

DIRBTINĖS BIČIŲ KOLONIJOS ALGORITMAI IR JŲ TAIKYMAS SPRENDŽIANT MARŠRUTŲ OPTIMIZAVIMO UŽDAVINIUS

Donatas Kavaliauskas, Gražvydas Felinskas

Šiaulių universitetas, Matematikos ir informatikos fakultetas

Įvadas

Viena iš aktualių informatikos mokslo sričių yra optimizavimo metodai ir algoritmai. Sprendžiant vis sudėtingesnius optimizavimo uždavinius, svarbu naudoti nestandartinius optimizavimo algoritmus, kurie būtų pagrįsti ne pavieniais sprendiniais. Vienas iš tokių algoritmų – bičių spiečiaus algoritmas, sukurtas mėgdžiojant gamtą. Jame naudojamas kolektyvinis intelektas. Bičių spiečiaus algoritmą ir jo principus pirmieji suformavo Yonezawa, Kikuchi (1995), Lučić, Teodorović (2001), Karaboga (2005), Karaboga, Basturk (2007) [4].

Tyrimo tikslas – išanalizuoti dirbtinės bičių kolonijos algoritmus ir sukurti programinę maršrutų optimizavimo uždavinių sprendimo realizaciją.

Uždaviniai: apžvelgti dalelių spiečių sistemų (angl. *Swarm systems*) algoritmus; išanalizuoti dirbtinės bičių kolonijos (angl. *Artificial Bee Colony* – ABC) algoritmo paradigmas ir jų taikymą; suprojektuoti ABC algoritmo funkcijas ir pateikti šio algoritmo procedūrų praktinį paaiškinimą; sukurti dirbtinės bičių kolonijos algoritmo programinę realizaciją ir ją pritaikyti maršrutų optimizavimo uždaviniams spręsti; atlikti sukurtos programinės įrangos testavimą ir pateikti skaičiavimo rezultatus.

Optimizavimo algoritmai

Optimizavimu vadinama elemento (sprendinio), kurio numatytojo kriterijaus reikšmė būtų minimali (arba maksimali), paieška apibrėžtoje uždavinio aibėje. Optimizavimo uždaviniams spręsti sukurta daug algoritmų. Tikslieji optimizavimo algoritmai garantuoja optimalų sprendinį. Tačiau surasti sprendinį galima tik mažos dimensijos uždaviniuose, nes bandant patikrinti visus galimus sprendinius labai išauga laiko ir kitų išteklių sąnaudos [7], [8].

Pagal euristinius algoritmus kur kas greičiau randamas apytikslis sprendinys (dažniausiai nuo optimalaus besiskiriantis vos keliais procentais). Tokie algoritmai grindžiami taisyklių rinkiniais, kurie vadinami euristikomis. Euristikos pritaiko algoritmą prie konkretaus optimizavimo uždavinio su apribojimais ir prielaidomis.

Metaeuristiniais metodais aprašoma kurios nors klasės uždavinių sprendimo idėja, principas. Taigi, metaeuristiniai algoritmai pranoksta euristinius algoritmus, t. y. tai, kas yra išvystyta, modernizuota tradiciniais euristiniais algoritmais. Optimizavimo uždaviniams spręsti naudojami šie metaeuristiniai algoritmai: lokioji paieška, tabu paieška, genetiniai algoritmai, godžiosios randomizuotos adaptyvios paieškos procedūros (GRASP), skruzdžių kolonijų elgsenos imitavimo algoritmai, paieška kintamoje aplinkoje, bičių kolonijų elgsenos imitavimo algoritmai ir kiti. Jie taikomi sprendžiant įvairius optimizavimo uždavinius: tvarkaraščių ir gamybos procesų optimizavimo,

maršrutų optimizavimo, logistikos, tinklo paskirstymo optimizavimo, automobilių srautų sankryžose optimizavimo ir kitus [5], [7], [8].

Naudojant panašias optimizavimo metodikas galima spręsti daug įvairių uždavinių. Šiame straipsnyje aprašoma, kaip pavyko dirbtinių bičių spiečiaus algoritmą realizuoti keliaujančio pirklio uždavinių grupėje [5], [6], [9].

