

# Grafų teorijos uždavinių sprendimo kompiuterinėmis matematikos sistemomis galimybių analizė

Sigita Turskienė, Kristina Kiriliauskaitė

*Šiaulių Universitetas*

Vytauto g. 19, Šiauliai

E. paštas: sigita@fm.su.lt; kristina.kiriliauskaite@gmail.com

**Santrauka.** Straipsnyje analizuojamas grafų teorijos uždavinių sprendimo kompiuterinėmis matematikos sistemomis (KMS) MAPLE ir MATHEMATICA galimybės, atskleidžiant pagrindinių komandų sintaksę, panašumus bei skirtumus, paskirtį. Tai palengvina KMS pasirinkimą praktiniams uždaviniams spręsti ir grafų teorijai mokyti(s). Išmokti kurti sudėtingesnes programas grafų teorijos uždaviniams spręsti KMS nesunku, jeigu turima pradinių programavimo ir algoritmavimo žinių bent viena universalioji programavimo kalba. KMS grafų teorijos uždavinių sprendimo galimybių analizė palengvina taikomųjų programų kūrimą.

**Raktiniai žodžiai:** kompiuterinės matematikos sistemos, grafų teorija.

## Įvadas

Sprendžiant praktinius vadybos uždavinius, dažnai susiduriama su užduotimis, reikalaujančiomis nubraižyti tam tikrą situaciją modeliuojantį grafą, rasti optimalų agento maršrutą, Oilerio ciklus ir pan. Kompiuterinių technologijų diegimas grafų teorijos mokyme(si) leidžia nesunkiai sukurti įvairius grafus, išsamiau tyrinėti jų savybes, pastebėti procesų dėsningumus, besimokančiųjų žinias sieti su kūrybine veikla. Tokia kompiuterine technologija gali būti universalios kompiuterinės matematikos sistemos (KMS), turinčios šiuolaikišką ir patogią vartotojo grafinę sąsają [6], į problemą orientuotą programavimo kalbą [5], grupę komandų grafų teorijos uždaviniams spręsti [1]. Kaip rodo [3] tyrimai, KMS diegimas matematikos mokyme reikalauja specialių pastangų tiek iš mokytojų, tiek iš besimokančiųjų pusės. Išsamiai KMS programavimo kalbų galimybės apžvelgiamos [5]. Programų vartotojo grafinės sąsajos kūrimo galimybių lyginamoji analizė atlikta [6]. Grafų teorijos uždaviniams spręsti galima naudoti ir specializuotas mokomąsias programas, tačiau jų pasiūla labai maža. MAPLE sistema sukurta aplikacija „Grafų teorijos uždavinių sprendimas“ aprašyta D. Gaubienės darbe [1]. Tyrimus būtina tęsti. Pirmą prielaidą – pastovus KMS naujų grafų teorijos uždavinių sprendimo komandų diegimas, jų galimybių tobulėjimas. Antra prielaidą – nėra išsamų tyrimų, kurie įvertintų grafų teorijos uždavinių sprendimo KMS galimybes. Į nagrinėjamų sistemų sąrašą įtrauktos šios KMS: MATHEMATICA 7, MAPLE 12. Sistema MAPLE plačiai naudojama universitetuose, kolegijose, mokyklose; MATHEMATICA – labiau universitetuose ir mokslinių tyrimų centruose. Abi pasirinktos sistemos turi į problemą orientuotas vidines programavimo kalbas, t.y.

skirtas matematikos uždaviniams spręsti. Jų vartojimo sritys yra plačios, nemažai dėmesio skiriama rezultatams vizualizuoti. Pasirinktas sistemų lyginimo matas yra kiek primityvus, nes jų paskirtys yra panašios. Bet uždavinių sprendimo galimybės KMS yra skirtingos (skiriasi komandų kiekiu, vartojamais uždavinių sprendimo algoritmais ir jų pateikimu). Grafų teorijos uždavinius MAPLE ir MATHEMATICA sistemomis galima spręsti dvejopai: 1. Rašyti aplikacijas [1] ar programas [5] grafams kurti ir veiksmams su jais atlikti. Tam pakanka žinoti grafų kūrimo algoritmus [2], sistemos programavimo kalbos elementus [5], vartotojo grafinės sąsajos kūrimo galimybes [6], kompiuterinės grafikos galimybes [4]. 2. Komandų eilutėje vykdyti KMS siūlomas grafų kūrimo, veiksmų su jais ar jų elementais komandas. Šio darbo tikslai: pristatyti MAPLE ir MATHEMATICA sistemų grafų teorijos uždavinių sprendimo galimybių analizę, išsiaiškinant tam skirtų komandų sintaksę, skirtumus bei panašumus, grafų kūrimo bei vaizdavimo galimybes, grafų savybių ir veiksmų su jais komandas; išnagrinėti KMS galimybes uždaviniams su grafais spręsti. Tai palengvins KMS taikymą praktiniam darbui ir dalyko mokymui(si).

