



**Vilniaus
universitetas**

Informatikos ir informatinio mąstymo mokomoji veikla

Srautų algoritmai

Mokyklos pedagogika



Kuriame
Lietuvos ateitį
2014–2020 metų
Europos Sąjungos
fondų investicijų
veiksmų programa



**Vilniaus
universitetas**

Informatikos ir informatinio mąstymo mokomosios veiklos sukurtos įgyvendinant projektą „Aukštųjų mokyklų tinklo optimizavimas ir studijų kokybės gerinimas Šiaulių universitetą prijungiant prie Vilniaus universiteto“ (Nr. 09.3.1-ESFA-V-738-03-0001), finansuojamą iš Europos socialinio fondo lėšų pagal 2014–2020 metų Europos Sąjungos fondų investicijų veiksmų programos 9 prioriteto „Visuomenės švietimas ir žmogiškųjų išteklių potencialo didinimas“ įgyvendinimo priemonę Nr. 09.3.1-ESFA-V-738 „Aukštųjų mokyklų tinklo tobulinimas“.

Metodinė medžiaga „Srautų algoritmai“ skirta Mokyklos pedagogikos studijų programos moduliui „Informatikos didaktika“. Tikslinė grupė – būsimi informatikos pagrindinio ar vidurinio ugdymo mokytojai. Medžiaga siejasi su Bendrosiomis informatikos ir matematikos programomis, algoritmų ir programavimo pasiekimų sritimi. Atlikdami veikloje numatytas užduotis mokytojai išsiaiškins, kas yra srautas grafe, kokie yra pagrindiniai srautų algoritmai, kaip ir kur juos taikyti. Aptariamas Fordo ir Falkersono algoritmas maksimalaus srauto uždaviniui.

Veiklą parengė: dr. Vaidas Giedrimas

Redagavo: Viktoras Dagys

Vilnius, 2022

Tikslas

Susipažinti su srautais grafuose bei algoritmais jiems apdoroti.

Ryšys su bendrosiomis programomis

Informatika: Duomenų vaizdavimas

Informacinės technologijos: Piešimas kompiuteriu; Duomenų apdorojimas ir pateikimas skaičiuokle

Integracija su matematika: Matematinis mąstymas; Problemų sprendimas

Srautų algoritmai

Dauguma šiuolaikinių sistemų yra išskirstytosios, sudarytos iš daugelio tarpusavyje sąveikaujančių elementų. Organizuojant tokių sistemų darbą yra labai svarbu nustatyti galimus jų apribojimus ir kiek tas sistemas dar galima optimizuoti. Štai keletas tokio tipo uždavinių:

- **Darbų paskirstymas komandoje.** Turime grupę darbuotojų, kurių kiekvienas gali atlikti tam tikrus darbus, tačiau jų įkainiai yra skirtingi. Reikia taip paskirstyti darbus, kad išlaidos darbo užmokesčiui būtų mažiausios.
- **Darbovietės atranka.** Įmonė yra paskelbusi apie keletą skirtingų laisvų darbo vietų. Visi norintys gali teikti paraiškas. Be to, galima aplikuoti į dvi ar daugiau darbo vietų vienu metu (panašiai, kaip LAMABPO stojant į aukštąsias mokyklas). Mūsų uždavinys yra taip pasirinkti kandidatus, kad būtų įdarbintas didžiausias jų skaičius.
- **Logistika.** Reikia rasti mažiausiai sąnaudų reikalaujančią (efektyviausią) tiekimo grandinę, kai yra keletas gamyklų ir keletas parduotuvių. Dažnai šiuo atveju yra ir keletas tarpinių sandėlių laikinam prekių saugojimui.
- **Didžiausio srauto paieška.** Nagrinėjama, kiek duotas tinklas gali daugiausiai perduoti prekių iš gamyklų į parduotuves. Šis uždavinys labai populiarus ir kitose srityse, pvz., telekomunikacijų bendrovei gali būti svarbu skaičiuoti, koks yra maksimalus jos tinklo duomenų pralaidumas mieste (ar šalyje), ir šios analizės pagrindu planuoti tinklo plėtrą.