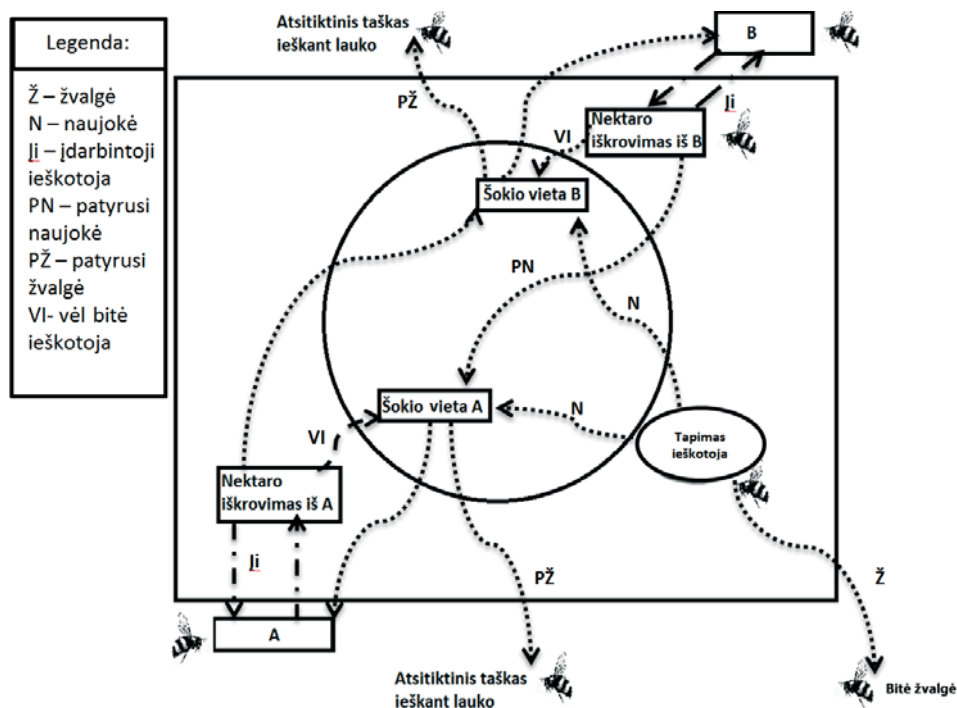
Kolektyvinis intelektas

Algoritmai, grindžiami manipuliavimu tarp sprendinių rinkinių ar grupių, šiuolaikinėje literatūroje vadinami kolektyvinio intelekto algoritmais (Bonabeu, 1999, Karaboga, Akay, 2009). Išskiriami tokie algoritmai: dirbtinis bičių spiečiaus algoritmas, skruzdžių kolonijos optimizavimo algoritmas, krioklio algoritmas, minios metodas ir kiti. Tokių metodų pagrindas – paprastų gyventojų, agentų, bičių sąveikavimas vienas su kitu ir su savo aplinka. Veikėjai, sąveikaudami tarpusavyje, dalijasi informacija, ją apdoroja, pasiskirsto funkcijas kolektyve. Tai yra svarbiausi veiksniai, leidžiantys siekti geriausių įmanomų rezultatų sprendžiant optimizavimo uždavinius [2], [4], [12].

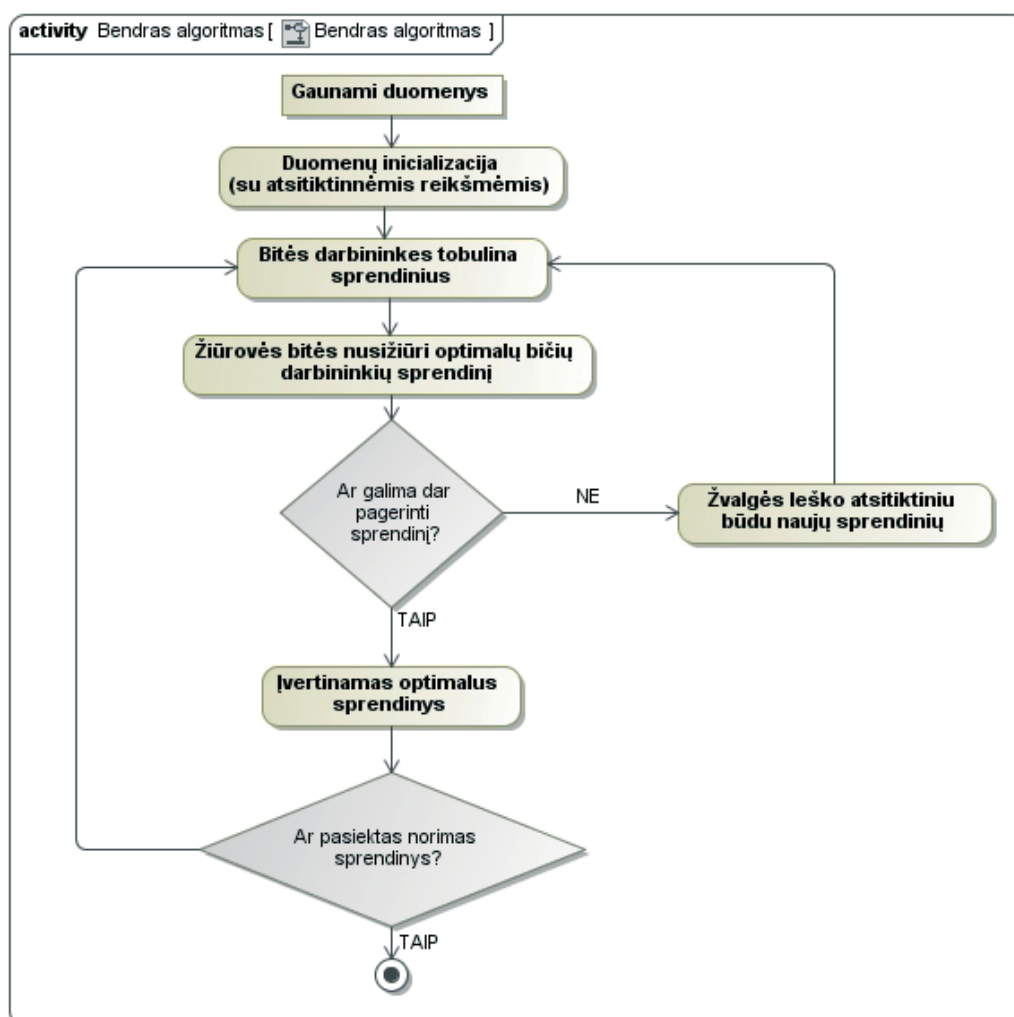
Bičių elgesys gamtoje

Bitės gamtoje veikia kaip dinamiška sistema, jos renka informaciją iš aplinkos ir pagal ją pritaiko savo elgesį. Kol renkama informacija ir bandoma prisitaikyti prie aplinkos, bitės neatlieka visų savo grupės užduočių. Visas socialinės vabzdžių kolonijos elgesys priklauso nuo darbų, kurie susiję su jų morfologija, pasidalijimo. Pagrindiniai bičių kolonijos komponentai (1 pav.):

