

**RESEARCH OF REALIABILITY CHARACTERISTICS
TO DETERMINE THE OUTPUT QUALITY OF
MECHATRONICS**

M. Karkauskas

Summary

Examining the quality of products and the reliability of the latest the literatures sources. Overview of the main methods which are used to ensure product quality and reliability. The methods are realized through the mathematical package Maple, and it is automated. Prepared technology of product reliability and quality .

INTEGRALINĖS DIFERENCIALINĖS SISTEMOS APYTIKSLIŲ SPRENDINIŲ KONSTRAVIMAS

Olga Lavcel

doktorantė

Mykolo Romerio universitetas

el. p. olgal@mruni.lt

Anotacija. Nagrinėjama integralinė diferencialinė sistema, kurios kiekvienos lygties kairioji pusė yra viena nepriklausoma Biurgerso arba Kortevego-de-VriesoS netiesinė lygtis, o dešinioji pusė yra integralinis operatorius. Tokios sistemos gaunamos taikant vidurkinimo metodą silpnai netiesinėms hiperbolinėms sistemoms, modeliuojančioms dujų ir hidrodinamikos, tamprumo teorijos, plazmos fizikos ir kitus reiškinius. Darbo tikslas - sukonstruoti šios sistemos specialaus pavidalo artinius.

Reikšminiai žodžiai: netiesinės diferencialinės lygtys, artinių konstravimas, vidurkinimo metodas.

Įvadas

Darbe nagrinėjama integralinė diferencialinė sistema

$$\begin{aligned} & \frac{\partial U_j(\tau, y)}{\partial \tau} + a_j U_j(\tau, y) \frac{\partial U_j(\tau, y)}{\partial y} + b_j \frac{\partial^2 U_j(\tau, y)}{\partial y^2} + c_j \frac{\partial^3 U_j(\tau, y)}{\partial y^3} = \\ & = \lim_{T \rightarrow \infty} \frac{1}{T} \int_0^T \sum_{\substack{k=1 \\ k \neq j}}^n \sum_{\substack{l=1 \\ l \neq j \\ l \neq k}}^n d_{jkl} U_k(\tau, y + \lambda_{jk} s) \frac{\partial U_l(\tau, y + \lambda_{jl} s)}{\partial y} + e_{jkl} U_l(\tau, y + \lambda_{jl} s) \frac{\partial U_k(\tau, y + \lambda_{jk} s)}{\partial y} ds \end{aligned} \quad (1)$$

su pradinėmis sąlygomis

$$U_j(\tau, y) = U_0 + U_{j1c}^0 \cos y + U_{j1s}^0 \sin y, \quad j=1,2,3,\dots,n. \quad (2)$$

Čia $U_j(\tau, y)$ - ieškoma funkcija, koeficientai

$a_j, b_j, c_j, d_{jkl}, e_{jkl}$ - žinomi skaičiai.

Tokios sistemos gaunamos pritaikius vidurkinimo metodą silpnai netiesinėms hiperbolinėms sistemoms, modeliuojančioms dujų ir hidrodinamikos, tamprumo teorijos, plazmos fizikos ir kitus reiškinius. Nagrinėjama (1) integralinė diferencialinė sistema, literatūroje dažniausiai paliekama neišspręsta, kaip atskiras uždavinys arba buvo sprendžiama skaitiniais metodais (Maslov, V.; Mosolov, P. 1990, Štaras, A. 1977, Krylovas, A., Čiegis, R. 2001).

Darbo tikslas - sukonstruoti specialaus šia sistemą pavidalo artinius.

Literatūros apžvalga

Nagrinėjama sistema buvo gauta straipsnyje (Štaras, A. 1977) ir apibendrinta darbe (Krylovas, A. 1983).

Integralinę diferencialinę sistemą su pirmomis išvestinėmis straipsnyje nagrinėjo (Majda, A.; Rosales, R.; Schonbek, M. 1987, Sharma, V. 2004, Krylovas, A.; Miškinis P. 2007)

Integralinė diferencialinė sistema su antromis ir trečiosiomis išvestinėmis straipsnyje nagrinėjo (Krylovas, A.; Čiegis, R. 2001).

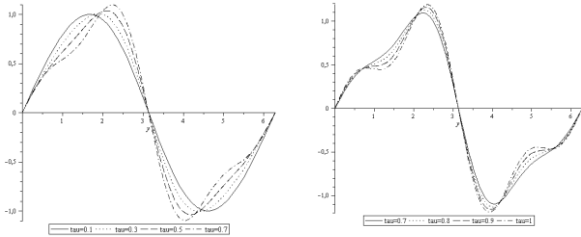
Sistemos tyrimas

Šiame darbe apsiribuosime pirmąja išvestine, t.y. imame uždavinio atskirą atvejį, kai $b_j, c_j = 0, a_j = \pm 1$. Funkcijų periodas 2π .

$$\begin{cases} \frac{\partial}{\partial \tau} V(\tau, y) + V(\tau, y) \frac{\partial}{\partial y} V(\tau, y) = 0 \\ V(0, y) = \sin y \end{cases} \quad (10)$$

Konstruodami artinius, gavome funkciją:

$$U(\tau, y) = (1 - \frac{1}{8}\tau^2 + \frac{1}{192}\tau^4) \sin y + (-\frac{1}{2}\tau + \frac{1}{6}\tau^3 - \frac{7}{240}\tau^5) \sin 2y + (\frac{3}{8}\tau^2 - \frac{11}{128}\tau^4) \sin 3y$$



pav 2. Grafinis sprendimo pavaizdavimas

Spręsdami sistemą tikėjomės sužinoti, kiek reikia imti harmonikų.

Išspręskime (10) uždavinį kitų būdu - nuosekliųjų iteracijų metodu:

$$V^{k+1} = \sin y - \int_0^\tau V^k \cdot V_y^k d\tau$$

$$V^0 = \sin y$$

$$V^1 = \sin y - \int_0^\tau \sin y \cdot \cos y d\tau = \sin y - \frac{1}{2}\tau \cdot \sin 2y$$

$$V^2 = (-\frac{1}{8}\tau^2 + 1) \sin y - \frac{1}{2}\tau \sin 2y + \frac{3}{8}\tau^2 \sin 3y - \frac{1}{12}\tau^3 \sin 4y$$

$$V^3 = (1 - \frac{1}{8}\tau^2 - \frac{1}{64}\tau^4 - \frac{1}{384}\tau^6) \sin y + (-\frac{1}{2}\tau + \frac{1}{6}\tau^3 -$$

$$-\frac{1}{384}\tau^5) \sin 2y + (\frac{3}{8}\tau^2 - \frac{7}{128}\tau^4 + \frac{1}{384}\tau^6) \sin 3y +$$

$$+ (-\frac{1}{3}\tau^3 + \frac{3}{160}\tau^5) \sin 4y + (\frac{65}{384}\tau^4 - \frac{5}{1152}\tau^6) \sin 5y -$$

$$-\frac{43}{640}\tau^5 \sin 6y + \frac{7}{384}\tau^6 \sin 7y - \frac{1}{504}\tau^7 \sin 8y$$

$$V^4 = (1 - \frac{1}{8}\tau^2 + \frac{1}{192}\tau^4 - \frac{13}{3072}\tau^6 - \frac{1003}{368640}\tau^8 - \frac{9359}{22118400}\tau^{10} - \frac{89}{2211840}\tau^{12} - \frac{1}{774144}\tau^{14}) \sin y +$$

$$+ (-\frac{1}{2}\tau + \frac{1}{6}\tau^3 - \frac{7}{240}\tau^5 + \frac{1}{480}\tau^7 + \frac{10771}{6635520}\tau^9 + \frac{773}{3379200}\tau^{11} + \frac{1013}{402554880}\tau^{13}) \sin 2y + (\frac{3}{8}\tau^2 -$$

$$-\frac{27}{128}\tau^4 - \frac{11}{5120}\tau^6 - \frac{13}{7680}\tau^8 - \frac{2873}{7372800}\tau^{10} - \frac{367}{13762560}\tau^{12} - \frac{5}{5419008}\tau^{14}) \sin 3y + (-\frac{1}{3}\tau^3 + \frac{17}{120}\tau^5 -$$

$$-\frac{41}{13440}\tau^7 - \frac{343}{276480}\tau^9 - \frac{19777}{170311680}\tau^{11} + \frac{439}{100638720}\tau^{13}) \sin 4y + (\frac{125}{384}\tau^4 - \frac{835}{9216}\tau^6 - \frac{5}{2048}\tau^8 +$$

$$+ \frac{2263}{4128768}\tau^{10} + \frac{215}{3538944}\tau^{12} - \frac{5}{5419008}\tau^{14}) \sin 5y + (-\frac{39}{160}\tau^5 + \frac{31}{560}\tau^7 - \frac{6479}{3440640}\tau^9 - \frac{397}{28385280}\tau^{11} -$$

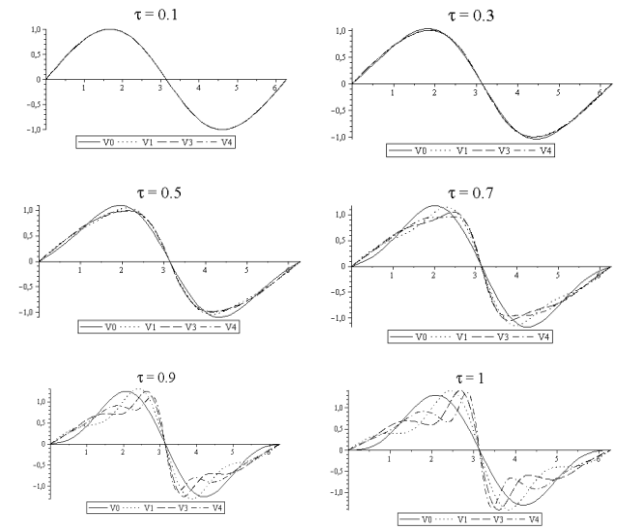
$$-\frac{26836992}{26836992}\tau^{13}) \sin 6y + (\frac{7441}{46080}\tau^6 - \frac{7099}{245760}\tau^8 + \frac{34927}{44236800}\tau^{10} - \frac{79}{1327104}\tau^{12} + \frac{1}{774144}\tau^{14}) \sin 7y +$$

$$+ (-\frac{7873}{80640}\tau^7 + \frac{7523}{552960}\tau^9 - \frac{2743}{10137600}\tau^{11} + \frac{1}{55296}\tau^{13}) \sin 8y + (\frac{29907}{573440}\tau^8 - \frac{179309}{34406400}\tau^{10} +$$

$$+ \frac{11}{107520}\tau^{12} - \frac{1}{602112}\tau^{14}) \sin 9y + (-\frac{153281}{6193152}\tau^9 + \frac{128587}{85155840}\tau^{11} - \frac{5765}{241532928}\tau^{13}) \sin 10y +$$

$$+ \frac{206569}{20643840}\tau^{10} - \frac{6017}{217694720}\tau^{12} + \frac{11}{5419008}\tau^{14}) \sin 11y + (-\frac{929611}{283852800}\tau^{11} - \frac{1801}{33546240}\tau^{13}) \sin 12y +$$

$$+ (\frac{314119}{371589120}\tau^{12} - \frac{65}{16257024}\tau^{14}) \sin 13y - \frac{3091}{19169280}\tau^{13} \sin 14y + \frac{5}{258048}\tau^{14} \sin 15y - \frac{1}{952560}\tau^{15} \sin 16y$$



pav 3. Grafinis sprendimo pavaizdavimas

Pastebėkime, kad sprendžiant lygtį nuosekliųjų iteracijų metodu, mes nežinome, kiek iteracijų reikia skaičiuoti.

Išspręskime (10) diferencialinę lygtį su pradine sąlyga tiesioginiu būdu.

Neišreikštinės funkcijos pavidalu užrašome sprendinį:

$$V(\tau, y) = \sin(y - V(\tau, y)\tau). \quad (11)$$

Patikrinkime, ar tai yra teisingas sprendinys.

Funkcijos $V(\tau, y)$ išvestinė pagal τ

$$\frac{\partial}{\partial \tau} V(\tau, y) = -\frac{\cos(y - V(\tau, y)\tau)V(\tau, y)}{1 + \cos(y - V(\tau, y)\tau)\tau} \quad (12)$$

Funkcijos $V(\tau, y)$ išvestinė pagal y :

$$\frac{\partial}{\partial y} V(\tau, y) = \frac{\cos(y - V(\tau, y)\tau)}{1 + \cos(y - V(\tau, y)\tau)\tau} \quad (13)$$

Irašius (12) ir (13) į lygtį gauname tapatybę:

$$-\frac{\cos(y - V(\tau, y)\tau)V(\tau, y)}{1 + \cos(y - V(\tau, y)\tau)\tau} +$$

$$+ V(\tau, y)\frac{\cos(y - V(\tau, y)\tau)}{1 + \cos(y - V(\tau, y)\tau)\tau} = 0 \quad (14)$$

Išskleiskime funkciją $V(\tau, y)$ Teiloro formule.