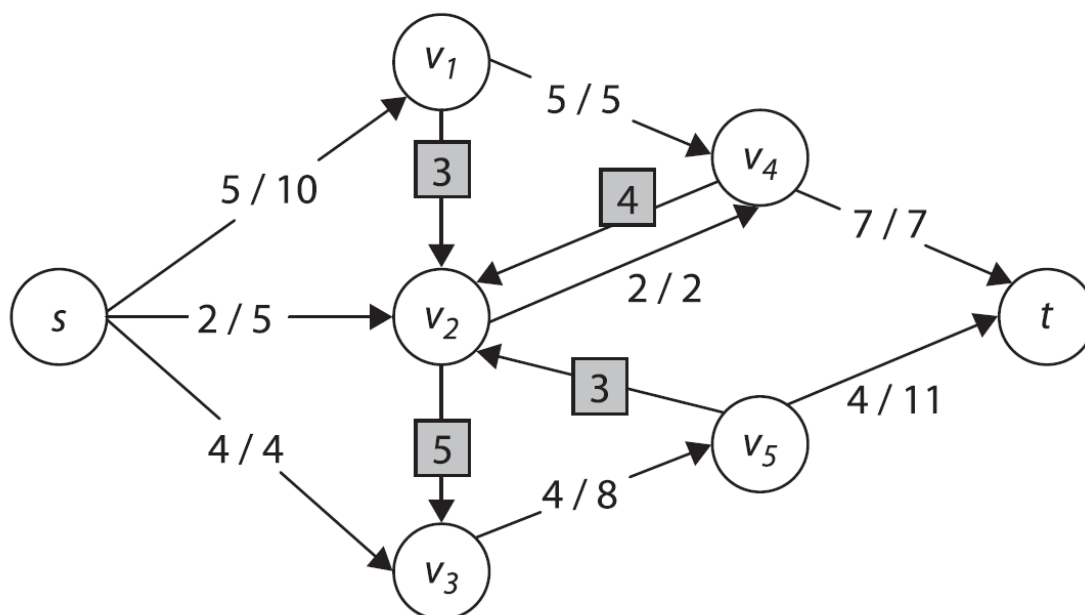
Tokie uždaviniai sprendžiami naudojant srautų algoritmus.

Pagrindinės sąvokos

Inžinerijoje dažnai naudojami lankais susietų viršūnių (elementų) **tinklai**. Tokiuose tinkluose lankai turi kryptį ir svorį (maksimalų pralaidumą). Kaip jau minėta, tinklais yra modeliuojamos telekomunikacijų, vandentiekio, kanalizacijos, elektros tiekimo ir kitos sistemos. Matematikoje tinklu vadinamas jungusis kryptinis grafas $G = (V, E)$, kuriame išskirtos bent dvi (o kartais ir daugiau) ypatingos viršūnės: **šaltinis** ir **imtuvas**.

Daroma prielaida, kad šaltinis gali generuoti be galo daug produkcijos, o imtuvas – sunaudoti viską, kas jam yra tiekiamas, t. y., nėra numatyta prastovų (kurios realiame gyvenime visgi atsitinka).

Tinkluose kiekvienas lankas paprastai nusakomas ne vienu, o dviem svoriais: srautu ir talpa. **Srautu** tarp viršūnių u ir v vadinama funkcija $f(u, v)$, kurios reikšmė nusako, kiek vienetų kažko jau yra tiekama iš vienos viršūnės į kitą. **Talpa** vadinama funkcija $c(u, v)$, kurios reikšmė nusako, kiek maksimaliai vienetų tuo lanku galima perduoti. Pavyzdžiui, tarp viršūnių s ir v_1 šiuo metu yra perduodami 5 vienetai, o daugiausiai galima perduoti – 10 (1 pav.), taiga šį lanką dar galima geriau panaudoti. Kai kurie kiti lankai jau yra maksimaliai naudojami (v_1-v_4 , v_4-t), nes jų srautas f ir talpa c sutampa.