- *Maisto šaltiniai* (angl. *food sources*). Maisto šaltinio vertė priklauso nuo įvairių parametrų – atstumas iki lizdo ar avilio, maisto kiekis.
- *Ieškotojos* (angl. *foragers*). Ieškotojos yra trijų rūšių:
 - *neįdarbintosios ieškotojos* (angl. *unemployed foragers*): jei manoma, kad bitė neturi informacijos apie maisto šaltinio vietą, ji laikoma neįdarbintąja bite. Yra dvi neįdarbintųjų bičių rūšys:
 - *bitės žvalgės* (angl. *scout bee*): tai bitės, kurios pradeda ieškoti maisto šaltinio atsitiktinai,
 - *bitės naujokės* (angl. *recruit*): tai bitės, kurios neturėdamos darbo skrenda į šokio vietą, kad gautų iš kitų bičių informacijos apie maisto šaltinius. Šios bitės nusikopijuoja informaciją ir pagal ją pradeda ieškoti maisto;
 - *įdarbintosios ieškotojos* (angl. *employed foragers*): kai naujokės bitės randa maisto šaltinį, jos yra paaukštinamos iki įdarbintųjų ieškotojų, kurios įsima kelią iki maisto šaltinio. Kai bitė paima dalį nektaro, kurį gali panešti, ji grįžta į avilį padėti



1 pav. Tipiškas bičių maitinimosi elgesys [4]