## 1 Grafų teorijos uždavinių sprendimo sistema MAPLE galimybės

Greitus ir vizualesius grafų teorijos uždavinių sprendinius, esant įvairioms pradinėms sąlygoms, užtikrina sistemos MAPLE šios galimybės: skaitiniai ir simboliniai skaičiavimai, nesudėtingi programavimo kalbos elementai, patogios grafiškos priemonės, animacijos galimybės, specializuotos komandos. Uždaviniams su grafais spręsti sistema MAPLE siūlo šias priemones: 1. Paketo networks (paketas sukurtas MAPLE V sistemai) komandos. 2. Paketo GraphTheory ir jo popakečio SpecialGraphs komandos. Pagal komandų paskirtį išskiriamos šios paketų galimybės: Grafų kūrimas: 1. Paketo networks: pilnasis grafas iš  $n$  viršūnių (komanda complete), grafas iš nurodyto skaičiaus viršūnių ir briaunų (komanda graph), naujas grafas be viršūnių ir briaunų (komanda new), atsitiktinis grafas iš  $n$  viršūnių ir  $m$  briaunų (komanda random), tuščiasis grafas (komanda void), grafo kopija (komanda duplicate) ir kt. 2. Paketo GraphTheory: paprastasis ne jungusis ir jungusis  $n$  viršūnių grafas (komanda Graph), pilnasis  $n$  viršūnių grafas (komanda CompleteGraph), orientuotas grafas (komanda Digraph), grafas, kurio visos viršūnės sudaro ciklą (komanda CycleGraph) ir kt. 3. Popakečio SpecialGraphs: pilnasis dvejetainis medis (komanda Complete BinaryTree), tinklelio grafas (komanda GridGraph), žvaigždės grafas (komanda StarGraph), Peterseno grafas (komanda PetersenGraph) ir kt. 4. Grafo įrašymas į failą/skaitymas iš failo (komandos ImportGraph, ExportGraph). Grafo vaizdavimas: 1. Paketo networks komanda draw vaizduoja duotąjį grafą. 2. Paketo GraphTheory komanda DrawGraph vaizduoja duotąjį grafą MAPLE grafiku, komanda DrawNetwork vaizduoja tinklinį grafą, komanda HighlightedTrail vaizduoja spalvotą kelią grafe. Veiksmai su grafais ir jo elementais: 1. Paketo GraphTheory: grafo viršūnės pridėjimas (komanda AddVertex), grafo viršūnės ištrynimasis (komanda DeleteVertex), grafo viršūnių koordinacių nustatymas (komanda GetVertexPositions), koordinacių priskyrimas grafo viršūnėms (komanda SetVertexPositions), briaunos pridėjimas (komanda AddEdge), briaunos ištrynimasis (komanda DeleteEdge), grafo kopija (komanda CopyGraph), grafo ciklų radimas (komanda CycleBasis), grafo vertimas orientuotu grafu (MakeDirected), grafo vertimas svoriniu (MakeWeighted), grafo briaunos svoris (GetEdgeWeight), gra-