1 pav. Tinklas

Tokiame tinkle yra laikomasi šių apribojimų:

- Srautas $f(u, v)$ negali viršyti talpos $c(u, v)$. Jeigu viršūnės tarpusavyje nesujungtos, tai laikoma, kad jų talpa yra lygi nuliui;
- Jokia kita viršūnė, išskyrus s , nieko negamina. Jokia kita viršūnė išskyrus t , nieko nevartoja, o tik perduoda. Taigi vadinamųjų nutekėjimų ar varžos šiame sraute nėra.
- Jei tarp viršūnių u ir v yra srautas $f(u, v)$, tai laikoma, kad kryptiniame grafe tarp viršūnių v ir u yra $-f(u, v)$, t. y. priešingas srautas. Ši taisyklė galioja net tada, kai atskiro lanko tarp v ir u net nėra.

Maksimalaus srauto uždavinys

Reikia rasti didžiausią galimą srautą tarp šaltinio s ir imtuvo t , įvertinant tarpinių lankų talpas.

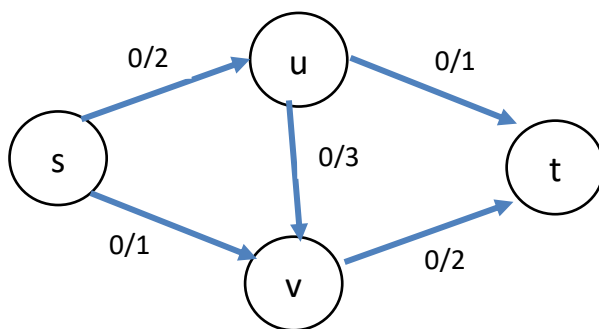
Dėmesio! Skirtingai nuo klasikinių grafų teorijos uždavinių čia ieškoma ne vieno kelio, o jų junginio srautui užtikrinti, t. y., iš viršūnės (ar į ją) srautas gali ištekti (ar įtekėti) keliais lankais vienu metu. Taip pat svarbu, kad tekėjimas vyksta tik viena kryptimi ir jei jau lankas yra užimtas srautui viena kryptimi užtikrinti, jis jau negali būti naudojamas priešingos krypties srautui tarp viršūnių.

Fordo ir Falkersono algoritmas

Elementarus godusis uždavinio sprendimo algoritmas būtų toks:

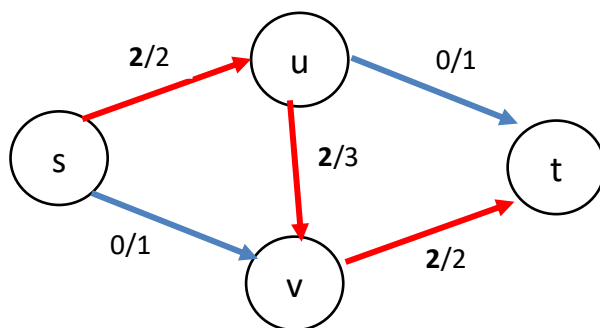
1. Randamas ir pasirenkamas didžiausios talpos lankas iš viršūnės S .
2. Randamas (jei įmanoma) srautas nuo tuo lanku pasiekiamos viršūnės iki imtuvo T pasirenkant būtinus tarpinius lankus.
3. Ieškoma, ar dar galima srautą padidinti, panaudojus kitus lankus.

Kaip ir įprasta godiesiems algoritmams, šis nebūtinai ras optimalų sprendimą. Panagrinėkime pavyzdį (2 pav.).