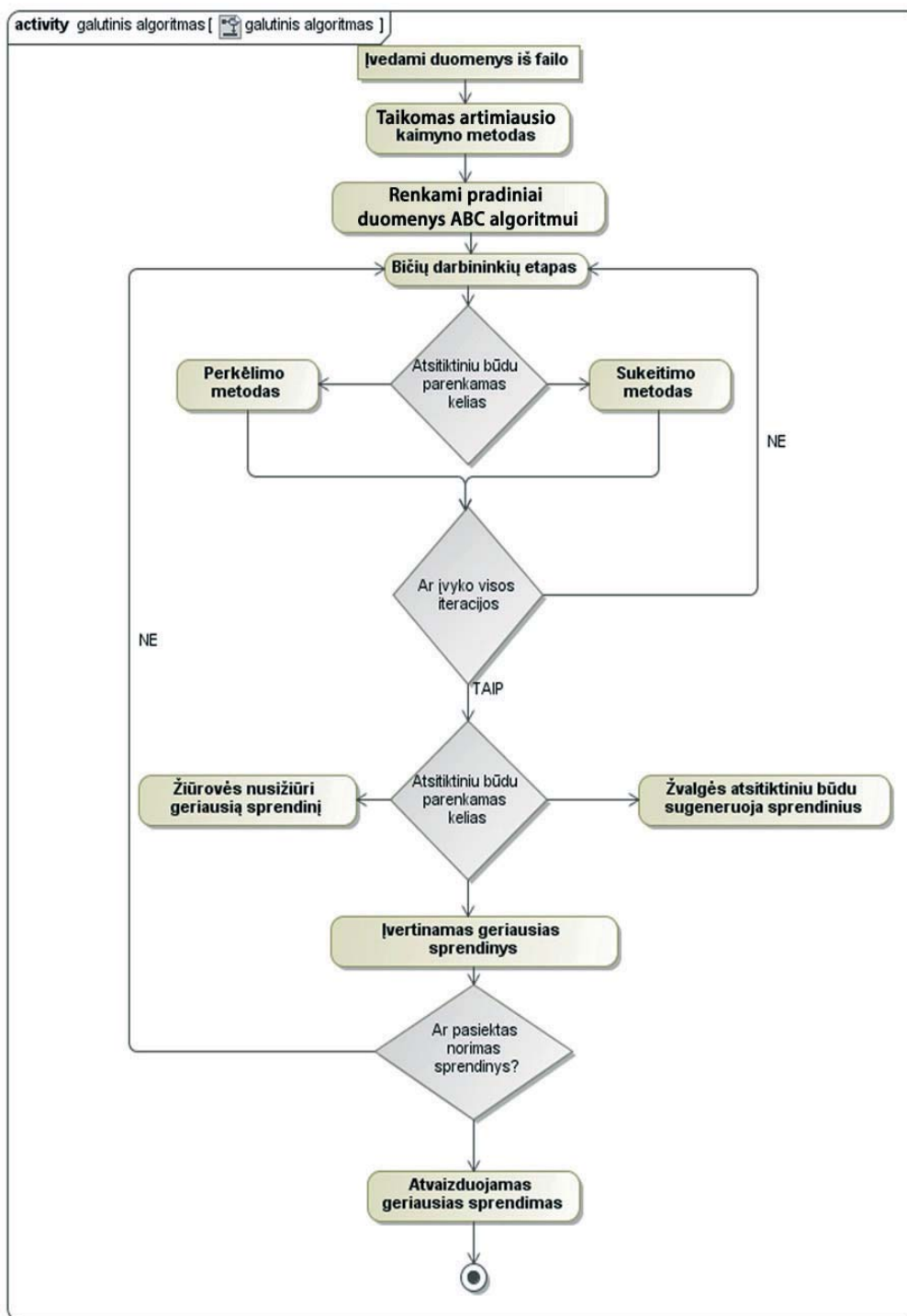
2 pav. ABC algoritmas

nektaro maisto saugykloje. Yra trys variantai, susiję su maisto šaltinio vieta:

- jei likusio nektaro kiekis yra mažas ar jo nebėlikę, bitė pamiršta kelią į šį šaltinį ir tampa neįdarbintąja bite;
 - jei likusio nektaro kiekis yra pakankamas, bitė gali toliau skristi rinkti maisto, nesidalydama informacija apie maisto šaltinį su kitomis bitėmis;
 - bitė gali skristi į šokio vietą ir šokdama informuoti kitas bites apie maisto šaltinį.
- *Patyrusios ieškotojos* (angl. *experienced foragers*). Šio tipo bitės naudoja savo atmintį:

- jos gali būti *inspektorėmis*, kontroliuojančiomis neseniai surastų maisto šaltinių būklę;
- jos gali vėl tapti *ieškotojomis*, besinaudojančiomis informacija iš šokio. Šios bitės bando nustatyti to paties maisto šaltinio kokybę pagal kitų bičių informaciją;
- jos gali tapti *žvalgėmis* ir ieškoti naujų kelių, jei visi maisto šaltiniai išseko;
- jos gali tapti *naujokėmis*, kurios ieško naujų maisto šaltinių pagal kitų bičių šokį [4].

Pagal bičių elgseną gamtoje kuriami įvairūs optimizavimo metodai. Išskiriami algoritmai, kuriuose



3 pav. Modifikuotas ABC algoritmas, pagal kurį sukurta programinė realizacija

atsižvelgiama į maitinimosi elgseną, poravimosi elgseną, bičių karalienę ir į kitus ypatumus. Bičių algoritmus savo moksliniuose darbuose nagrinėjo daug įvairių autorių [4]. Be šio darbe nagrinėjamo ABC algoritmo, dar galima paminėti bičių kolonijos optimizavimo (angl. *Bee Colony Optimization*) algoritmą [11] ir bičių poravimosi optimizavimo algoritmą (angl. *Honey-bee mating optimization algorithm*, HBMO). Pastarąjį sudaro viena karalienė (angl. *queen*), keletas tūkstančių tranų (angl. *drone*), bitės darbininkės (angl. *workers*) ir trano bei karalienės jaunikliai (angl. *brood*) [1], [3].

ABC algoritmas

Dirbtinių bičių spiečiaus algoritmas (ABC) yra optimizavimo algoritmas, grindžiamas tam tikru manipuliavimu sprendinių rinkiniais, grupėmis. Kadangi ABC algoritmas priklauso metaeuristinių optimizavimo metodų pogrupiui, jis negarantuoja, kad bus rasti optimalūs sprendiniai. Tačiau šis algoritmas leidžia gerokai sumažinti skaičiavimų, o tai aktualu sprendžiant didelės dimensijos uždavinius su dideliais duomenų kiekiais, nes padeda taupyti kompiuterinių skaičiavimų išteklius, o gaunami sprendiniai dažniausiai yra aukštos kokybės, t. y. artimi optimaliam sprendiniui [4].

Bičių spiečiaus algoritmas susideda iš trijų bičių grupių: *darbininkės* (angl. *employed bees*), *žiūrovės* (angl. *onlookers*) ir *žvalgės* (angl. *scouts*).

Žiūrovės ir žvalgės taip pat gali būti vadinamos nedirbančiomis bitėmis. Darbininkės ieško geresnio kelio iki maisto šaltinio nei prieš tai žinojo. Jei jos randa geresnį maisto šaltinį, įsimena kelią ir grįžta į šokio vietą pasidalyti informacija apie maisto kiekį su žiūrovėmis. Visos žiūrovės laukia darbininkių šokio, kad sužinotų maisto šaltinio vietą ir jo kiekį. Jos išanalizuoja visų šokusių bičių informaciją apie maisto šaltinius ir atsirenka geriausią šaltinį. Darbininkės gali nuspręsti tapti žvalgėmis. Tada jos bando atsitiktinai ieškoti naujo kelio prie neatrastų maisto šaltinių [2], [4], [9], [10], [12]. Pagal tai galima sudaryti ABC algoritmą (2 pav.).

Sistemos projektavimas ir programinė realizacija

Atlikus ABC algoritmo ir keliaujančio pirklio uždavinio analizę, atliktas sistemos projektavimas ir programinė realizacija (3 pav.).