fo viršūnių sujungimas (komanda `Contract`), grafų sąjunga (komanda `GraphUnion`), dviejų grafų jungtis (komanda `GraphJoin`), grafo konvertavimas pasirinktu formatu (komanda `ConvertGraph`). 2. Paketo `networks`: viršūnės pridėjimas (komanda `addvertex`), briaunos pridėjimas (komanda `addedge`), briaunos ištrynimasis (komanda `delete`), dviejų grafų sujungimas (komanda `gunion`), viršūnių grupė (komanda `shrink`) ir kt. Informacija apie grafo laipsnius: 1. Paketo `GraphTheory`: kiek grafo viršūnių jungiasi su nurodyta viršūne (komanda `Degree`), kiek yra briaunų, kurios nukreiptos į orientuotą grafo viršūnę (komanda `InDegree`), kiek yra briaunų, išeinančių iš nurodytos viršūnės (komanda `OutDegree`), visų grafo viršūnių laipsniai (komanda `DegreeSequence`), grafo didžiausio laipsnio nustatymas (komanda `MaximumDegree`), grafo mažiausio laipsnio nustatymas (komanda `MinimumDegree`), skaičius briaunų, kurias pašalinus grafas nebus jungsis (komanda `EdgeConnectivity`) ir kt. 2. Paketo `networks`: kiek yra briaunų, kurios nukreiptos į grafo viršūnę (komanda `indegree`), kiek yra briaunų, išeinančių iš nurodytos viršūnės (komanda `OutDegree`), grafo viršūnių laipsniai (komanda `degreeseq`), grafo didžiausias laipsnis (komanda `maxdegree`), grafo komponentų skaičius (komanda `bicomponents`) ir kt. Informacija apie grafą: 1. Paketo `GraphTheory`: grafo viršūnių sąrašas (komanda `Vertices`), grafo briaunų sąrašas (komanda `Edges`), jungiųjų viršūnių nustatymas (komanda `ArticulationPoints`), grafo ciklo nustatymas (funkcija `IsAcyclic`), mažiausias skaičius briaunų ar viršūnių, kurias pašalinus grafas tampa nejungiuoju (komanda `EdgeConnectivity`, `VertexConnectivity`), grafo briaunos įeinančios ar išeinančios iš nurodytos viršūnės (komanda `IncidentEdges`), grafo jungumas (komanda `IsConnected`), ar grafas yra orientuotas (komanda `IsDirected`), ar grafas yra svorinis (komanda `IsWeighted`), ar grafas turi Oilerio ciklą (komanda `IsEulerian`), ar grafas turi Hamiltono ciklą (komanda `IsHamiltonian`), gretimų grafo viršūnių nurodymas (komanda `Neighborhood`), grafo briaunų/viršūnių skaičius (komandos `NumberOfEdges/NumberOfVertices`), grafo matrica (komanda `IncidenceMatrix`), grafo trumpiausio ciklo ilgis (komanda `Girth`), grafo diametras (komanda `Diameter`) ir kt. 2. Paketo `networks`: grafo viršūnių sąrašas (komanda `vertices`), grafo viršūnės ištrynimasis (komanda `delete`), grafo matrica (komanda `incidence`), grafo viršūnių vardai (komanda `getlabel`), grafo viršūnių laipsniai (komanda `vdegree`), grafo ciklai (komanda `fundcyc`), trumpiausio ciklo ilgis (komanda `girth`), grafo diametras (komanda `diameter`) ir kt. Taikomųjų grafų teorijos uždavinių sprendimo komandos: 1. Paketo `GraphTheory`: pigiausio svorinio kelio radimas (komandos `BellmanFordAlgorithm` ir `DijkstraAlgorithm`), trumpiausio kelio radimas (funkcija `Distance`), grafo trumpiausio kelio nuo vienos viršūnės iki kitos radimas (komanda `ShortestPath`), keliaujančio pirklio uždavinys (komanda `TravelingSalesman`) ir kt. 2. Paketo `networks`: kelio nuo viršūnės  $a$  iki viršūnės  $b$  radimas (komanda `path`), trumpiausias kelias (komanda `shortpathstree`), visų porų trumpiausi keliai (komanda `allpairs`), ilgiausias grafo kelias nuo vienos viršūnės iki kitos (komanda `dinic`) ir kt. Palyginus sistemos MAPLE siūlomas priemonės, išryškėja: 1. Abu paketai turi komandas grafams kurti, vizualizuoti, kopijuoti, saugoti; veiksams su grafais ir jų elementais; informacijai apie grafą gauti; taikomiesiems uždaviniams spręsti. 2. Paketo `networks` daugumos pagrindinių komandų pavadinimai yra anglų kalbos žodžiai (juos turi ir paketo `GraphTheory` komandos), kurie rašomi iš mažosios raidės (MAPLE kalboje didžiosios ir mažosios raidės traktuojamos skirtingai), o paketo `GraphTheory` – iš didžiosios raidės. 3. Paketai turi skirtingą komandų skaičių: `networks` (76 komandos), `GraphTheory` (151 komanda). 4. Abiejų paketų komandoms galima pateikti įvairius