2 pav. Paprastas tinklas, kuriame srauto dar nėra

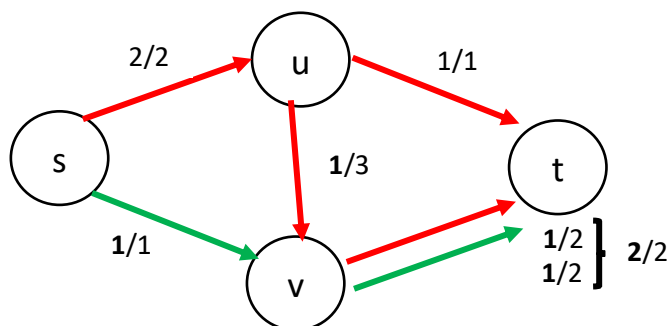
Didžiausias galimas srautas iš viršūnės s yra lanku $s-v$, jį ir pasirinkime. Kadangi maksimali jo talpa yra 2, tai tokio dydžio srautą bandykime nukreipti kitais lankais iki pat t . Didžiausio pralaidumo lankas iš u yra $u-v$, be to, jo talpa yra netgi didesnė už 2, todėl jį pasirinkime. Ir iš v turime vienintelį lanką $v-t$, kurio talpos užtenka perduoti 2 srauto vienetų. Uždavinys atrodo išspręstas (3 pav.). Tačiau gali kilti klausimas, ar čia jau didžiausias galimas srautas (ko, beje, reikalavo ir uždavinio sąlyga)?



3 pav. Dalinis maksimalaus srauto sprendimas

Ar yra didesnio tiekimo iš s galimybė? Taip, šaltinis s generuoja ir lanku $s-v$ gali būti perduodamas dar vienas srauto vienetas, tačiau viršūnėje v toks srautas ir užstrigtų, nes $v-t$ talpa jau išnaudota, o tarp viršūnių v ir u lanko nebėra (nes teka srautas $u-v$). Vadinasi iš pradžių buvo priimtas netinkamas sprendimas.

Grafas nedidelis, todėl jame empiriniu būdu galime rasti didesnį srautą tarp s ir t , perduodantį 3 vienetų (4 pav.). Atkreipkite dėmesį, kad čia atsisakyta godžiojo algoritmo ir srautas iš u padalintas keliems lankams, užuot užėmus visą lanką $u-v$. Kartu iš dalies atsilaisvino ir lankas $v-t$, kuris panaudotas toliau perduoti srautui, ateinančiam $s-v$ lanku.



4 pav. Optimalus maksimalaus srauto sprendimas

Esant didesniam lankų ir viršūnių skaičiui empirinis būdas yra nepatogus, todėl naudojami kiti algoritmai. Vieną iš dažniausiai naudojamų algoritmų maksimalaus srauto uždaviniui spręsti 1956 m. pasiūlė amerikiečių mokslininkai Lesteris R. Fordas ir Dilbertas R. Fulkersonas.

Šis algoritmas yra toks:

1. Pirmiausia imamas bet koks pradinis kelias nuo S iki T , apskaičiuojamas jo maksimalus srautas. Maksimalus srautas šiuo atveju yra lygus silpniausios to kelio grandies (lanko) talpai.

2. Ieškoma papildomo kelio, kuriuo dar galima perduoti srautą iš šaltinio į imtuvą. Jei toks kelias randamas, juo papildomas esamas (dar vadinamas **auginančiu srautu**) kelias, o maksimalaus srauto reikšmė atitinkamai didinama.
3. Kartojamas 3-asis punktas, kol nebebus galima papildyti esamo srauto jokių kitų kelių. Gauta maksimalaus srauto reikšmė ir bus pati didžiausia galima.

Algoritmas primena godųjį, tačiau išsaugoma informacija apie tai, kurie lankai jau buvo pasirinkti, kad probleminėse situacijose (kaip 3 pav.) būtų galima grįžti atgal ir priimti kitus sprendimus. Tam naudojama dar viena – **liekamojo grafo** (angl. *residual graph*) struktūra. Kas yra tas liekamasis grafas? Tai tas pats grafas, tik papildytas dar keletu lankų, vadinamų **netiesioginiais**. Šalia tų lankų, kuriais jau perduodame srautą, braižome po dar vieną, priešingos krypties lanką, kurio svoris yra jau perduodamo srauto dydis.