1. Pradiniai duomenys pateikiami iš tekstinio failo, nes ABC algoritmas yra skirtas dideliame kiekiui duomenų.
2. Pradiniai duomenys ABC algoritmui sugeneruojami artimiausio kaimyno metodu, kuris yra euristinis algoritmas, skirtas keliaujančio pirklio uždaviniui spręsti.
3. Geriausi ABC algoritmo ir artimiausio kaimyno metodo sprendiniai atvaizduojami greta vienas kito, kad vartotojas galėtų įvertinti jų skirtumus.
4. Abu algoritmai atvaizduojami grafiškai, kad vartotojui būtų lengviau suprasti sprendinius.
5. Vartotojui leidžiama keisti ABC algoritmo parametrus: bičių spiečiaus dydį, iteracijų skaičių ir kitus.
6. ABC algoritmo darbininkių etape optimizuojant sprendinį realizuotos perkėlimo (angl. *Shift*) (4 pav.) ir sukeitimo (angl. *Swap*) (5 pav.) funkcijos, t. y. vieno

15	57	14	22	72	79	26	56	42	40
----	----	----	----	----	----	----	----	----	----

14	15	57	22	72	79	26	56	42	40
----	----	----	----	----	----	----	----	----	----

4 pav. Perkėlimo funkcija

Parametrai:

Iteracijų skaičius:
300

Bičių spiečiuje skaičius:
100

Bičių darbininkių etapas:
70

Nusižiūrincios į bičių, kurios taps žvalgėmis arba žiūrovėmis, procentas
20

Uždaryti

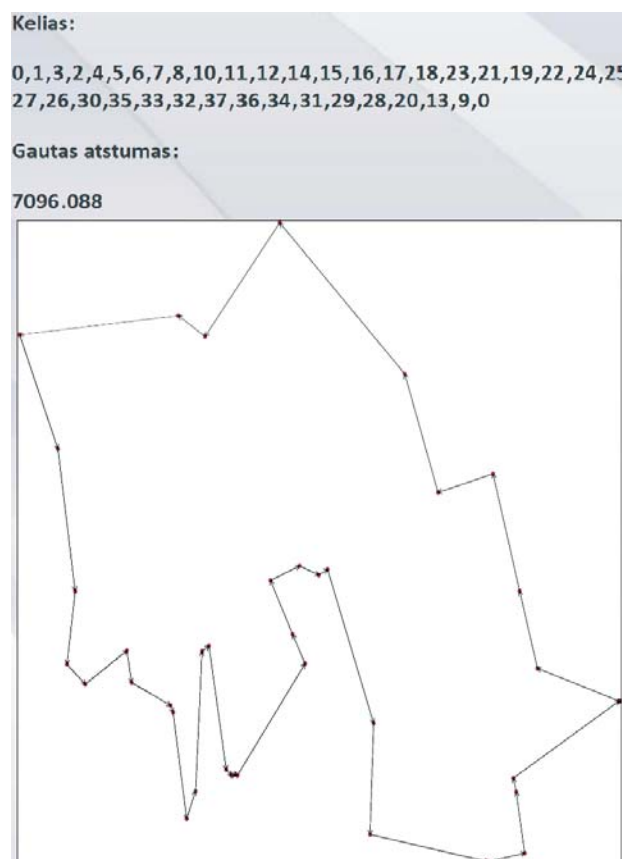
Parametrai

Pasirinkite fail

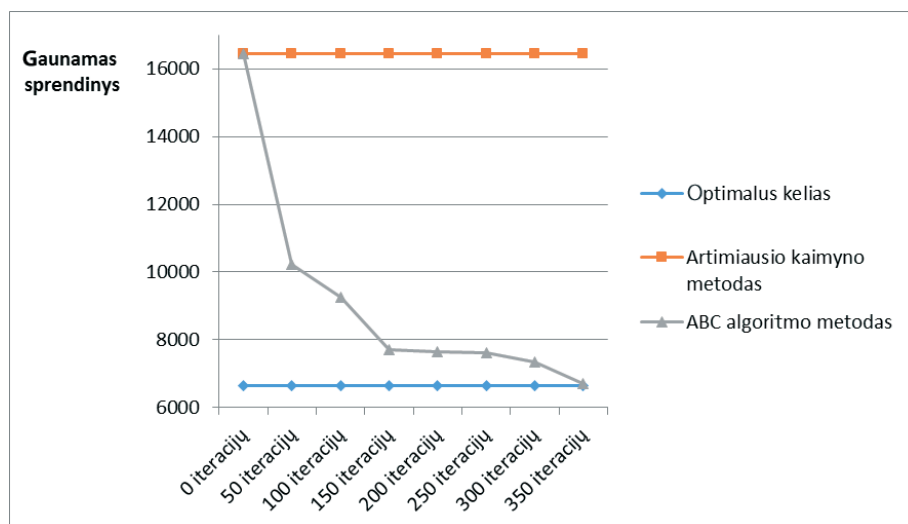
Pasirinkti failą

Duomenys:

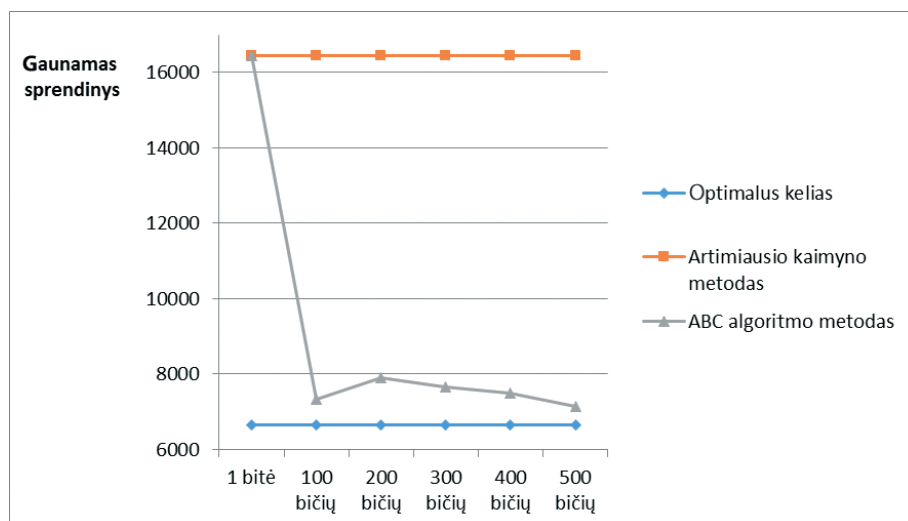
6 pav. Skaičiavimo parametrų įvedimas



7 pav. Programos ABC rezultatų atvaizdavimas



8 pav. ABC algoritmo optimalaus kelio priklausomybė nuo iteracijų skaičiaus



9 pav. ABC algoritmo optimalaus kelio priklausomybė nuo bičių skaičiaus

maršruto mazgo perkėlimas į kitą vietą ir dviejų maršruto mazgų sukeitimas. Į šokio vietą skris tik geriausią sprendinį turinti bitė.

- Atsitiktiniu būdu parenkama, kuo taps darbininkės, baigusios optimizuoti sprendinį – žvalgėmis ar žiūrovėmis. Bitės, kurios tapo žvalgėmis, ankstesnį sprendinį pamiršta ir atsitiktiniu būdu sugeneruoja naują sprendinį. Bitės, kurios tapo žiūrovėmis, ankstesnį sprendinį pamiršta ir nusižiūri geriausią bičių darbininkių sprendinį.

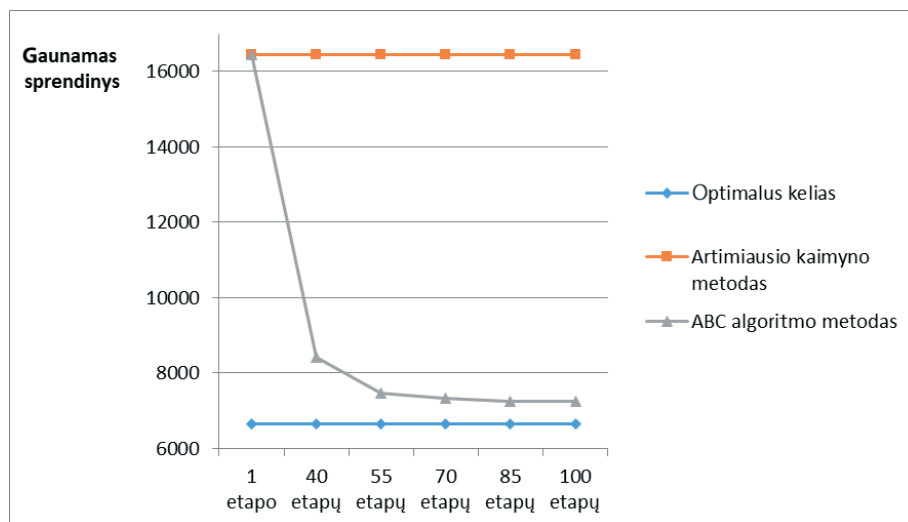
Taigi, darbininkės atsitiktine tvarka atlieka perkėlimo ar sukeitimo funkcijas. Be to, pridėti papildomi parametrai, kuriais galima valdyti perkėlimo ir sukeitimo funkcijų iteracijų skaičių ir pakeisti bičių spiečiaus proporcijas procentais – kuri dalis tampa žvalgėmis ar žiūrovėmis. Bičių spiečiaus algoritmas, pagal kurį sukurta programinė realizacija, pavaizduotas 3 pav.

Sukurtos programos vartotojo sąsaja labai paprasta – tereikia pasirinkti duomenų failą. ABC algoritmo parametrai nustatomi atskirame skirčiuke (6 pav.), kuris atidaromas paspaudus mygtuką „Parametrai“. Nustačius reikiamus parametrus, apskaičiuotas sprendinys atvaizduotas 7 pav.