Pažymėję $D_j U$ funkcijos j -osios eilės išvestines, gauname:

$$V(\tau, y) = V(0, y) + D_1 V(0, y)\tau + \frac{1}{2} D_2 V(0, y)\tau^2 +$$

$$+ \frac{1}{6} D_3 V(0, y)\tau^3 + \frac{1}{24} D_4 V(0, y)\tau^4 + \quad (15)$$

$$+ \frac{1}{120} D_5 V(0, y)\tau^5 + O(\tau^6)$$

Čia

$$V(0, y) = \sin y$$

$$\frac{\partial}{\partial \tau} V(0, y) = -\frac{1}{2} \sin 2y$$

$$\frac{\partial^2}{\partial \tau^2} V(0, y) = -\frac{1}{4} \sin y + \frac{3}{4} \sin 3y$$

$$\frac{\partial^3}{\partial \tau^3} V(0, y) = \sin 2y - 2 \sin 4y$$

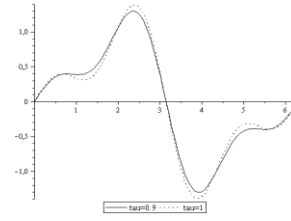
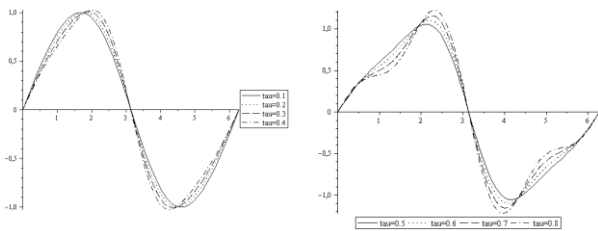
$$\frac{\partial^4}{\partial \tau^4} V(0, y) = \frac{1}{8} \sin y - \frac{81}{16} \sin 3y + \frac{125}{16} \sin 5y$$

Taikome Teiloro formulę:

$$V(\tau, y) = (1 - \frac{1}{8} \tau^2 + \frac{1}{192} \tau^4) \sin y + (-\frac{1}{2} \tau +$$

$$+ \frac{1}{6} \tau^3) \sin 2y + (\frac{3}{8} \tau^2 - \frac{81}{384} \tau^4) \sin 3y - \quad (16)$$

$$- \frac{1}{3} \tau^3 \sin 4y + \frac{125}{384} \tau^4 \sin 5y$$



pav. 4. Grafinis sprendimo pavaizdavimas

Paėmė visas trim budais gautas funkcijas, pastebėkime, kad koeficientai sutampa iki trečios harmonikos τ^4 , o po to skiriasi, priklausomai nuo sprendimo metodo.

Programai testuoti imkime atvejį kai $a = 0$, $b = 0$, $f = 0$, $d = 0$, bei $U_0 = 0$, $V_0 = 0$, $U_{1s0} = 1$, $U_{1c0} = 0$, $V_{1s0} = 1$, $V_{1c0} = 0$.

Gauname sistemą:

$$\begin{cases} U_\tau = -UU_y \\ V_\tau = VV_y \\ V(0, y) = \sin y \\ U(0, y) = \sin y \end{cases} \quad (17)$$

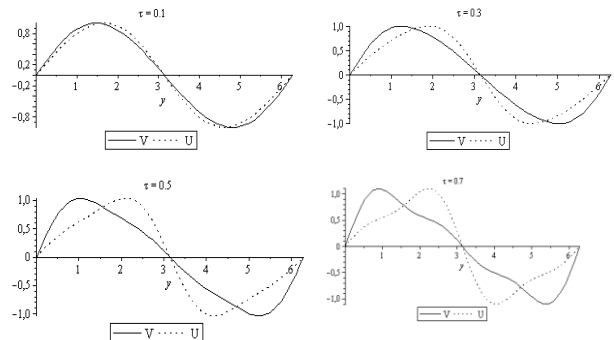
Konstruodami artinius, gavome funkcijas:

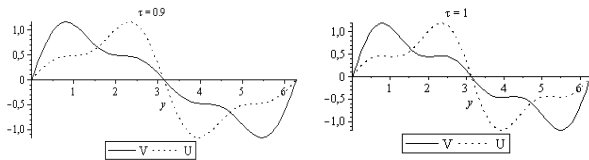
$$U(\tau, y) = (1 - \frac{1}{8} \tau^2 + \frac{1}{192} \tau^4) \sin y + (-\frac{1}{2} \tau + \frac{1}{6} \tau^3 -$$

$$- \frac{7}{240} \tau^5) \sin 2y + (\frac{3}{8} \tau^2 - \frac{11}{128} \tau^4) \sin 3y \quad (18)$$

$$V(\tau, y) = (1 - \frac{1}{8} \tau^2 + \frac{1}{192} \tau^4) \sin y + (\frac{1}{2} \tau - \frac{1}{6} \tau^3 +$$

$$+ \frac{7}{240} \tau^5) \sin 2y + (\frac{3}{8} \tau^2 - \frac{11}{128} \tau^4) \sin 3y \quad (19)$$





pav 5. Grafinis sprendimo pavaizdavimas

Išvados

1. Maple terpėje parašyta programa, skirta integralinės diferencialinės sistemos artinių konstravimui.
2. Atliktas programos testavimas, esant skirtingoms pradinėms sąlygoms.
3. Tolesnis darbas: atlikti programos testavimą rezonansiniu atveju, palyginti gautus rezultatus su žinomais, gautais skaitiniais metodais, nagrinėti naujus integralinės diferencialinės sistemos atvejus.

Literatūra:

- Krylovas, A.; Miškinis P. 2007. Equations of a one-dimensional barotropic gas, (Nauka, Moscow), (Russian).
- Štaras, A. 1977. Asymptotic integration of weakly nonlinear partial differential equations, Dokl. Akad. Nauk SSSR, 237, 525–528; English transl. Soviet Math. Dokl., 18 (1977) 1462-1466 (1978).
- Krylovas, A. 1983. Asymptotic integration of first-order hyperbolic systems, Litovsk. Mat. Sb., 23:4, 12-17; MR, 86g, 35119; Lithuanian Math. J., 23 (1983), 362–366.
- Krylov, A. 1987. A method of investigating weakly nonlinear interaction between one-dimensional waves. J. Appl. Math. Mech. 51, no. 6, 716--722 (1989); translated from Prikl. Mat. Mekh. 51 (1987), no. 6, 933--940(Russian).
- Majda, A.; Rosales, M. 1987. A Canonical System of integrodifferential Equations Aris in Resonant Nonlinearsistemos Acoustics.
- Sharma, V. 2004. Wave interaction in a nonequilibrium gas flow.
- Krylovas, A.; Miškinis P. 2007. Absoliučiai tamprios nesvarios stygos netiesiniai svyravimai. Asimptotikų konstravimas. Liet. matem. rink. 47 spec nr, 123-127.
- Miškinis, P. 2003. Netiesiniai ir nelokaliniai integruojamieji modeliai, Technika.
- Krylovas, A.; Čiegis, R. 2001. Asymptotic Approximation of Hyperbolic Weakly Nonlinear System, J. Nonlinear Math.Phys, 8(4), 458-470.
- Krylovas, A.; Čiegis, R. 2001. Asymptotical analysis of one dimensional gas dynamics equations. Mathematical Modelling and Analysis, 6(1) :117-128.
- Krylovas, A.; Čiegis, R. 2004. Review of Numerical Asymptotic Averaging for Weakly Nonlinear Hyperbolic Waves. Mathematical Modelling and Analysis, 9(3) :209-222.

Krylovas, A. 2008. Asymptotic Method for Approximation of Resonant Interaction of Nonlinear Multidimensional Hyperbolic Waves. Mathematical Modelling and Analysis, 13(1): 47-54.

Padėkos

Dėkoju prof. Aleksandriui Krylovui už pagalbą rengiant straipsnį.

INTEGRATED DIFFERENTIAL SYSTEM OF ESTIMATION DESIGN SOLUTIONS

O. Lavcel

Summary

Examining a differential system is integrated with the left side of each equation are independent of one Burgers or Korteweg-de Vries nonlinear equations, while the right-hand side is an integral operator. Such systems are obtained by the application of the averaging method for weakly nonlinear hyperbolic system, modeling gas and hydrodynamics, elastic theory, plasma physics and other phenomena. The aim of this scheme to construct a special form nearer.

STUDENTŲ ŽINIŲ VERTINIMO SUBJEKTYVUMAS

Rima Kriauzienė

Mykolo Romerio universitetas,
kriauziene@gmail.com

Anotacija. Dėstytojas turi daug pareigų, viena iš jų – įvertinti studentų žinias. Straipsnio autorė tik šiais metais pradėjo dirbti dėstytoja, tai susidūrė su problema, įvertinant studentų žinias. Šia tema yra parašyta nemažai straipsnių, kurių autoriai prieina prie tos pačios nuomonės, kad net pakankamai detalai suderinę vertinimo kriterijus, skirtingi dėstytojai tuos pačius darbus vertina skirtingai. Buvo atliktas tyrimas, kurio tikslas – MRU studentų matematikos žinių tikrinimo rezultatų tyrimas, kai to paties studento darbą nepriklausomai tikrino keli dėstytojai. MRU pirmojo kurso studentai rašė tiesinės algebros kontrolinį darbą, kurį sudarė penki uždaviniai. Tyrimui buvo atrinkti 15 studentų darbų, padarytos jų kopijos. Tuomet šiuos darbus nepriklausomai patikrino 6 dėstytojai. Atlikta bendra ir atskirų užduočių duomenų analizė. Gautos išvados: skirtingų dėstytojų įvertis yra skirtingas, taip pat atkreiptinas dėmesys į studento klaidų analizę: ar jos atsitiktinės, ar tai yra studento tam tikrų žinių, gebėjimų bei įgūdžių trūkumas. Numatomas tolimesnis tyrimas: atskirti studentus pagal pažangumą ir tuomet nagrinėti jų klaidas, kadangi tikėtina, kad mažiau pažangių studentų įvertinimo skirtumų išsi-barstymas būtų didesnis.

Reikšminiai žodžiai: Matematikos dėstymas, žinių vertinimo testai, vertinimo subjektyvumas.

Įvadas

Viena iš dėstytojų pareigų yra savarankiškų darbų sudarymas, jų taisymas ir įvertinimas. Šis darbas užima gana nemažai laiko. Sunkiausi uždaviniai būna du: sudaryti lygiaverčius variantus ir įvertinti darbą. Šios problemos yra ne naujos, nagrinėjamos ne viename straipsnyje, taip pat yra pasiūlyta jų išsprendimo alternatyvų (Krylovas *ir kt.* 2002: 158-162; Krylovas *ir kt.* 2007: 254-258; Krylovas, Raulynaitis 2005: 261-265; Krylovas *ir kt.* 2007: 249-253).

Kadangi šio straipsnio autorė šiais metais tik pradėjo dirbti dėstytoja, tai ir ji susidūrė su šia problema, o beto dar stokoja šioje srityje patirties, todėl su prof. A. Krylovu nutarė atlikti tyrimą. Prof. A. Krylovas ir dar keli straipsnių bendraautoriai daug metų tiria studentų žinių patikrinimo ir įvertinimo problemą, prieina prie tos pačios nuomonės, net pakankamai detalai suderinę vertinimo kriterijus, skirtingi dėstytojai tuos pačius darbus įvertina skirtingai (Krylovas *ir kt.* 2002: 158-162; Krylovas *ir kt.* 2007: 254-258). Straipsniuose autoriai pateikia alternatyvas šioms problemoms išspręsti. Viena iš tokių darbo įvertinimo išsprendimo alternatyvų yra taikyti uždarojo tipo testus su išlygiagretintais atsakymų variantais (Krylovas *ir kt.* 2007: 249-253).

Šio straipsnio tikslas – MRU studentų matematikos žinių tikrinimo rezultatų tyrimas, kai to paties studento darbą nepriklausomai tikrino keli dėstytojai.