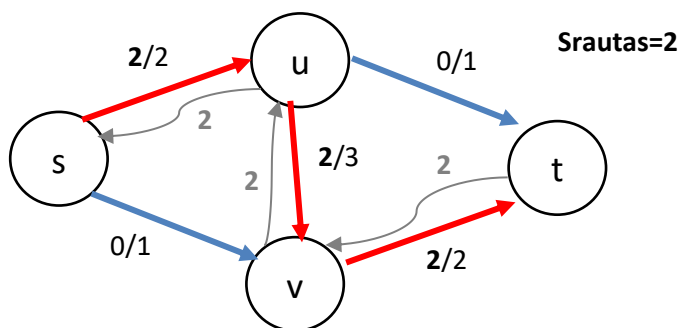
5 pav. Dilbertas R. Fulkersonas

https://en.wikipedia.org/wiki/D._R._Fulkerson#/media/File:Delbert_Ray_Fulkerson.png

Ieškant naujų kelių netiesioginiai lankai taip pat gali būti naudojami. Tuo atveju yra atitinkamu dydžiu (kokio dydžio srautą reikia perduoti) mažinamas jų svoris. Kaip jau minėta, netiesioginių lankų naudojimas padeda atšaukti arba pakoreguoti anksčiau priimtus sprendimus, tačiau nėra būtinybės saugoti visą sprendimų istoriją chronologine tvarka, o tai labai sumažina algoritmo sudėtingumą.

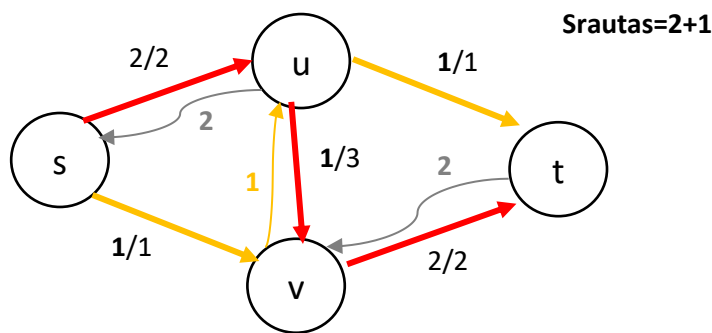
Dauguma maksimalaus srauto nustatymo algoritmų literatūroje aiškinami naudojant matematinę sąvoką ir matematinius simbolius. Toks pateikimas gali būti sudėtingas moksleiviams, todėl rekomenduojama tiek pačius uždavinius, tiek jų sprendimo algoritmus pateikti per gyvenimiškus pavyzdžius arba grafiškai. Šaltiniuose naudojami skirtingi liekamojo grafo ir netiesioginių lankų vaizdavimo būdai. Panagrinėkime keletą jų.

1. **Klasikinis būdas.** Ant lanko žymima esamo srauto ir talpos pora. Esamo srauto dydis taip pat dubliuojamas ant priešingo lanko. Imkime 2 pav. pavaizduotą tinklą. Po pirmojo žingsnio, sukūrus srautą $s-u-v-t$, liekamasis grafas atrodys taip, kaip parodyta 6 paveiksle.