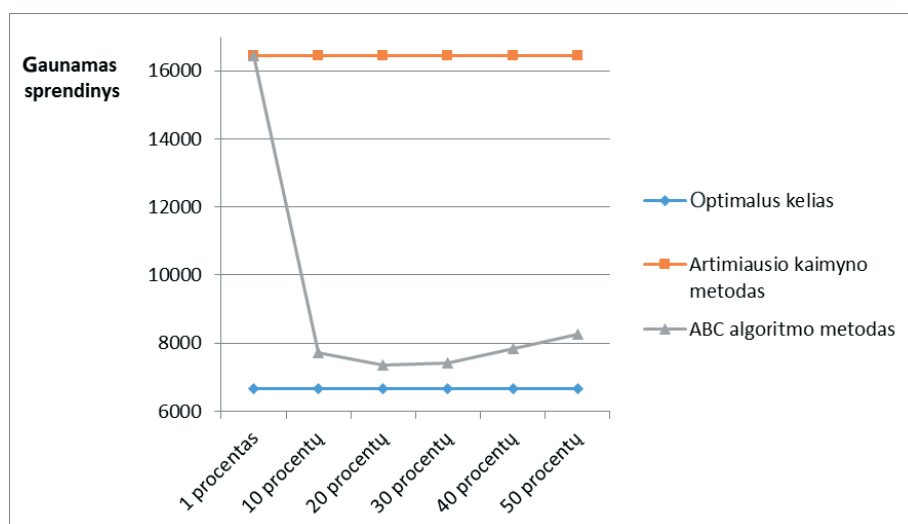
Programos testavimas ir skaičiavimo rezultatai

Sukurta programa buvo testuojama keičiant parametrus, rezultatai analizuojami lyginant su žinomu optimaliu sprendiniu. Kai ABC algoritme kas 50 vienetų keičiamas tik iteracijų skaičius, o kiti parametrai paliekami tokie, kokie numatyti, pastebėta, jog po 350 iteracijų priartėjama prie optimalaus sprendinio. Kiekvienas eksperimentas buvo pakartotas 20 kartų ir nustatytas rezultatų vidurkis. Tai atspindi 8 pav.

Atlikus skaičiavimus vis keičiant bičių skaičių ir apskaičiuojant rezultatų vidurkį, pastebėta, jog numatytas parametras (100 bičių) yra artimas optimaliai reikšmei (9 pav.). Tačiau bičių skaičių padidinus iki 200 rezultatas atitolo nuo optimalaus sprendinio. Tai galėjo atsitikti dėl to, kad ABC algoritmas yra stochastinis – kiekvieną kartą jį naudojant su tais pačiais pradiniais duomenimis gaunamas skirtingas rezultatas, o kiekvienas eksperimentas atliktas pradedant nuo artimiausio kaimyno metodu gauto sprendinio, bet ne bandant pagerinti vieno bandymo rezultatą.



10 pav. ABC algoritmo optimalaus kelio priklausomybė nuo bičių darbininkių etapo



11 pav. ABC algoritmo optimalaus kelio priklausomybė nuo procento bičių, kurios keičia darbo kryptį

Atlikus eksperimentus su bičių darbininkių etapo parametru ir išvedus rezultatų vidurkį paaiškėjo, kad didinant etapų skaičių ir pasiekus reikšmę 55 gaunamas sprendinys mažai beartėja prie optimalaus sprendinio. Eksperimentų rezultatai pateikti 10 pav.

Atlikus ABC algoritmo rezultatų priklausomybės nuo bičių, kurios keičia darbo kryptį, procento eksperimentus ir suskaičiavus rezultatų vidurkį pastebėta, kad ieškant optimalių sprendinių numatytoji parametro reikšmė (20 %) yra efektyviausia. Jei šio parametro reikšmė didinama, gaunami keliai tolsta nuo optimalaus kelio, taip pat kaip ir mažinant šio parametro reikšmę (11 pav.).

Išvados

1. Bičių spiečiaus algoritmai praktikoje dažniausiai taikomi didelės dimensijos kombinatoriniams optimizavimo uždaviniams spręsti; jie efektyvūs, kai reikia taupyti skaičiavimo išteklius.
2. Programai kurti buvo naudojama *JavaScript* kalba. Šios kalbos pranašumas dirbant su masyvais yra

realizuota perkėlimo funkcija. Ji palengvino programos realizaciją ir sumažino kodo eilučių skaičių.

3. Programa veikia be klaidų, gali rasti sprendinius, artimus optimaliems, todėl ją galima naudoti ir didesnės dimensijos uždaviniams spręsti.
4. Išanalizavus ABC algoritmo rezultatus nustatyta, kad prie optimalaus kelio priartėjama maždaug po 150 iteracijų, kai turimi 39 mazgai.
5. Sprendžiant keliaujančio pirklio uždavinius nepakanka apsiriboti klasikinio algoritmo rezultatais, nes ABC metodu gauti rezultatai yra maždaug 2 kartus geresni.
6. Išanalizavus bičių žiūrovių ir bičių žvalgių procento populiacijoje parametrai paaiškėjo, jog artimiausias optimaliam sprendiniui rezultatas gaunamas, kai šis parametras atitinka 15–30 intervalą.

Literatūra

1. Abbass H. A., Teo J., 2001, A True Annealing Approach to the Marriage in Honey-Bees Optimization Algorithm. Prieiga per internetą: <http://pdf.aminer.org/000/905/860/a_true_annealing_approach_to_the_marriage_in_honey_bees.pdf> [žiūrėta 2013-11-04].