Tyrimo eiga

Mykolo Romerio universiteto pirmojo kurso studentai rudens semestro viduryje rašė tiesinės algebros kontrolinį darbą, kurį sudarė penki uždaviniai. Tyrimui buvo atrinkti 15 studentų darbų, kuriems atrinkti atsižvelgėme į reikalavimą, kad darbas būtų pakankamai tvarkingas ir įskaitomas. Padarėme šių darbų kopijas ir paprašėme dėstytojų patikrinti darbus. Esame dėkingi dalyvavusiems tyrime dėstytojams: prof. Jonui Kleizai (VGTU), doc. Mečislavui Meilūnui, doc. Olgai Suboč (VGTU), doc. Hamletui Vladislavui Markšaičiui (MRU), lekt. Kristinai Kaulakytei (VU, MRU), asist. Joanai Kastickaitei (MRU). Jiems buvo pateikti studentų darbai, prie darbo buvo prisegtas lapelis su užduotimis ir prie kiekvienos užduoties nurodytas maksimalus taškų skaičius, suteikiamas teisingai atlikus užduotį.

Pateikiame užduoties pavyzdį:

1. Bankas moka 9% metinių palūkanų, kurias perskaičiuoja kas 4 mėnesiai. Klientas įneša 4000Lt ir kas keturis mėnesius sąskaitą papildo vienoda suma. Po 2 metų sąskaitoje buvo 6963,58 Lt. Kiek litų palūkanų gavo klientas?

4 taškai

2. Apskaičiuokite determinantą:

$$\begin{vmatrix} 3 & -2 & -2 & 3 \\ 2 & 2 & 1 & -2 \\ 4 & 3 & 3 & 4 \\ -2 & 1 & 2 & 3 \end{vmatrix}$$

4 taškai

3. Rasti: $A \cdot B^{-1} \cdot C^T$.

$$A = \begin{pmatrix} 2 & 3 & 1 \end{pmatrix}, B = \begin{pmatrix} 1 & -1 & 0 \\ -1 & 1 & 1 \\ 2 & -1 & 1 \end{pmatrix}, C = \begin{pmatrix} -5 & 2 & 3 \end{pmatrix}.$$

4 taškai

4. Išspręskite sistemą Gauso metodu.

$$\begin{cases} x_1 - 3x_2 + 4x_3 - x_4 = 1, \\ 7x_1 + 3x_2 - 5x_3 + 5x_4 = 10, \\ 2x_1 + 2x_2 - 3x_3 + 2x_4 = 3. \end{cases}$$

5 taškai

5. Išspręskite grafiškai šį tiesinio programavimo uždavinį: $\min(x - 2y)$, kai $\begin{cases} x + 2y \leq 4, \\ x + y \geq 1, \\ x \geq 0, \\ y \geq \frac{1}{2}. \end{cases}$

3 taškai

Surinkus duomenis ir atlikus skaičiavimus, gavome tokius koreliacijos koeficientus:

$r_1 = 0,927$	$r_2 = 0,939$	$r_3 = 0,829$	$r_4 = 0,907$	$r_5 = 0,829$	$r_6 = 0,953$
---------------	---------------	---------------	---------------	---------------	---------------

Matome, kad dėstytojai vertina kontrolinius darbus statistiškai labai panašiai.

tytojui, atimtas rezultatas, vertinant kitam eksperimente dalyvavusiam dėstytojui.

Įrašykime į lentelę gautus skirtumus, kai iš studento rezultato, vertinant darbą jo tikrajam dė-

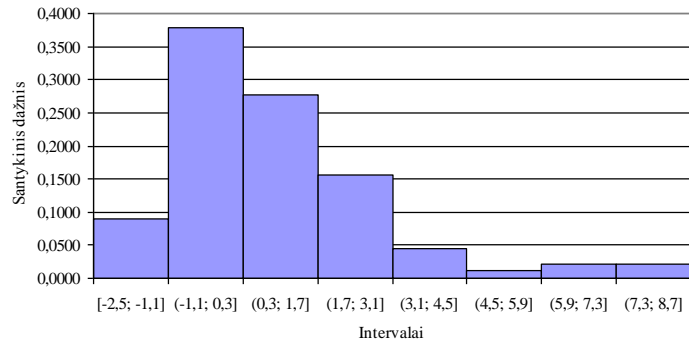
1 lentelė. Vertinimo skirtumų lentelė

Skirtumas	-2,5	-2	-1,75	-1,5	-1,25	-1	-0,5	-0,25	0	0,5	0,75	1	1,5
Dažnis	1	1	1	4	1	5	19	2	8	10	4	9	2

Skirtumas	1,75	2	2,5	3	3,5	3,75	4	4,5	5	5,5	6	7	8,5
Dažnis	2	5	2	5	0	2	2	0	1	0	1	1	2

Iš viso turime 90 skaičių, t.y. 15×6 (15 darbų taisė 6 dėstytojai). Skirtumus gavome tokius, nes dėstytojas užduotį galėjo įvertinti įvairiai: 2t arba 2,25t, arba 2,75t ir t.t. Vertintojas turėjo tik vienintelį apribojimą, t.y. negalėjo viršyti už užduotį skirto maksimalaus taškų skaičiaus.

Šiuos duomenis randame diagramoje sugrupavus į intervalus, kurių ilgis lygus 1,4 (Aksomaitis, 2000). Tuomet duomenų histograma atrodo taip:



1 pav. Skirtumų histograma

Iš histogramos galime pastebėti, kad tikras įvertis buvo kiek didesnis, t.y. tikras dėstytojas vertino kiek nuolaidžiau. Tikimybė gauti

kitą įvertinimą yra 0,52, tai galime pamatyti iš pateiktos lentelės.

2 lentelė. Vertinimo skirtumų lentelė

Įvertinimo skirtumas balais	-1	0	1	2	3	4
Dažnis	13	43	20	9	2	3
Santykinis dažnis	0,14	0,48	0,22	0,10	0,02	0,03

Lentelėje yra parodytas įvertinimo skirtumas balais, t.y. kokio dydžio įvertinimą galėtų gauti studentas, vertinant kitam atsitiktinai parinktam dėstytojui. Tikimybė gauti didesnę įvertinimą yra 0,14, o mažesnę - 0,38, lyginant su tikroju dėstytoju. Tačiau išsibarstymas nėra didelis. Tikimybė gauti mažesnę negu 2 balai įvertį yra tik 0,16. Didžiausias svyravimas yra ties 1 balu, t.y. gauti didesnę 1 balu arba mažesnę 1 balu pažymį. Galime padaryti išvadą, kad skirtingi dėstytojai tų pačių uždavinių klaidas vertina skirtingai (Krylovas *ir kt.* 2002: 158-162; Krylovas *ir kt.* 2007: 249-253), tačiau šie skirtumai nėra labai dideli, nes buvo gana

griežtai apibrėžtos vertinimo taisyklės: prie kiekvienos užduoties buvo nurodytas maksimalus taškų skaičius.

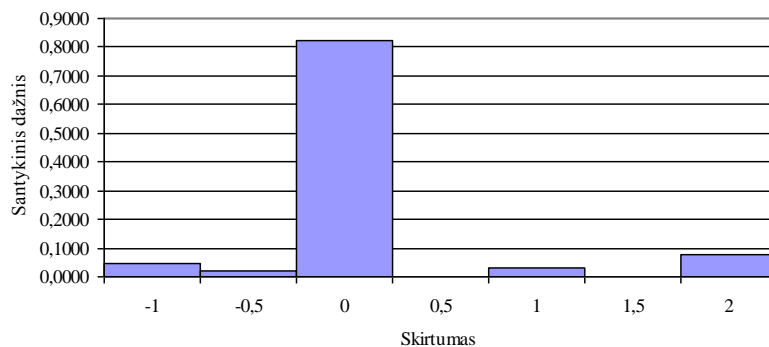
Užduočių tyrimas

1 užduotis. Už šią užduotį studentas galėjo gauti iki 4 taškų. Įrašysime į lentelę skirtumus, kurie gauti, kai iš studento taškų, gautų jo tikrajam dėstytojui vertinant šią užduotį, atimti taškai, gauti vertinant jau kitam eksperimente dalyvavusiam dėstytojui.

3 lentelė. Pirmos užduoties įverčių skirtumai

Skirtumas	-1	-0,5	0	0,5	1	1,5	2
Dažnis	4	2	74	0	3	0	7

Šiuos duomenis pavaizduojame histograma:



2 pav. Pirmos užduoties įverčių skirtumų histograma

Iš histogramos galima pastebėti, jog šią užduotį dėstytojai vertino beveik vienodai, lyginant su tikruoju dėstytoju. Tikimybė gauti kitą pažymį yra tik 0,18. O taip galėjo atsitikti dėl to, kad šį uždavinį daugelis, t.y. 10 studentų iš 15, išsprendė teisingai. Tokių uždavinių kontroliniame darbe reikėtų vengti, nes šis uždavinys neduoda jokios informacijos apie studentų žinias ar spragas. Jis yra per lengvas šiems studentams.

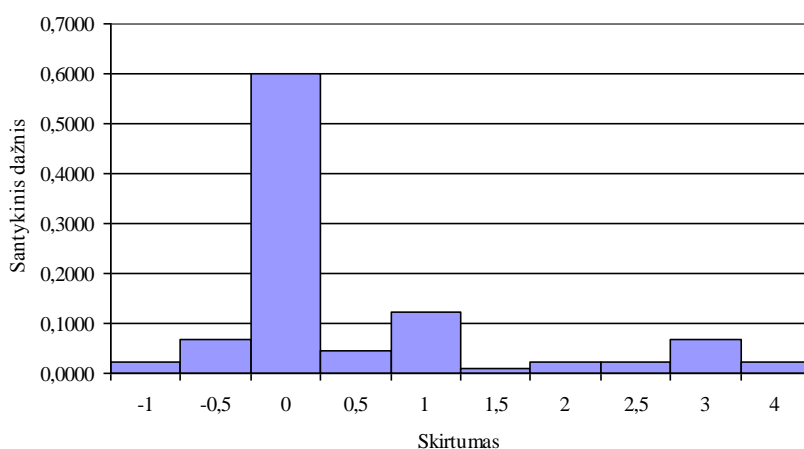
jos apie studentų žinias ar spragas. Jis yra per lengvas šiems studentams.

2 užduootis. Šios užduoties maksimalus galimų surinkti taškų skaičius yra 4. Vėlgi pateikiame lentelę su skirtumais, kurie gauti, kai iš studento taškų, gautų jo tikrajam dėstytojui vertinant šią užduotį, atimti taškai, gauti vertinant jau kitam eksperimente dalyvavusiam dėstytojui.

4 lentelė. Antros užduoties įverčių skirtumai

Skirtumas	-1	-0,5	0	0,5	1	1,5	2	2,5	3	3,5	4
Dažnis	2	6	54	4	11	1	2	2	6	0	2

Šiuos užduoties duomenų histograma atrodo taip:



3 pav. Antros užduoties įverčių skirtumų histograma

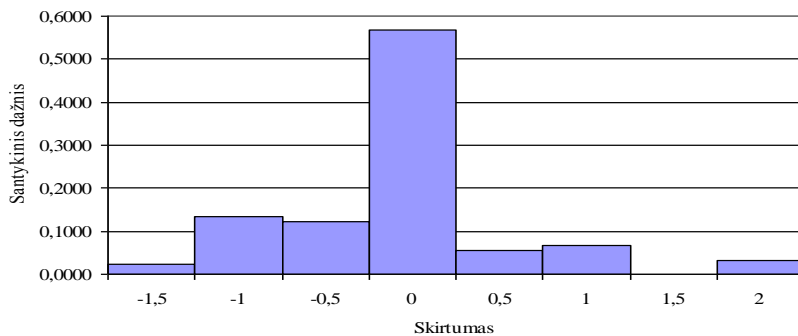
Iš pateiktos dažnių lentelės ir histogramos galime pastebėti, jog gauti kitą įvertinimą už šią užduotį yra 0,4.

3 užduootis. Už užduotį galima buvo surinkti 4 taškus. Pateikiame skirtumų dažnių lentelę:

5 lentelė. Trečios užduoties įverčių skirtumai

Skirtumas	-1,5	-1	-0,5	0	0,5	1	1,5	2
Dažnis	2	12	11	51	5	6	0	3

Šiuos užduoties duomenų histograma atrodo taip:



4 pav. Trečios užduoties įverčių skirtumų histograma

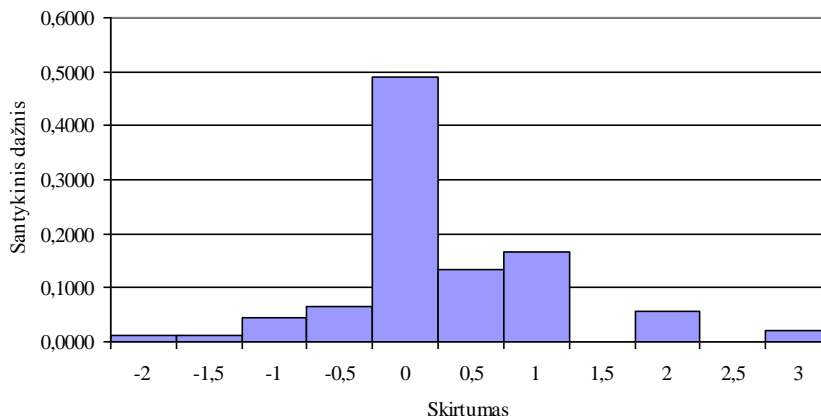
Iš histogramos matyti, jog gauti kitą įvertinimą yra 0,43.