6 pav. Tinklas po pirmojo žingsnio

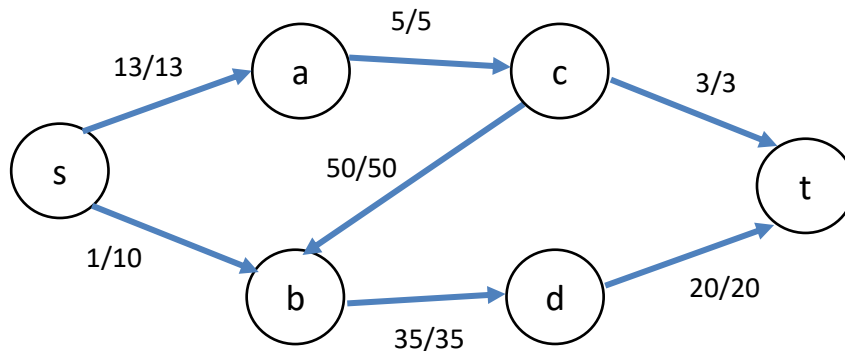
Kitame žingsnyje $s-u$ lankas nebegali būti naudojamas, nes jau pasiekta maksimali talpa, todėl imamas $s-v$ lankas. Bėda ta, kad $v-t$ irgi jau užimtas. Tačiau čia pasinaudojame likutinio grafo lanku $v-u$ ir sukuriame srautą $s-v-u-t$ (7 pav.) atitinkamai keisdami tiek $v-u$, tiek tiesioginių lankų svorius.



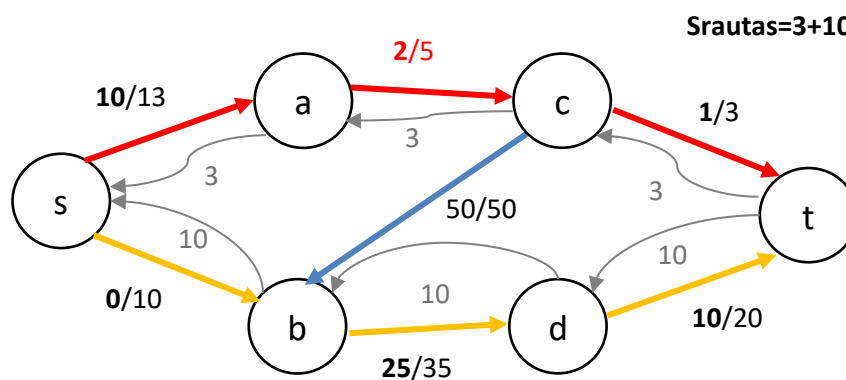
7 pav. Tinklas po antrojo žingsnio

Gali kilti klausimas – kaip įmanoma vienu metu tarp u ir v viršūnių užtikrinti priešingus srautus? Trumpas atsakymas yra – neįmanoma. Tačiau čia reikia akcentuoti, kad liekamojo grafo lankai yra tik pagalbiniai ir į juos nebekreipiame dėmesio po to, kai srautas jau suplanuotas. O jei dabar pažvelgsime į tikruosius lankus, matysime, kad srautas ateinantis iš s , viršūnėje u išskirstomas: 1 vienetas jo nukreipiamas tiesiai į t , o kitas teka į v , kur susijungia su iš s atitekančiu 1 vieneto srautu ir taip suformuoja 2 vienetų srautą iki t . Pasiekiame 4 pav. gautą sprendinį, tik jau ne empiriniu būdu o taikydami algoritmą.

2. **Likučio žymėjimas šalia talpos.** Skirtingai nuo kitų čia pateiktų pavyzdžių, srauto planavimo metu ant kiekvieno tiesioginio lanko žymimas ne srauto dydis, bet potencialas (kiek dar yra nepanaudota iki galimos talpos).