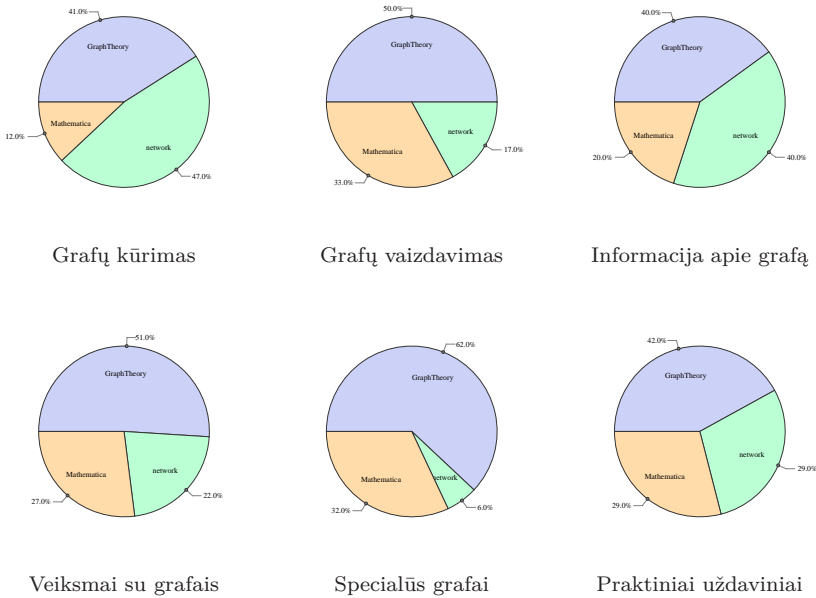
pradinius duomenis: viršūnių skaičius, briaunų skaičius, viršūnių jungtis, kraštinių nustatymus ir kt.

## **2 Grafų teorijos uždavinių sprendimo galimybės sistema MATHEMATICA**

Sistemos MATHEMATICA pagrindinės komandos yra sistemos branduolyje ir pakete Needs [„GraphUtilities“]: 1. Grafų kūrimas ir vaizdavimas: dvimatis grafas (komanda GraphPlot), erdvinis grafas (komanda GraphPlot3D), specializuoti grafai (komanda GraphData), medžio tipo grafas (komanda TreePlot). 2. Grafo laipsnio nustatymas: grafo viršūnių laipsnis (komanda Degree), į viršūnę įeinančių briaunų skaičius (komanda InDegree), išeinančių briaunų skaičius (komanda OutDegree). 3. Veiksmai su grafais ir jo elementais: grafo viršūnių žymėjimas (komanda GraphCoordinates), koordinatų priskyrimas grafo viršūnėms (parametras VertexCoordinateRules), grafo viršūnių sąrašas (komanda Bicomponents), viršūnės pridėjimas (komanda AddVertex), grafo viršūnių sąrašas (komanda Vertices), viršūnės ištrynimasis (komanda DeleteVertex), briaunos pridėjimas (komanda AddEdge), briaunos ištrynimasis (komanda DeleteEdge), grafo briaunų sąrašas (komanda EdgeList), ciklo radimas grafe (komanda FindCycle), grafo vertimas orientuotu (komanda MakeDirected), grafo briaunų svoriai (komanda GetEdgeWeights). 4. Taikomųjų uždavinių sprendimas: trumpiausio kelio briaunų, viršūnių skaičius; trumpiausias kelias nuo vienos viršūnės iki kitos (komandos GraphDistance, GraphPath, Shortestpath), grafo Hamiltono ciklas (komandos FindHamiltonianCycle, HamiltonianCycles, Hamiltonianpath) ir kt. 5. Informacija apie grafą: grafo nurodytos viršūnės kaimynės (komanda NeighborhoodVertices), grafo matrica (komanda AdjanceMatrix), grafo viršūnių koordinatės (komanda GraphCoordinates), grafo viršūnių laipsnis (komanda Degrees), didžiausias nepriklausomų briaunų skaičius (komanda MaximalIndependentEd) ir kt. Palyginus sistemos MATHEMATICA siūlomas priemonės, nustatyta: 1. Sistema turi komandas grafams kurti, vizualizuoti, kopijuoti; veiksams su grafais ir jų elementais; informacijai apie grafą gauti; taikomiesiems uždaviniams spręsti. 2. Pagrindinių komandų pavadinimai yra anglų kalbos žodžiai, kurie rašomi iš didžiosios raidės (MATHEMATICA kalboje didžiosios ir mažosios raidės traktuojamos skirtingai). 3. Palyginus su MAPLE sistema, turi mažiau komandų grafų teorijos uždaviniams spręsti.