4 uždutis. Užduoties maksimalus galimų surinkti taškų skaičius yra 5. Skirtumų dažnių lentelė atrodo taip:

6 lentelė. Ketvirtos užduoties įverčių skirtumai

Skirtumas	-2	-1,5	-1	-0,5	0	0,5	1	1,5	2	2,5	3
Dažnis	3	5	16	6	30	6	5	6	3	2	2

Duomenis pateikiame histograma:



5 pav. Ketvirtos užduoties įverčių skirtumų histograma

Iš pateiktos dažnių lentelės ir histogramos galime pastebėti, jog gauti kitą įvertinimą už šią užduotį yra 0,67. Kadangi uždavinio sąlyga buvo tokia: „išspręskite sistemą Gauso metodu“ ir maksimalus galimų surinkti taškų skaičius buvo 5, todėl ir išsibarstymas didelis. Kiekvienas dėstytojas tas pačias aritmetines klaidas įvertina skirtingai. Todėl

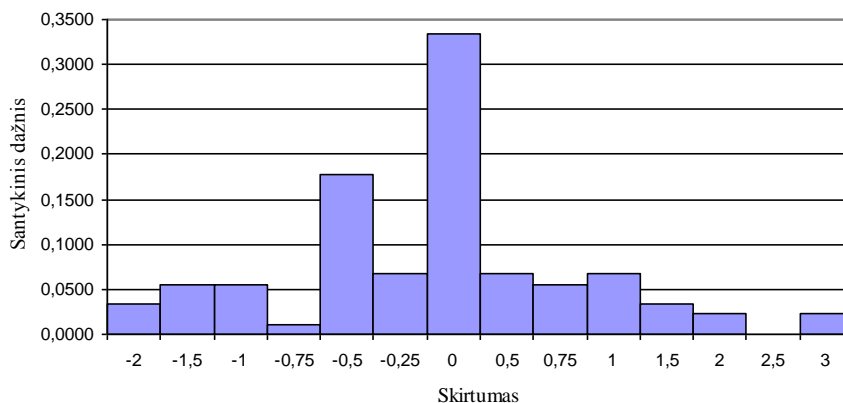
galima būtų sugriežtinti vertinimo kriterijus. Pavyzdžiui, suvedė į trapezijos pavidalą – 2t, suvedamas padarė klaidą, bet iki galo sprendė gerai – 4t ir pan.

5 uždutis. Už šią užduotį galima buvo surinkti 3 taškus. Pateikiame lentelę su skirtumais:

7 lentelė. Penktos užduoties įverčių skirtumai

Skirtumas	-2	-1,5	-1	-0,75	-0,5	-0,25	0	0,5	0,75	1	1,5	2	3
Dažnis	3	5	5	1	16	6	30	6	5	6	3	2	2

Šiuos duomenis pavaizduojame histograma:



6 pav. Penktos užduoties įverčių skirtumų histograma

Matome, jog gauti kitą įvertinimą yra 0,67. Uždavinio sąlyga buvo tokia: „Išspręskite grafiškai šį tiesinio programavimo uždavinį“. Išsibarstymas labai didelis lygiai dėl tos pačios priežasties, kaip ir prieš tai buvusiam uždavinyje. Pavyzdžiui, studentas randa tašką ir teisingai apskaičiuoja funkcijos maksimumą, tačiau sritį pavaizduoja neteisingai. Todėl vieni dėstytojai įvertina maksimumu, kiti 2t, o dar kiti 2,5t.

Tyrimo išvados

1. Atlikus tyrimą pasitvirtino žinomas faktas, jog skirtingų dėstytojų įvertis yra skirtingas. Šiai problemai išspręsti, kaip alternatyva galima siūlyti testus. Lengvesnis problemos išsprendimo būdas – vertinimo kriterijų sugriežtinimas, tačiau tai užima nemažai laiko.

2. Atkreiptinas dėmesys į studento klaidų analizę: ar jos atsitiktinės, ar tai yra studento tam tikrų žinių, gebėjimų bei įgūdžių trūkumas (Krylovas *ir kt.* 2002: 158-162; Krylovas *ir kt.* 2007: 254-258). Čia labiausiai pasireiškia vertinimo subjektyvumas. Todėl tipinių klaidų žinojimas galėtų būti vienas iš vertinimo kriterijų.

3. Numatomas tolimesnis tyrimas: atskirti studentus pagal pažangumą ir tuomet nagrinėti jų klaidas, kadangi tikėtina, kad mažiau pažangių studentų įvertinimo skirtumų išsibarstymas būtų didesnis.

Padėkos

Dėkoju prof. Aleksandrui Krylovui už pagalbą rengiant straipsnį.

Esame dėkingi dalyvavusiems tyrime dėstytojams: prof. Jonui Kleizai (VGTU), doc. Mečislavui Meilūnui, doc. Olgai Suboč (VGTU), doc. Hamletui Vladislavui Markšaičiui (MRU), lekt. Kristinai Kaulakytei (VU, MRU), asist. Joanai Kastickaitei (MRU).

Literatūra

- Aksomaitis, A. 2000. *Tikimybių teorija ir statistika*. Kaunas: Technologija. 347p. ISBN 9986-13-893-0.
- Apynis, A.; Stankus, E. 2001. *Matematika*. Vilnius: TEV. 357p. ISBN-9955-491-08-6.
- Gage, N. L.; Berliner, D., C. 1994. *Pedagoginė psichologija*. Vilnius: Alma litera. ISBN-9986-02-021-2.
- Krylovas, A.; Raulynaitis, J.; Suboč, O. 2002. Dėstytojų nuomonės apie studentų klaidas. *Lietuvos matematikos rinkinys* 46 (spec. nr.): 158 – 162.
- Krylovas, A.; Suboč, O.; Kosareva, N. 2007. Studentų žinių įverčių analizė. *Lietuvos matematikos rinkinys* 47 (spec. nr.): 254 – 258.
- Krylovas, A.; Raulynaitis, J. 2005. VGTU pirmakursių mokyklinės matematikos žinių bei įgūdžių tikrinimas. *Lietuvos matematikos rinkinys* 45 (spec. nr.): 261 – 265.

Krylovas, A.; Suboč, O.; Kosareva, N. 2007. Diskrečiosios matematikos žinių tikrinimo testų lygiagrečiųjų variantų lygiavertiškumo tyrimas. *Lietuvos matematikos rinkinys* 47 (spec. nr.): 249 – 253.

SUBJECTIVE ASSESSMENT OF STUDENT'S KNOWLEDGE

R. Kriauziene

Summary

Teacher has many responsibilities, one of them - to assess student's knowledge. The author only in the following year he began to work the teacher is faced with the problem of assessing student's knowledge. This topic has written many articles, the authors have access to the same opinion that, even in sufficient detail agreed assessment criteria, different teachers of the same work in different ways. The study, which aims - MRU student knowledge of mathematics test results of the investigation, when the same student's work independently verify several teachers. MRU students wrote the first course of linear algebra controlling the work, which consisted of five tasks. Were selected in the sample of 15 students made their copies. The following works independently verified 6 teachers. Done in general and the individual tasks of data analysis. Derived from the findings: different teachers, different estimates are also addressed in the student errors in the analysis: whether they are accidental, or is a student of certain knowledge, skills and skills shortages. Expected next test: separation of students according to progress and then analyze their errors, because it is likely that the less advanced student assessment Tolerance of differences is the greater.

MARŠUTO PAIEŠKOS MODULIO ĮGYVENDINIMAS, KURIANT TURISTINIO MARŠRUTO SUDARYMO INFORMACINĘ SISTEMĄ

Romas Rachmanovas

magistrantas

Vilniaus Gedimino technikos universitetas

Romas.Rachmanovas@fm.vgtu.lt

Anotacija. Straipsnyje aptariamos problemos, su kuriomis susiduriama, kuriant turistinio maršruto sudarymo informacinę sistemą. Išskiriami vieno – paieškos modulio, kūrimo klausimai. Apžvelgiamos technologijos ir aplinkos, kuriose įgyvendinamas paieškos modulis. Kadangi, kuriant internetinę turistinio maršruto sudarymo informacinę sistemą, svarbu tinkamai pasirinkti algoritmą ir tinkamą jo įgyvendinimo būdą, aprašomi kelio maršruto paieškos algoritmo alternatyvos. Pateikiama A* algoritmo apžvalga, algoritmo įgyvendinimas, jau sukurtų technologijų kontekste. Pateikiamos galimos duomenų struktūros, bei sukurto maršruto paieškos modulio integracija su kitomis sistemos dalimis. Pateikiama technologijų analizė ir pateikiami tyrimo rezultatai. Straipsnio pabaigoje pateiktos išvados ir rekomendacijos, kurios būtų naudingos įgyvendinant maršruto paieškos modulį informacinėje sistemoje.

Reikšminiai žodžiai: maršruto, paieškos, modulis, gis, A*, algoritmas

Įvadas

Šiuo metu, kasdieninėje veikloje vis didesnę pritaikymą randa įvairios geografinės informacinės sistemos (GIS) (ang. *geographic information system*). Dauguma jų yra skirtos paprastiems vartotojams, sistemos pasiekiamos internetu ir pateikia įvairiapusę informaciją ir atlieka daugybę funkcijų. Dažniausiai, tokios sistemos naudojamos pramogų, turizmo, pažintinėms reikmėms, tačiau yra ir skirtų verslui, darbui. Šios informacinės sistemos leidžia vartotojui matyti išsamų žemėlapi, tame žemėlapyje matyti įvairius objektus, gauti su jais susijusią informaciją. Taip pat, dažniausiai, yra galimybė vykdyti įvairių objektų paiešką ir matyti jų buvimo vietą žemėlapyje. Taip pat, dauguma tokių svetainių suteikia vartotojui galimybę susiplanuoti maršrutą įtraukiant du arba daugiau norimų vietos taškų, sužinoti trumpiausią, greičiausią ar kitokį kelią.

Nemažai maršruto planavimo sistemų yra įgyvendintos, kaip įprastos stalinės programos. Kaip pavyzdys tokios programos, galėtų būti Microsoft paketas AutoRoute. Tokios ir panašių programų privalumas yra tame, kad nebūtinas interneto ryšys, bei galimybė tiesiogiai jungtis prie globalios pozicionavimo sistemos (GPS) įrenginių. Beje panašaus tipo programų mes galime rasti kiekviename nešiojamame GPS įrenginyje.

Internetu pasiekiamą informacinę sistemą teikia savų privalumų. Tai – galimybė matyti iš karto atnaujintą informaciją, sistemos daugiavartotojiškumas, centralizuotas valdymas, bendras informacijos kūrimas. Įvairių kūrimų priemonių, technologijų ir karkasų gausa leidžia kurti įvairių funkcionalumą turinčias sistemas.

Tokių sistemų kūrimas susiduria su tokiomis pat problemomis, kaip ir kuriant įprastas internetines sistemas. Daugiavartotojiškumas kelia didelius spartos reikalavimus sistemai. GIS maršruto paieškos sistemos greitaveika nemaža dalimi priklauso nuo pagrindinio jo modulio – maršruto paieškos modulio, įgyvendinimo. Tinkamas algoritmo ir jį įgyvendinančių technologijų pasirinkimas įtakoja visos sistemos darbą.

Toliau pateikiama informacija padės kuriant ir įgyvendinant panašias sistemas. Pateikiama analizė ir siūlymai pagrindinio, maršruto paieškos, modulio kūrimui.

Sąlygos įtakojančios algoritmo pasirinkimą

Iki šiol tokių informacinių sistemų plėtrą stabdė patikimų ir detalių geografinių duomenų paketų nebuvimas. Geografinių duomenų paketai buvo ir yra brangus produktas. Ne kiekvienam kuriamam projektui apsimoka finansiškai naudoti komercinius duomenų paketus. Tačiau net ir kai kurių komercinių projektų žemėlapių detalumas nebuvo patenkinamas. Galima prisiminti, kad, dar visai neseniai, dauguma GPS įrenginių neturėjo detalių

Rytų Europos žemėlapių. Tačiau kokią įtaką žemėlapių detalumas turi mūsų nagrinėjamam moduliui? Duomenų kiekis turi tiesioginę įtaką maršruto paieškos algoritmo veikimo greičiui. Kuo detalesnis žemėlapis, tuo daugiau yra įvairių kelio atkarpų, juos jungiančių mazgų, taigi ir didesnis skaičiavimo laikas. Vėl gi, jeigu mes nagrinėsime sistemą, kurioje reikia surasti maršrutą tarp tokių stambių objektų - kaip miestai, tai duomenų detalumas gali būti mažesnis, bei algoritmo pasirinkimas įvairesnis.