2. Abdulsalam M. F., Bakar A. A., 2012, A Cluster-Based Deviation Detection Task Using the Artificial Bee Colony (ABC) Algorithm. Prieiga per internetą: <<http://docsdrive.com/pdfs/medwelljournals/ijscmp/2012/71-78.pdf>> [žiūrėta 2013-11-04].
3. Afshar A., Bozorg Haddad O., Marino M. A., Adams B. J., 2006, Honey-bee mating optimization (HBMO) algorithm for optimal reservoir operation. Prieiga per internetą: <http://robotics.csie.ncku.edu.tw/HCI_Project_2009/徐福瑜2.pdf> [žiūrėta 2013-11-04].
4. Chan F. T. S., Tiwari M. K., 2007, Swarm Intelligence Focus on Ant and Particle Swarm Optimization. *I-tech Education and Publishing*. Vienna, Austria.
5. Felinskas G., 2007, Euristinių metodų tyrimas ir taikymas ribotų išteklių tvarkaraščiams optimizuoti. *Daktaro disertacija*. Vilniaus universitetas. P. 58–72. Prieiga per internetą: <http://www.mii.lt/files/mii_dis_07_felinskas.pdf> [žiūrėta 2013-11-04].
6. Hahsler M., Hornik K., TSP – Infrastructure for the Traveling Salesperson Problem. P. 15-37, 113-145. Prieiga per internetą: <<http://cran.r-project.org/web/packages/TSP/vignettes/TSP.pdf>> [žiūrėta 2013-11-04].
7. Listopadskis N., Kombinatorinio optimizavimo uždaviniai ir jų sprendimo algoritmai. Prieiga per internetą: <<http://www.fmf.lt/ft/studiju-programos/taikomoji-matematika/S-18508/straipsnis/Kombinatorinio-optimizavimo-uzdaviniai-ir-ju-sprendimo-algoritmai?name=S-18508&l=5&p=1>> [žiūrėta 2013-11-04].
8. Misevičius A., Blonskis J., Bukšnaitis V., 2007, Kombinatorinis optimizavimas ir metaeuristiniai metodai: teoriniai aspektai. Prieiga per internetą: <http://www.leidykla.eu/fileadmin/Informacijos_mokslai/42-43/213-219.pdf> [žiūrėta 2013-11-04].
9. Pathak N., Tiwari S. P., 2012, Traveling Salesman Problem Using Bee Colony With SPV. Prieiga per internetą: <<http://www.ijscce.org/attachments/File/v2i3/C0716052312.pdf>> [žiūrėta 2013-11-04].
10. Stanarevic N., Tuba M., Bacanin N., 2011, Modified artificial bee colony algorithm for constrained problems optimization. Prieiga per internetą: <<http://www.naun.org/multimedia/NAUN/m3as/20-418.pdf>> [žiūrėta 2013-10-04].
11. Teodorović D., Bee Colony Optimization (BCO). Prieiga per internetą: <<http://www.sf.bg.ac.rs/downloads/katedre/oi/1.BCO-Book-Chapter.pdf>> [žiūrėta 2013-11-04].
12. Tsai P. W., Pan J. S., Liao B. Y., Chu S. C., 2009, Enchanced artificial bee colony optimization. Prieiga per internetą: <<http://iiirc.hitsz.edu.cn/course/CI/Section5/ENHANCED%20ARTIFICIAL%20BEE%20COLONY%20OPTIMIZATION.pdf>> [žiūrėta 2013-11-04].

ARTIFICIAL BEE COLONY ALGORITHMS AND THEIR APPLICATION TO ROUTE OPTIMIZATION PROBLEMS

Donatas Kavaliauskas, Gražvydas Felinskas

Summary

This paper consists of a short description of swarm systems algorithms, deeper analysis of artificial bee colony (ABC) algorithm and its application for solving traveling salesman problems. A software realization of this algorithm was projected and created. The testing of created software system and analysis of computational results were done. The software is tested by changing the settings of the program and the results are analyzed in comparison with a known optimal solution. An ABC algorithm has been tested for searchable optimal path dependence of the number of iterations, the number of bees, the number of worker bees stages, and bee percentage, which are changing the work direction. Qualitative and quantitative estimates of the parameters of the program are presented in the analysis section of the computational results and conclusions.

Keywords: Optimization, artificial intelligence, meta-heuristics, swarm systems, bee colony.

DIRBTINĖS BIIČIŲ KOLONIJOS ALGORITMAI IR JŲ TAIKYMAS SPRENDŽIANT MARŠRUTŲ OPTIMIZAVIMO UŽDAVINIUS

Donatas Kavaliauskas, Gražvydas Felinskas

Santrauka

Šiame darbe apžvelgiami dalelių spiečių sistemų algoritmai. Plačiau analizuojamas dirbtinių bičių kolonijų (ABC) algoritmas ir jo naudojimas keliaujančio pirklio uždaviniams spręsti. Suprojektuota šio algoritmo programinė realizacija, sukurta programa ištestuota ir pateikta skaičiavimo rezultatų analizė. Programa buvo testuojama keičiant jos parametrus ir rezultatai analizuojami lyginant su žinomu optimaliu sprendiniu. Buvo tiriama pagal ABC algoritmą nustatomo optimalaus kelio priklausomybė nuo iteracijų skaičiaus, bičių skaičiaus, bičių darbininkių etapų skaičiaus, bičių, kurios keičia darbo kryptį, procento. Skaičiavimo rezultatų analizės skyriuje ir išvadose pateikiami kokybiniai ir kiekybiniai programos parametrų įverčiai.

Prasminiai žodžiai: optimizavimas, dirbtinis intelektas, metaeuristikos, spiečių sistemos, bičių kolonijos.

Įteikta 2014-01-15