## **3 Grafų teorijos elementų realizavimo galimybių sistemomis MAPLE ir MATHEMATICA palyginimas**

Išsamus KMS grafų teorijos elementų realizavimo galimybių nagrinėjimas užima daug laiko. Be to, kai kurios sistemų savybės išryškėja tik detalai su jomis susipažinus. KMS grafų teorijos pagrindinių elementų galimybių įsisavinimui palengvinti pateiksime nagrinėjamų KMS (MAPLE, MATHEMATICA) dar neaptartus panašumus ir skirtumus. KMS grafų teorijos uždavinių sprendimo galimybių skirtumai: 1. Skiriasi komandų išsidėstymu sistemose: MAPLE sistemoje komandos yra sudėtos į paketus, MATHEMATICA sistemoje dauguma komandų yra branduolyje. 2. Vartoja skirtingas vidines procedūrinio programavimo kalbas, kurių sintaksė artima PASCAL ir C kalboms [4]. Kaip rodo darbo praktika, tokia programinių priemonių simbiozė duo-



1 pav. KMS pagrindinės komandos grafų teorijos uždaviniams spręsti.

da gerus praktinio darbo rezultatus. 3. Turi skirtingą kiekį komandų pagrindiniams grafų teorijos uždaviniams spręsti (1 pav.). Pagal vartojimo paskirtį KMS pagrindinės komandos suskirstytos į 6 grupes: grafų kūrimo, grafo vaizdavimo, informacijos apie grafą, veiksmai su grafais, specializuoti grafai, taikomųjų uždavinių sprendimo. 4. Specializuoti grafai kuriami skirtingai: MATHEMATICA sistemoje yra viena komanda GraphData, kurioje grafų tipas nurodomas parametru, o MAPLE sistema pakete GraphTheory yra popaketis SpecialGraphs, kuriame yra komandos specializuotiems grafams kurti.

KMS grafų teorijos uždavinių sprendimo galimybių panašumai: 1. Turi panašius ar sutampančius komandų grafų teorijos uždaviniams spręsti pavadinimus: AddVertex, AddEdge, Delete Edge, MakeDirected, GetEdgeWeight ir kt. Todėl išsamiai galima nagrinėti vienos sistemos galimybes, o kitas įsisavinti analogiškai. 2. Pagrindinės komandos yra anglų kalbos žodžiai ar jų sutrumpinimai, todėl nesunku suprasti komandos paskirtį iš pavadinimo. 3. Sukuria to paties arba labai panašaus grafinio vaizdo grafus. 4. Visos komandos yra griežtai tipizuotos ir vartotojo programose naudojamos pagal paskirtį. 5. KMS leidžia kurti vartotojo grafinę sąsają jų pagrindiniuose languose. Taip palengvinamas interaktyvių grafų prezentacijų ar atvaizdavimų generavimas.

## Išvados

1. Sistemomis MAPLE ir MATHEMATICA galima kurti grafų teorijos kursuose sutinkamus pagrindinius grafus, atlikti jų analizę, spręsti taikomojus uždavinius.

2. Palyginus suformuluotų grafų teorijos uždavinių sprendimo sistemomis MAPLE ir MATHEMATICA pagrindinius skirtumus ir panašumus, nustatyta, kad sistema MATHEMATICA turi mažiau standartinių funkcijų grafų teorijos uždaviniams spręsti.

## Literatūra

- [1] D. Gaubienė. *Grafų teorijos uždavinių sprendimo programa*. Šiauliai, 2008.
- [2] Y.F. Hu. Efficient, high-quality force-directed graph drawing. *The Mathematica Journal*, **10**(1):37–71, 2006.
- [3] S. Turskienė. Computer technology and teaching mathematics in secondary schools. *Informatics in Education*, **1**:149–156, 2002.
- [4] S. Turskienė. *Uždavinių sprendimas kompiuterinėmis matematikos sistemomis*. VšĮ Šiaulių universiteto leidykla, Šiauliai, 2003.
- [5] S. Turskienė. *Kompiuterinių matematikos sistemų programavimo galimybės*. VšĮ Šiaulių universiteto leidykla, Šiauliai, 2004.
- [6] S. Turskienė. Kompiuterinių matematikos sistemų programų grafinės vartotojo sąsajos kūrimo galimybių lyginamoji analizė. *Informacijos mokslai*, **34**:36–40, 2005.

## SUMMARY

### **Opportunities Analysis the solution of the problems of the graph theory with computer mathematics systems**

*S. Turskienė, K. Kiriliauskaitė*

In this paper we analyze solutions of the problems of the graph theory with computer mathematics systems MAPLE and MATHEMATICA with purpose to reveal the opportunity of the syntax of the main commands, their similarities and differences. It facilitates the choice of the computer mathematics systems for solving the practical problems of the graph theory and to study this theory. To learn to create sophisticated programs of the problems of the graph theory with computer mathematics systems is easy, if the initial knowledge of programming is at least one language of the programming. The analysis of the opportunities of computer mathematics systems by solving problems of the graph theory facilitate creation of the applied programs.

*Keywords:* computer mathematics systems, Graph theory.