Revoliucija tokių sistemų kūrimo padarė - vis didėjantis nemokamų geografinių duomenų pasirinkimas. Atsirado daug tarnybų ir projektų, kurios kaupia ir nemokamai pateikia geografinius duomenis nemokamam naudojimui. Vienas tokių yra – OpenStreetMap (OSM) projektas, kuriame gali dalyvauti kiekvienas žmogus, turintis GPS imtuvą ir norintis tobulinti jau esamų kelių bei gatvių žemėlapi. Projektas sukurtas 2004 metais, šiuo metu yra vienas detaliausių ir tiksliausių panašaus tipo projektų pasaulyje, nenusileidžiantis daugeliui kitų, kad ir komercinių, projektų. Šiam projektui daug informacijos pateikia ir kitos nesiekiančios pelno ir vyriausybės organizacijos. Šiuo metu sukaupti Lietuvos žemėlapio duomenys, taip pat, yra vieni iš detaliausių.

OSM leidžia eksportuoti duomenis XML formatu atsižvelgiant į geografines koordinates, galimas XML bylų parsisiuntimas pagal valstybę, galimas ir visos planetos duomenų parsisiuntimas. Mūsų nagrinėjama sistema apsiribojo Lietuvos masto duomenimis.

OSM turi įvairių priemonių konvertuoti jų duomenis į daugumą erdvinių duomenų bazių valdymo sistemų (DBVS), tad šis duomenų gavimo būdas labai rekomenduojamas, kuriant maršruto paieškos sistemas. 2009 metų balandžio duomenimis, Lietuvos žemėlapi sudarė apie 80000 įvairaus tipo kelio atkarpų, apie 60000 juos jungiančių mazgų, 200 miestų, 4000 kaimiškų vietovių ir priemiesčių ir t.t. Galybė įvairių lankytinų objektų, visuomeninių pastatų, gamtos objektų ir pan. Keliai ir gatvės sužymėtos pagal įvairius parametrus, tokius kaip: kelio danga, greitis, kryptis ir t.t.

Taigi, viena iš modulio įgyvendinimo sąlygų yra duomenų tipas ir jų detalumas, kurį jau apžvelgėme. Kita svarbus aspektas yra sistemos paskirtis ir funkcionalumas. Dažniausiai, norima, kad informacinė sistema gebėtų vykdyti trumpiausio maršruto paiešką per kelis tarpinius mazgus, kad turėtų galimybę pasirinkti norimą kelio dangą ar tipą. Neseniai ištobulėjusios asinchroninio komunikavimo technologijos leido atsirasti ir kitam patogiam funkcionalumui - dinaminei maršruto paieškai. Ko gero, daugelis yra matęs, kaip svetainėje jau apskaičiuotą ir parodytą žemėlapyje maršrutą galima dinamiškai keisti,

paprasciausiai tempiant pelyte paryškintą maršrutą, tokiu būdu atsiranda naujų maršruto taškų ir naujas maršrutas yra nedelsiant perskaičiuojamas. Kaip matysime, net ir šio funkcionalumo reikalavimas gali turėti įtakos algoritmo pasirinkimui.

Dar viena pagrindinė problema, įgyvendinant maršruto paiešką yra sistemos daugiavartotojiškumas. Stalinėje sistemoje, vartotojas gali ir sekundėlę palaukti, kol bus rastas, bei pavaizduotas maršrutas. Tačiau internetinėje sistemoje, vienu metu dirbančių vartotojų gali būti daug ir sistemos veikimo greitumas turi patenkinti vartotojų lūkesčius. Į užklausos atsakymo laiką įeina ne tik maršruto paieška, bet ir žemėlapių generavimas, dinaminis puslapio sukūrimas. Tam naudojamos įvairios spartinančios priemonės ir tai yra plati sritis tyrinėjimams.

Na ir galų gale, svarbi yra pati aplinka ir technologijos, su kuriomis įgyvendinamas maršruto paieškos modulis, bei saugomi gauti duomenis. Tai įtakoja duomenų bazės ir algoritmo programavimo kalbos pasirinkimą. Taip pat yra ir kitų klausimų, susijusių su grafiniu maršruto pateikimu, yra daug priemonių ir technologijų skirtų greitam ir patikimam to atlikimui, bet tai vėl atskira tema. Nereikia užmiršti ir kad protrūkis, kuriant internetines GIS sistemas, atsirado ištobulinius asinchroninio HTTP komunikavimo technologijas. Atkritusi būtinybė - su kiekviena HTTP užklausa perkrauti svetainės vaizdą, įgalino geografinių ir kelio planavimo sistemų atsiradimą interneto erdvėje. Šiuo metu, tokiose sistemose vartotojas gali interaktyviai stumdyti žemėlapi, iš karto matyti dominančią informaciją, dinamiškai keisti norimo maršruto parametrus. Tokios geografinės informacijos ir maršrutų paieškos sistemos reikalauja didelio interaktyvumo, spartaus veikimo.

Pirmiausia, įgyvendinant informacinę sistemą, reikia tinkamai pasirinkti duomenų bazės valdymo sistemą. Yra įvairių pasirinkimo kriterijų, tačiau mūsų nagrinėjamu atveju, svarbu, kad duomenų bazė turėtų erdvinius plėtinius, tai yra - gebėtų apdoroti erdvinius - geografinius duomenis. Tokius plėtinius, arba įskiepius, turi beveik kiekviena moderni DBVS. Žvelgiant į nemokamus produktus, galime vieną sistemą išskirti, tai - PostgreSQL ir jos plėtinys PostGis. Nors, kaip erdvinę DBVS, populiariau naudoti ir kitą nemokamą sistemą – MySQL, tačiau PostgreSQL įdirbis šioje srityje yra didžiausias. Ne paskutinėje vietoje yra ir tas faktas, kad šiai sistemai galima rašyti procedūras C kalba. Tai vėl, kaip matysime turi savo privalumų.

Algoritmo pasirinkimas

Atsižvelgiant į aukščiau aprašytas sąlygas, algoritmui keliamos tokios sąlygos – paieška turi vykti greitai, net ir esant dideliam duomenų kiekiui, bei vykdant užklauską daugeliui vartotojų. Turi būti pasiektas kuo didesnis interaktyvumas. Ir be abejo pirmenybė teikiama tiems algoritmams, kurie pateikia tikslų rezultatą.

Maršruto paieškos, ir bendru atveju, trumpiausio kelio radimo algoritmų yra daug. Visi jie turi savo privalumų ir trūkumų, bei turi savo pritaikymo sritį. Jų aprašymus galima rasti literatūroje, skirtoje grafų uždaviniams tirti.

Vienas labiausiai žinomų, trumpiausio kelio paieškos tarp dviejų taškų algoritmų, yra Dijkstros algoritmas. Šis algoritmas pateikia tikslų paieškos rezultatą, bei plačiai naudojamas kelio paieškos sistemose.

Jo algoritmo vykdymo sudėtingumas:

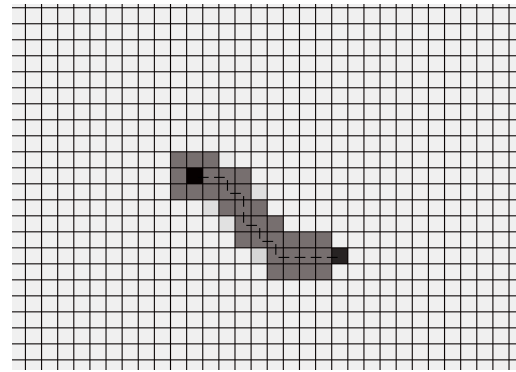
$$O(|E| + |V| \log |V|) \quad (1)$$

ir yra patenkinamas daugeliu taikymo atvejų.

Pasirenkant algoritmą, privaloma atsižvelgti į nagrinėjamų duomenų kiekį. Šiuo atveju, mazgų arba grafo viršūnių kiekį. Grafo viršūnės gali apibrėžti kelio sankryžas, susikirtimus, arba stambesnius objektus, tokius kaip miestai arba net valstybės. Taigi priklausomai nuo to tarp kokių objektų mes vykdome paiešką, priklausys ir grafo viršūnių kiekis. Anksčiau aprašytu pavyzdžiu, su turimais duomenimis – 200 miestų, 4000 kaimiškų vietovių, grafas nesudarys didelis, bei puikiai tiks ir Dijkstros algoritmas. Tačiau smulkinant paiešką ir atsižvelgiant į kiekvieną kelią ar takelį, grafas sparčiai didėja, tuo pačiu ir paieškos vykdymo laikas.

Daugeliui įvairių sričių algoritmų, kurių vykdymo laikas didėja esant didesniai duomenų kiekiui, problematikai panaikinti, sukurti euristiniai algoritmai. Nors jie gali pateikti ir netikslų rezultatą, tačiau tokių klaidingų rezultatų statistinis procentas yra mažas ir jų privalumas yra didelė vykdymo sparta. Ne išimtis yra ir trumpiausio maršruto paieškos sritis. Vienas populiariausių naudojamų algoritmų, trumpiausio maršruto paieškai yra A* euristinis algoritmas. Šis ir jo pagrindu patobulinti algoritmai plačiai naudojami didelėse maršruto paieškos sistemose.

A* euristikos principas yra toks, kad intuityviai aišku, jog trumpiausiais maršrutais turi eiti per sekantį mazgą, kuris yra geografiškai arčiau norimo tikslo, kitais žodžiais tariant, einant kryptimi link tikslo.

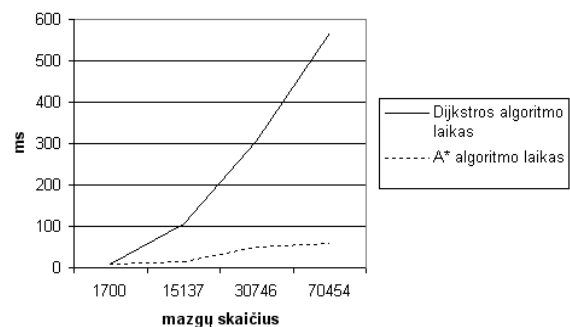


1 pav. A* maršruto paieška

Nesileidžiant į detales, kurias galima rasti šaltinyje (Hart 1972), algoritme naudojamos kelios skaičiavimo reikšmės: tikslus jau surasto kelio atstumas + prognozuojamas atstumas iki galutinio tikslo. Šį prognozuojamą atstumą galima imti įvairių, tačiau geografinėse sistemose priimta imti atstumą, kuris jungia du taškus sujungtus tiesia linija. Čia ir padeda erdvinės DBVS procedūros, gebančios nustatyti atstumą tarp dviejų taškų, išreikštų koordinatėmis.

Šis algoritmas kaip ir Dijkstros, priklauso godžiųjų algoritmų klasei. Kiekvieno žingsnio metu teikiama pirmenybė viršūnei, kurios jau surasto maršruto ilgio ir prognozuojamo atstumo suma mažiausia ir jeigu pasirodo, kad ši suma yra didesnė už kokio nors kaimyninės viršūnės atitinkamą parametą, tada ir imama ši sekanti viršūnė. Šis veiksmo vykdymo laikas, kai išrenkama viršūnė su mažiausia suma, priklauso nuo duomenų struktūros, kurioje saugomas grafas. Ši struktūra tiesiogiai įtakoja algoritmo vykdymo sudėtingumą (beje Dijkstros algoritmo taip pat). Tačiau standartinėse C++ programavimo bibliotekose yra efektyvių, tam skirtų, bibliotekų. Viena tokių yra PriorityQueue.

Žemiau pateiktas palyginimui, įgyvendintų algoritmų vykdymo laikų rezultatai (su OSM duomenimis):

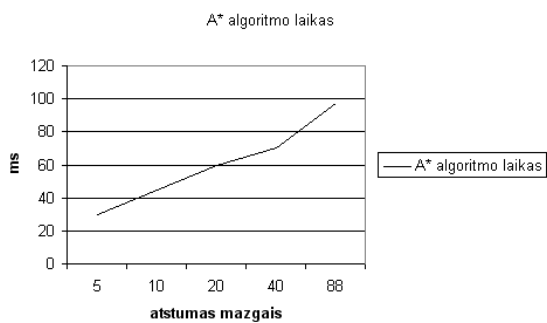


2 pav. Algoritmų vykdymo laikų palyginimas.

A* algoritmo sudėtingumas, skirtingai nuo Dijkstros algoritmo, nepriklauso nuo grafo dydžio, kitaip tariant, duomenų detalumo. Tai taip pat, viena iš svarbių aplinkybių, į kurią derėtų atsižvelgti projektuotojui. Taigi šis tinka labai smulkiems žemėlapiams, kaip pavyzdžiui, pėsčiųjų maršruto paieškai, kur daug įvairių kelių ir takelių. A* algoritmo sudėtingumas yra:

$$O(\log h^*(x)) \quad (2)$$

ir priklauso nuo atstumo tarp ieškomo mazgo ir tikslo. Kuo tikslas yra toliau, tuo paieška užtruks ilgiau. Tačiau ši savybė yra naudinga įgyvendinant dinaminę maršruto paieškos funkcionalumą, aprašytą kiek anksčiau. Tokios dinaminės paieškos įterpiami tarpiniai mazgai pasižymi tuo, kad jie yra santykinai arti jau kitų esančių mazgų. Vadinasi naujas įterptas tarpinis mazgas yra arti jau esančio ir papildomas kelio apskaičiavimas užtrunka nedaug laiko. Taip pat tas liečia ir mazgo pašalinimą.



3 pav. A* veikimo priklausomybė nuo atstumo

Galima bandyti eksperimentuoti su A* euristikos parametrais. Bandant paspartinti algoritmą, galima didinti prognozuojamą paieškos atstumą. Tačiau praktinio įgyvendinimo metu tai nedavė apčiuopiamo pagreitejimo efekto, o klaidų kiekis išaugo. Tad rekomenduotina naudoti euristiką, kuri remiasi tiesioginiu geografinio atstumo apskaičiavimu.

Įdomu tai, kad nors ir A* algoritmas yra euristinis, tačiau įvykdžius per 2000 atsitiktinių maršruto paieškų, nė vienas rezultatas nesiskiria nuo tikslųjų Dijkstros algoritmo sprendinių. Taigi euristika pagal tiesioginį geografinį atstumą yra efektyvi ir tiksli.

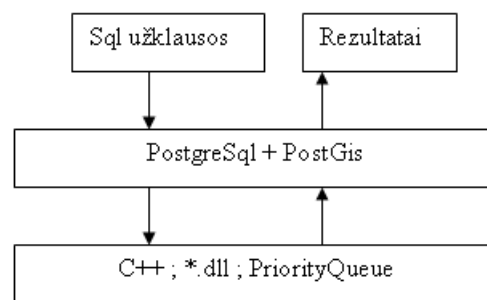
Modulio programavimas ir realizavimas

Sparčiai maršruto paieškai svarbu ne tik algoritmo pasirinkimas ir duomenų kiekis, tačiau ir tinkamas programavimo kalbos ir kitų technologijų pasirinkimas.

PostgreSQL privalumas - patogi ir patikima sąsaja su išoriniais moduliais, parašytais C kalba. Ši DBVS geba aprašyti procedūras, kurios gali komunikuoti su standartinėmis dinaminio susijimo bibliotekomis. Windows terpėje, tai yra DLL standartas, UNIX aplinkoje, tai įprasti *.so archyvai. Dėl šių, o taip pat dėl patikimo PostGis erdvinio plėtinio, ši DBVS labai rekomenduotina projektuotojui, kuriančiam panašaus tipo sistemas.

Taip pat ši DBVS yra gerai dokumentuota ir šaltinyje (PostgreSQL 8.2 Documentation) daug informacijos susijusios su papildomų C kalbos modulių kūrimu.

Žemiau pateikia koncepcinę modulio realizavimo schema



4 pav. Koncepcinė modulio realizavimo schema

Kita svarbi aplinkybė, dėl kurios verta pasirinkti PostgreSQL DBVS yra ta, kad OpenStreetMap projektas yra orientuotas į šios duomenų bazės naudojimą ir turi patikimas priemones duomenų eksportui į šią DBVS. Šios priemonės, pvz.: osm2pgsql, ne tik eksportuoja duomenis, bet ir sukuria visas reikalingas schemas ir lenteles duomenų bazėje.

Dar viena iš problemų, kuriant maršruto paieškos modulį yra ta, kad iš OpenStreetMap importuoti duomenys nėra topologiškai apibrėžti. Tačiau, čia vėl padeda PostGis modulio erdvinės funkcijos. Nesunku, įprastų SQL užklausų ir funkcijų pagalba, duomenis topologiškai sutvarkyti - tam naudojamos erdvinio susikirtimo arba susilietimo funkcijos.

Daugelis paieškai naudojamų parametrų gali būti paskaičiuoti iš anksto ir saugomi duomenų bazėje. Tokie parametrai, kaip atstumas tarp dviejų galų, yra apskaičiuojamas erdvinio funkcijų pagalba ir gali būti išsaugomas duomenų bazėje - specialiai tam sukurtame stulpelyje. Apie tokias erdvinės funkcijas, bei apie topologinį geografinių duomenų skirstymą siūloma plačiau susipažinti projektuotojui.

Dar vienas svarbus įgyvendinimo aspektas yra duomenų apsikeitimas tarp DBVS ir C kalbos modulio. PostgreSQL DBVS išskiria atskirą bendrai naudojamos atminties plotą, iš kurio C kalbos modulis gali pasiekti duomenis net ir vykdydamas kelias užklausas iš eilės. Apie tai plačiau parašyta (PostgreSQL 8.2 Documentation). Praktinio realizavimo metu buvo pastebėta, kad toks duomenų apsikeitimo laikas yra pastovaus dydžio ir yra optimalus, tiek esant ir mažam duomenų kiekiui, tiek ir nagrinėjant didelius duomenų grafus.

Išvados

1. Su esamu duomenų kiekiu, kuriuos gavome importavę OpenStreetMap projekto duomenis, dar galima rinktis kokį algoritmą naudoti. Abu apžvelgti algoritmai veikia sparčiai, bei yra tinkami daudoti.
2. A* puikiai tinka dinaminės paieškos funkcionalumui realizuoti.
3. Reikia atsižvelgti į visą vartotojui pateikiamos informacijos generavimo laiką (trumpiausias kelias, žemėlapiai).
4. Euristicai atsižvelgiant į geografinį atstumą, nagrinėjamu atveju, yra optimali.
5. Paieškos vykdymo laikas svarbus daugiavartotojiškai aplinkai.
6. Projektuotojui, kuriančiam panašią sistemą, rekomenduotinos straipsnyje išvardintos technologijos bei geografinių duomenų gavimo būdas.

Padėkos

Dėkoju dr. Olgai Suboč už pagalbą rengiant straipsnį.

Literatūra

- T. H. Cormen, C. E. Leiserson, R. L. Rivest. Introduction to Algorithms, Second Edition. MIT Press, 2001.
- S. Davis. GIS for Web Developers: Adding 'Where' to Your Web Applications. Pragmatic Bookshelf, 2007.
- Hart, P. E.; Nilsson, N. J.; Raphael, B. 1972. Correction to "A Formal Basis for the Heuristic Determination of Minimum Cost Paths". *SIGART Newsletter* 37: 28–29
- PostgreSQL 8.2 Documentation
[žiūrėta 2009 gegužės 2 d.]. Prieiga per internetą:
<<http://www.postgresql.org/docs/8.2/interactive/index.html>>.
- OpenStreetMap Documentation
[žiūrėta 2009 balandžio 22 d.]. Prieiga per internetą:
<<http://wiki.openstreetmap.org/wiki/Develop>>.

IMPLEMENTING ROUTE SEARCHING MODULE OF ROUTE PLANNING INFORMATION SYSTEM

R. Rachmanovas

Summary

This article describes problems and solutions for implementing route searching module of route planning information system. Article reviews main technologies and algorithms that are suitable for such project designing and programming. Are described the ways for geo data gathering and importing, and what influence this have for choosing algorithm. A* route searching algorithm and suitable data structures are described. Some experimental data are showed. Provided some offers and plans for programming and implementing similar projects.

DIDŽIAUSIO SRAUTO SKAIČIAVIMO ALGORITMŲ REALIZAVIMAS IR ANALIZĖ

Vytautas Sunklodas

magistrantas

Vilniaus Gedimino technikos universitetas

el. p. vsunklodas@gmail.com

Anotacija. Nagrinėjame pagrindinį uždavinį apie srautus tinkluose: tarkime, kad duotas orientuotas, įvertintasis tinklas $G = (V, E)$ ir apibrėžtos šaltinio bei nuotakio viršūnės s, t , tada randame didžiausią galimą srauto dydį. Šiame straipsnyje uždaviniui spręsti naudojamas Fordo ir Falkersono metodas bei Pakėlimo algoritmas. Eksperimentinėje dalyje analizuojami gauti skaičiavimų rezultatai, pateikiami atliktų eksperimentinių skaičiavimų palyginimo rezultatai kai pasirenkami skirtingų tipų ir dydžių tinklai.

Reikšminiai žodžiai: Didžiausias srautas, tinklas, Fordo ir Falkersono metodas, Pakėlimo algoritmas.

Įvadas

Yra daug uždavinių, kuriuose nagrinėjami tinklų srautai. Dažnai yra norima pasirinkti tokias tinklų atkarpas, kad srautas būtų didžiausias. Vienas iš pavyzdžių, tai būtų siekimas perduoti didžiausią informacijos kiekį kompiuteriniuose tinkluose. Kai yra sujungti kompiuteriai, serveriai, spausdintuvai ir kita kompiuterinė technika į bendrą kompiuterinį tinklą ir kai žinome kiekvienos linijos informacijos perdavimo greitį belieka tik pasirinkti duomenų perdavimo maršrutą, tokį, kad iš vieno kompiuterio (šaltinio, angl. *source*) į kitą kompiuterį (gavėją, angl. *target*) galėtumėme perduoti didžiausią informacijos kiekį.

Tokį uždavinį ir daugelį kitų, panašių į šį, galime apibendrintai vadinti kaip didžiausio srauto radimo tinkluose uždavinį. Šiam uždaviniui spręsti yra naudojami įvairūs metodai ir algoritmai, geriausiai žinomi iš jų yra Fordo ir Falkersono (*Ford-Fulkerson*) metodas (Ford, L. R.; Fulkerson, D. R. 1956) bei Pakėlimo algoritmas (Goldberg, A. V. 2008).

Didžiausio srauto radimas tinkluose

Naudojamas tinklas $G = (V, E)$ yra orientuotas, įvertintasis grafas, kurio visos briaunos $e = (u, v) \in E$ yra įvertintos neneigiamais skaičiais, jų svorį žymėsime $c(u, v) \geq 0$ ir vadinsime turimo tinklo atkarpos pralaidumu (Diestel 2005). Išskiriame dvi grafo viršūnes: šaltinį s ir nuotakį t , kur $s, t \in V$:

– pirmoji viršūnė s neturi įeinančių lankų,

– antroji viršūnė t neturi išeinančių lankų.

Sakome, kad srautas tinkle G apibrėžiamas funkcija $f : V \times V \rightarrow R$, kai yra išpildomos šios sąlygos (Čiegis 2007):

Tinklo pralaidumo ribojimas:

$$f(u, v) \leq c(u, v), \quad \forall u, v \in V, (u, v) \in E. \quad (1)$$

Ši sąlyga leidžia garantuoti, kad srautas iš vienos viršūnės į kitą viršūnę neviršys atitinkamos briaunos pralaidumo.

Antisimetriškumas:

$$f(u, v) = -f(v, u), \quad \forall u, v \in V. \quad (2)$$

Ši sąlyga parodo, kad srautas tekantis į priešingą pusę yra lygus, tik su neigiama reikšme. Iš šios sąlygos gauname, kad $f(u, u) = 0$

Srauto tvermės dėsnis:

$$\sum_{v \in V} f(u, v) = 0, \quad \forall u \in V \setminus \{s, t\}. \quad (3)$$

Sąlyga parodo, kad kiekvienos grafo viršūnės išeinančių srautų suma yra lygi nuliui. Pasinaudojant asimetriškumo savybe, lygybę galime užrašyti taip:

$$\sum_{u \in V} f(u, v) = 0, \quad \forall v \in V \setminus \{s, t\}. \quad (4)$$

Funkcija $f(u, v)$ apibrėžia medžiagos kiekį perduodamą briauna (u, v) iš viršūnės u į viršūnę v . Srauto f dydis yra:

$$|f| = \sum_{v \in V} f(s, v) = \sum_{v \in V} f(v, t), \quad (5)$$

t.y. srauto kiekis ištekantis iš šaltinio s yra lygus srauto kiekiui įtekančiam į nuotakį t .

Fordo Falkersono metodas

Fordo ir Falkersono metodo idėja yra labai paprasta: Pirmiausiai surandamas kelias p nuo šaltinio s iki nuotakio t , kuriuo galima pernešti papildomą medžiagos kiekį, tada atitinkamai padidiname srauto funkciją f ir šį procesą kartojame tol, kol egzistuoja toks kelias.

Įvairios šio metodo realizacijos dažniausiai grindžiamos trimis grafų teorijos sąvokomis:

- papildomojo tinklo,
- grafo pjūvio,
- didinančiojo kelio.

Papildomas tinklas. Kai turime tinklą $G = (V, E)$ ir srauto funkciją f , tada randame briaunas, kuriose galime padidinti srauto funkciją:

$$c_f(u, v) = c(u, v) - f(u, v). \quad (6)$$

Tinklo G papildomu tinklu, atitinkančiu srautą f vadinsime tinklą $G_f = (V, E_f)$, kur $E_f = \{(u, v) \in V \times V : c_f(u, v) > 0\}$. Papildomo tinklo kiekvienos briaunos $(u, v) \in E_f$ pralaidumas lygus $c_f(u, v)$.

Grafo pjūvis. Nagrinėjame grafo $G = (V, E)$ pjūvį $C = (S, T)$, $T = V \setminus S$, kai $s \in S$, $t \in T$. Pjūvio pralaidumu vadinsime visų pjūvį kertančių briaunų $D = \{e \in E : e = (u, v), u \in S, v \in T\}$ pralaidumų sumą $c(S, T) = \sum_{e \in D} c(e)$, kadangi turimas grafas yra orientuotas, t.y. grafo briaunos turi kryptį, tai sumuojame tik teigiamus pralaidumus turinčias briaunas.

Srautas per duotąjį pjūvį yra lygus

$$f(C, T) = \sum_{e \in D_f} f(e). \quad (7)$$

Didinantysis kelias. Didinančiuoju keliu, vadinsime bet kokį kelią p papildomajame tinkle $G_f = (V, E_f)$, kuris sujungia šaltinį s su nuotakiu t . Šiuo keliu galima pernešti papildomą medžiagos kiekį, todėl atitinkamai didiname srauto funkciją kiekvienoje kelio briaunoje ir šį

medžiagos kiekį vadinsime kelio p pralaidumu ir žymėsime $c(p)$, jį galime rasti taip:

$$c(p) = \min_{(u, v) \in p} c_f(u, v). \quad (8)$$

Fordo ir Falkersono pasiūlytas metodas yra tik bendras metodas, kurį taikant yra sudaromi didžiausio srauto uždavinio sprendimo algoritmai. Vienas tokių algoritmų yra Edmondso ir Karpo (*Edmund-Karp*) algoritmas (Joseph 2009), kurį ir naudosime norėdami realiizuoti Fordo ir Falkersono metodą.

Edmondso ir Karpo algoritmo sudėtingumas (Joseph 2009) $O(|V||E|^2)$. Kai grafas $G = (V, E)$ pilnasis, tada $|E| = |V|^2$ ir algoritmo sudėtingumas lygus $O(|V|^5)$. Jei grafas labai retas, t.y. $|E| = m|V|$, tai algoritmo veiksmų skaičius $O(|V|^3)$.

Pakėlimo algoritmas

Pakėlimo algoritme laikome, kad kiekviena tinklo viršūnė yra pakelta į tam tikrą aukštį – apibrėžiama aukščio funkcija $h: V \rightarrow \mathbf{N}$, kuri tenkina tokias sąlygas (Addison-Wesley 2005):

$$h(s) = |V|, h(t) = 0, h(u) \leq h(v) + 1, \text{ jei } (u, v) \in E_f.$$

Vykdamas algoritmą, srauto funkcija nebūtinai tenkina masės tvermės dėsnį ir medžiaga gali kauptis viršūnėje, t.y.

$$f(V, u) = \sum_{v \in V} f(v, u) \geq 0, \forall u \in V \setminus \{s\}, \quad (9)$$

bet ir tokią funkciją f vadinsime srautu. Taip pat kiekvienoje viršūnėje $u \in V \setminus \{s\}$ apibrėžiame perteklinį srautą:

$$e(u) = f(V, u). \quad (10)$$

Jei $e(u) > 0$, tai tokią viršūnę $u \in V \setminus \{s, t\}$ vadinsime perpildyta.

Pagrindinė Pakėlimo algoritmo idėja. Skystis gali tekėti tik iš aukščiau pakeltos viršūnės u į žemiau esančią viršūnę v , sujungtą briauna $(u, v) \in E_f$. Pradžioje iš šaltinio pradeda tekėti maksimalus skysčio kiekis, kuris lygus iš šaltinio išeinančių briaunų pralaidumų sumai. Jei kažkurioje viršūnėje u atsiranda perteklinis srautas, tačiau visos briaunos $(u, v) \in E_f$, kuriomis dar galima perkelti papildomą skysčio srautą, jungia u tik su tokio pačio ar didesnio aukščio viršūnėmis, tada u pakeliame į didesnę aukštį ir iš jos išteka perteklinis skysčio kiekis. Jei perteklinio skysčio kiekio nepavyksta perpilti į kaimynines viršūnes, tai jis išpilamas. Tokiu būdu pasiekiame, kad duotojoje viršūnėje galiotų masės tvermės dėsnis.

Vykdamas algoritmą, kai kurių viršūnių aukštis gali tapti didesniu už šaltinio aukštį, tada skystį vėl perpilame į šaltinį, t.y. mažiname bendrą medžiagos kiekį.

Tokiu algoritmu randame didžiausią srautą tinkle.

Stūmimo veiksmas. Iš viršūnės u perpilame tam tikrą kiekį į viršūnę v , jei srautas, tekantis briauna (u, v) , tampa lygiu briaunos pralaidumui, tai toks medžiagos stūmimo veiksmas vadinamas prisotinančiuoju. Ši briauna pašalinama iš papildomojo tinklo.

Stūmimo veiksmą atliekame, kai viršūnė u yra perpildyta ($e(u) > 0$), srautas briauna (u, v) dar neprisotintas ($c_f(u, v) > 0$), ir $h(u) = h(v) + 1$.

Kėlimo veiksmas. Šiuo veiksmu perpildytą viršūnę u pakeliame į didžiausią leistiną aukštį, jei visos briaunos $(u, v) \in E_f$ jungia u tik su žemesnėmis viršūnėmis, t.y. $h(u) \leq h(v)$. Įvykdę kėlimo veiksmą, galime iš u perpilti dalį perteklinio skysčio į kažkurią kaimyninę viršūnę.

Reikia įsitikinti, kad jei viršūnė u perpildyta, tai visada egzistuoja bent viena papildomojo tinklo briauna $(u, v) \in E_f$, kuri jungia u su kita viršūne. Kadangi $f(V, u) = e(u) > 0$, tai turi atsirasti bent viena viršūnė v , tokia, kad $f(v, u) > 0$. Tada papildomajame tinkle briaunos (u, v) pralaidumas yra

$$cf(u, v) = c(u, v) - f(u, v) = c(u, v) + f(v, u) > 0, \quad (11)$$

taigi $(u, v) \in E_f$.

Pakėlimo algoritmo sudėtingumas (Addison-Wesley 2005) yra $O(|E||V|^2)$. Kai grafas $G = (V, E)$ pilnasis, tada $|E| = |V|^2$ ir algoritmo sudėtingumas lygus $O(|V|^4)$. Jei grafas labai retas, t.y. $|E| = m|V|$, tai algoritmo veiksmų skaičius $O(|V|^3)$.

Eksperimentinė dalis

Pasirinkus C++ programavimo kalbą buvo realizuotas Fordo ir Falkersono metodas panaudojant Edmondso ir Karpo algoritmą ir Pakėlimo algoritmas bei atliekami tolimesni bandymai su turimomis programomis. Pirmiausiai buvo sugeneruoti įvairių dydžių tinklai turintys skirtingą taškų ir briaunų skaičių. Ir atlikti eksperimentiniai skaičiavimai su programomis.

Testavimams atlikti buvo pasirinkti tinklai turintys nuo 10 iki 500 viršūnių. Eksperimentiniai skaičiavimai buvo atlikti su pilnais tinklais ir vėliau išretintais (pilnų tinklų briaunų skaičius sumažintas 50% ir 80%). Taip pat pasirinkti tinklų atkarpų pralaidumai: iki 50, iki 500 ir iki

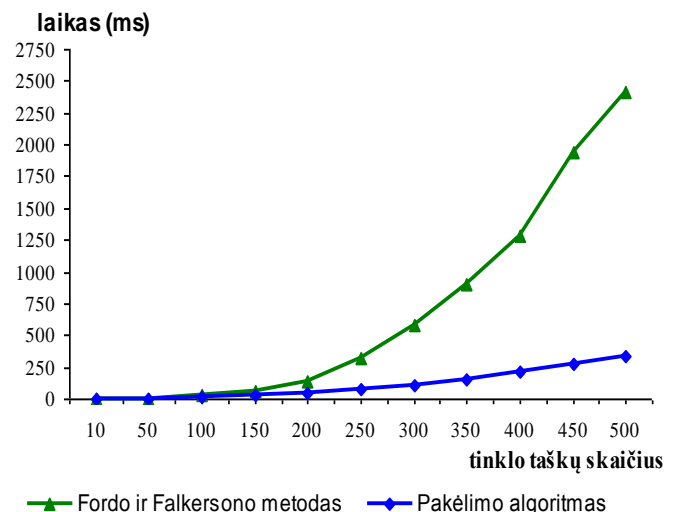
5000. Tačiau keičiant atkarpų pralaidumą programos skaičiavimų atlikimo laikas nebuvo įtakojamas arba įtakojamas labai nežymiai, todėl toliau pateikiami apibendrinti rezultatai visais trim atvejais.

Buvo pasirinkti ir skirtingo dydžio tinklai su fiksuotu kaimyninių viršūnių skaičiumi, kai kiekviena tinklo viršūnė yra sujungta tik su penkiomis tinklo viršūnėmis – taip gaunamas labai išretintas tinklas.

Žemiau matome gautus testavimų rezultatus pateiktus lentelėse bei grafikuose.

1 lentelė. Skaičiavimų atlikimo trukmė, kai pasirinkti pilnieji tinklai

Tinklo taškų skaičius	Taškus jungiančių briaunų skaičius	Skaičiavimo laikas (milisekundėmis)	
		Fordo ir Falkersono metodu	Pakėlimo algoritmu
10	90	<1	<1
50	2450	6	<1
100	9900	25	15
150	22350	63	31
200	39800	140	46
250	62250	328	78
300	89700	578	109
350	122150	906	151
400	159600	1281	218
450	202050	1937	276
500	249500	2409	343



1 pav. Kreivės vaizduojančios Fordo ir Falkersono metodo ir Pakėlimo algoritmo veiksmų atlikimo greitį priklausantį nuo pasirinktų tinklų dydžio, kai nagrinėjami pilnieji tinklai.

Iš gautų eksperimentinių rezultatų (1 lentelė) ir pirmajame paveiksle pateikto grafiko galime matyti, kad kai buvo pasirinkti įvairių dydžių pilnieji tinklai, tokiuose

tinkluose didžiausio srauto radimas Fordo ir Falkersono metodu užtruko daug ilgiau nei sprendžiant Pakėlimo algoritmu.

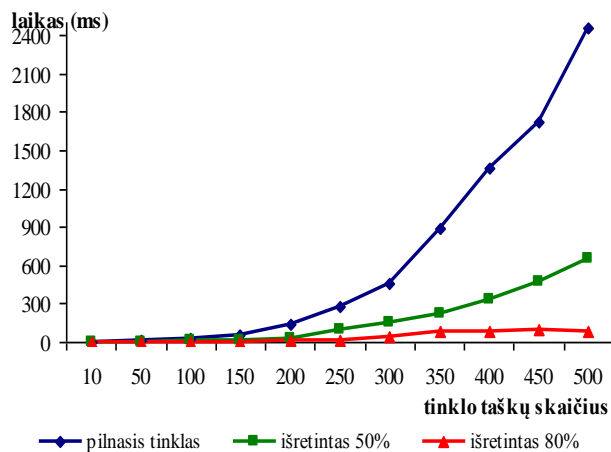
2 lentelė. Fordo ir Falkersono metodo skaičiavimų atlikimo trukmė, kai pasirinkti išretinti tinklai 50%

Tinklo taškų skaičius	Taškus jungiančių briaunų skaičius	Skaičiavimo laikas (milisekundėmis)
10	45	<1
50	1225	<1
100	4950	16
150	11175	15
200	19900	31
250	31125	96
300	44850	156
350	61075	229
400	79800	328
450	101025	479
500	124750	650

3 lentelė. Fordo ir Falkersono metodo skaičiavimų atlikimo trukmė, kai pasirinkti išretinti tinklai 80%

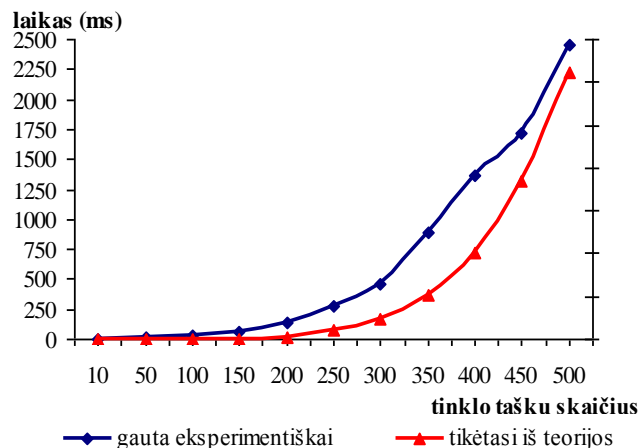
Tinklo taškų skaičius	Taškus jungiančių briaunų skaičius	Skaičiavimo laikas (milisekundėmis)
10	18	<1
50	490	<1
100	1980	<1
150	4470	<1
200	7960	15
250	12450	16
300	17940	47
350	24430	79
400	31920	78
450	40410	100
500	49900	80

Žemiau esančiame grafike pavaizduoti eksperimentų atlikimo metu gauti rezultatai (kurie pavaizduoti antroje ir trečioje lentelėse) su skirtingo dydžio tinklais bei skirtingu išretinimu, kai buvo pasirinkti pilnieji tinklai ir tinklai, kurių briaunų skaičius sumažintas 50% ir 80% (2 pav.).



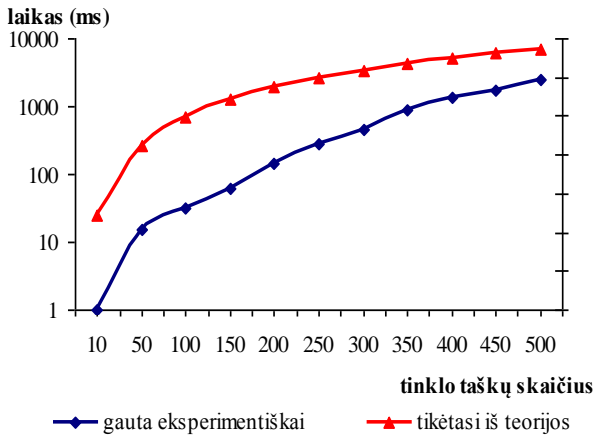
2 pav. Skaičiavimų atlikimo trukmė Fordo ir Falkersono metodu pasirinkus skirtingus tinklus

Žinant teorinį algoritmo sudėtingumą jį galime palyginti su eksperimentų metu gautais rezultatais. Tai pateikiame grafikais pavaizduotais 3, 4, 5 ir 6 paveiksluose.

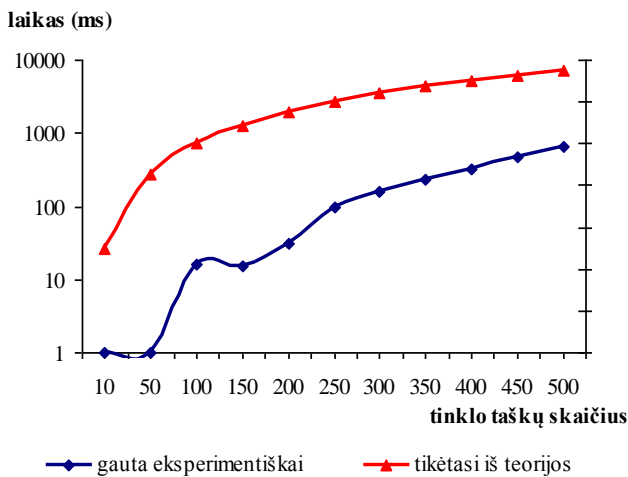


3 pav. Teorinių ir eksperimentinių rezultatų kreivės sprendžiant Fordo ir Falkersono metodu, kai pasirinkti pilnieji tinklai

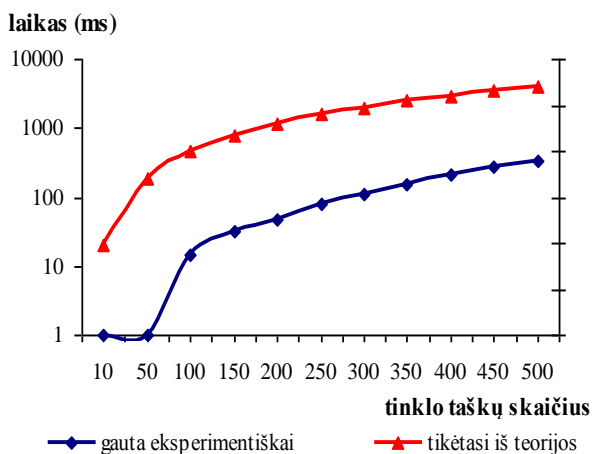
Tuos pačius rezultatus, naudotus trečiajame paveiksle, atvaizdavę logaritminėje skalėje gauname tokį grafiką (4 pav.):



4 pav. Teorinių ir eksperimentinių rezultatų kreivės logaritminėje skalėje, kai pasirinkti pilnieji tinklai ir sprendžiama Fordo ir Falkersono metodu



5 pav. Teorinių ir eksperimentinių rezultatų kreivės logaritminėje skalėje, kai pasirinkti 50% išretinti tinklai ir sprendžiama Fordo ir Falkersono metodu

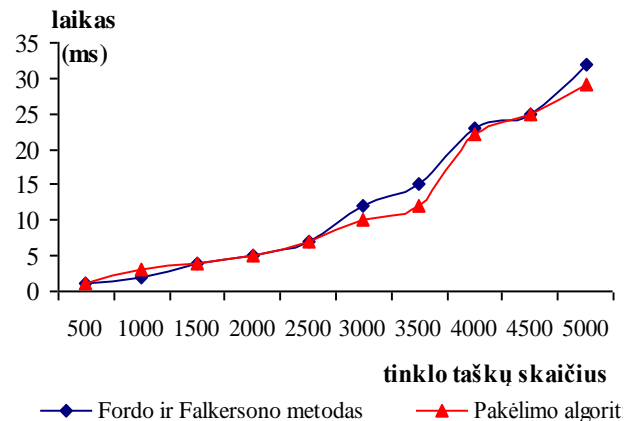


5 pav. Teorinių ir eksperimentinių rezultatų kreivės logaritminėje skalėje, kai pasirinkti pilnieji tinklai ir sprendžiama Pakėlimo algoritmu

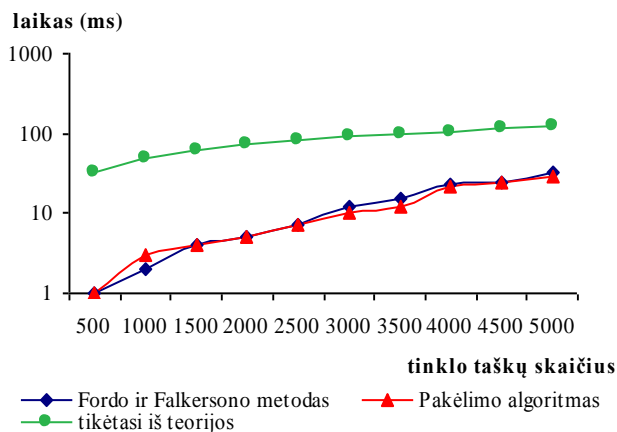
Iš šių gautų eksperimentinių rezultatų atvaizduotų grafikuose (3, 4, 5, 6 pav.) galime pastebėti, kad tiek eksperimentinių tiek iš teorijos gautų kreivių augimo kampai yra labai panašūs ir iš to galime daryti išvadą, kad Fordo ir Falkersono metodas bei Pakėlimo algoritmas, kurie buvo realizuoti programose, ne tik gauna teisingus atsakymus tačiau ir veiksmų atlikimo greitis yra toks koks apibrėžtas teorijoje.

4 lentelė. Skaičiavimų atlikimo trukmė, kai pasirinkti tinklai su fiksuotu kaimyninių viršūnių skaičiumi (kai kiekviena tinklo viršūnė yra sujungta tik su penkiomis tinklo viršūnėmis)

Tinklo taškų skaičius	Taškus jungiančių briaunų skaičius	Skaičiavimo laikas (milisekundėmis)	
		Fordo ir Falkersono metodu	Pakėlimo algoritmu
500	2495	1	1
1000	4995	2	3
1500	7495	4	4
2000	9995	5	5
2500	12495	7	7
3000	14995	12	10
3500	17495	15	12
4000	19995	23	22
4500	22495	25	25
5000	24995	32	29



7 pav. Kreivės vaizduojančios Fordo ir Falkersono metodo bei Pakėlimo algoritmo veiksmų atlikimo greitį priklausantį nuo pasirinktų tinklų dydžio, kai nagrinėjami labai išretinti tinklai.



8 pav. Teorinių ir eksperimentinių rezultatų kreivės logaritmi-
nėje skalėje, kai pasirinkti labai išretinti tinklai.

Iš septintame ir aštuntame paveiksluose pateiktų grafikų galime matyti, kad kai nagrinėjami labai išretinti tinklai, tai tiek Fordo ir Falkersono metodo, tiek Pakėlimo algoritmo veiksmų atlikimo greitis yra labai panašus tarpusavyje ir kreivių augimo kampai artimi teorinės kreivės kylimo kampui.

Išvados

1. Pagal atliktus eksperimentinius skaičiavimus galime teigti, kad didžiausio srauto radimo tinkluose uždavinių sprendimai, kai buvo pasirinkti pilni tinklai, Fordo ir Falkersono metodu užtruko kur kas ilgiau nei Pakėlimo algoritmu, tačiau pasirinkus labai išretintus tinklus, gauti skaičiavimų rezultatai tiek Fordo ir Falkersono metodu, tiek Pakėlimo algoritmu beveik nesiskyrė.

2. Gauti skaičiavimų rezultatai parodė, kad pilnuose tinkluose rasti didžiausią srautą, skaičiavimams sugaištama kur kas daugiau laiko nei išretintuose, kaip buvo teigiama ir teorijoje.

3. Lyginant eksperimentinius skaičiavimus su numatytais teoriniais, galime pastebėti, kad programose realizuoti Fordo ir Falkersono metodas bei Pakėlimo algoritmas „elgiasi“ labai panašiai, t.y veiksmų atlikimo greičiai labai artimi pateiktiems teorijoje.

4. Pagrįstom išvadom padaryti reikia atlikti dar daugiau skaičiavimų eksperimentų su didesniais ir įvairesniais tinklais.

Literatūra

Addison-Wesley, P. 2005. Preflow-Push Algorithm [žiūrėta 2009 balandžio 20 d.]. Prieiga per internetą: <<http://www.cs.sfu.ca/CC/405/ramesh/maxflow3.pdf>>.

Čiegis, R. 2007. Duomenų struktūros, algoritmai ir jų analizė. Vilnius: Technika. 324 p. ISBN: 978-9955-28-109-2.

Diestel, R. 2005. Graph Theory. Springer-Verlag.

Ford, L. R.; Fulkerson, D. R. 1956. Maximal Flow Through a Network. Canadian Journal of Mathematics.

Goldberg, A. V. 2008. The Partial Augment-Relabel Algorithm for the Maximum Flow Problem, in Proc. 16th Europ. Symp. Alg. Springer-Verlag.

Joseph, A. 2009. Edmonds-Karp Algorithm [žiūrėta 2009 balandžio 22d.]. Prieiga per internetą: <<http://www.cse.ohio-state.edu/~tamaldehy/course/794/ek-algo.pdf>>

Skiena, S. 1997. The Algorithm Design Manual [žiūrėta 2009 kovo 30 d.]. Prieiga per internetą: <<http://www.cs.umu.se/kurser/TDBA77/VT06/algorithms/INDEX.HTM>>.

IMPLEMENTATION AND ANALYSIS OF ALGORITHMS FOR DETERMINATION OF THE MAXIMUM FLOW IN GRAPHS

V. Sunklodas

Summary

This paper investigates a key problem in the area of flow networks: given an directed, weighted network $G = (V, E)$ with defined source and sink nodes (s and t, respectively), the goal is to compute the maximum flow in that flow network. This article unitizes the Ford-Fulkerson method and the General push-relabel algorithm to solve the initial problem. The experimental section dissects the results obtained from the calculations and provides a comparative analysis of the experimental calculations executed using networks of different sizes and types.