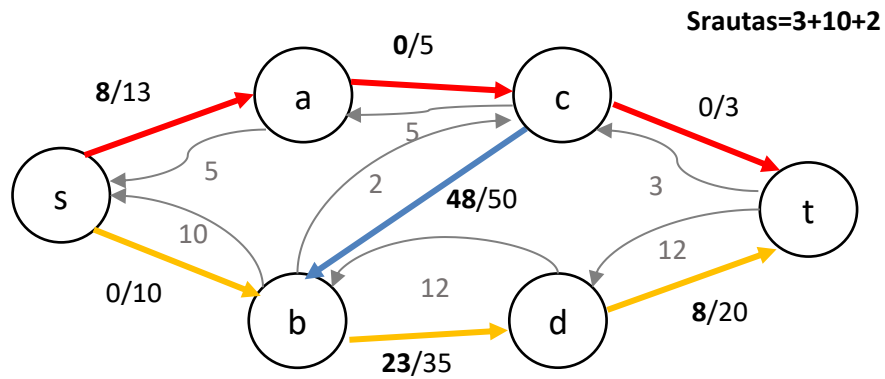


8 pav. Tinklas, kuriame srauto dar nėra



9 pav. Tinklas po dviejų srauto skirstymo žingsnių

9-e paveiksle matyti, kad trečią galimą srautą reikėtų planuoti nuo vadinamosios **silpnosios grandies** (angl. *bottleneck*), t. y., lanko $a-c$, kuriuo dar galima perduoti 2 srauto vienetų. Srautas papildomas nauju keliu $s-a-c-b-d-t$. Matome, kad likutiniame grafe sukuriamas tik lankas $b-c$. Lygiagretūs esamiems $s-a$, $a-c$, $b-d$, $d-t$ lankai 10-e paveiksle nebraižomi, nes tai nėra naujas kanalas, vamzdis ar pan., tiesiog panaudojamas jau buvusio lanko potencialas. Tačiau aiškinant temą pirmą kartą lentoje galima kelią $s-a-c-b-d-t$ nubraižyti net kita spalva, kad parodytume, kas čia atsiranda naujo.



10 pav. Tinklas po trijų srauto skirstymo žingsnių

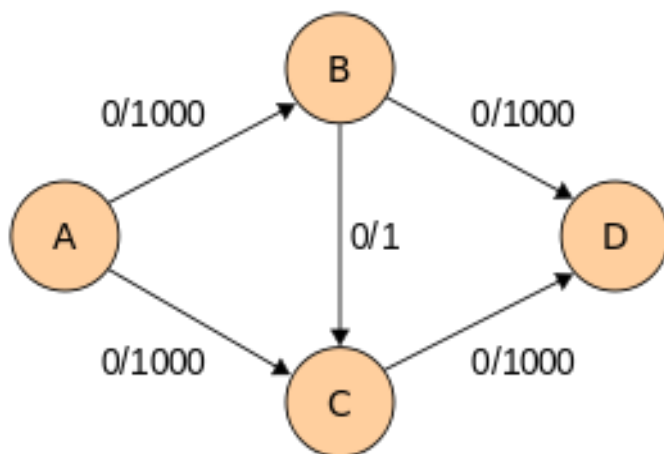
Abu paminėti vaizdavimo būdai yra lygiaverčiai. Konkretus būdas gali būti pasirenkamas atsižvelgiant į mokytojo, mokinių asmenines savybes ir (arba) būdus, kuriais jau buvo aiškintos kitos temos.

1-oji užduotis

Imkime bet kurį tinklą (kad ir iš 1 pav.) ir apskaičiuokime jo maksimalų srautą. Tada grafą patobulinkime: imkime vieną lanką ir jo viduryje įterpkime dar vieną viršūnę. Tarkime, tarp S ir V3 įterpkime N. Kadangi čia tik tarpinė stotelė, tai naujų lankų iki jos (S–N) ir nuo jos (N–V3) talpa yra tokia pati kaip ir buvusio perskelto lanko S–V3. Apskaičiuokime naujo tinklo srautą. Ar jis yra didesnis (mažesnis) už prieš tai buvusį? Kas pasikeistų, jei naująją viršūnę N sujungtume lanku dar su kuria nors kita viršūne (pvz., V2)?

2-oji užduotis

Imkime rombo formos tinklą (11 pav.) ir spręskime maksimalaus srauto uždavinį. Kiek žingsnių tam reikės? Kokią tendenciją pastebite?



11 pav. Tinklas, kuriame srauto dar nėra. Naudojami „srautas/talpa“ svoriai

https://en.wikipedia.org/wiki/Ford%E2%80%93Fulkerson_algorithm#/media/File:Ford-Fulkerson_example_0.svg

Edmondso ir Karpo algoritmas

Algoritmas nusakytas gana abstrakčiai, galimos skirtingos jo realizacijos, todėl kartais jis vadinamas ne algoritmu, o metodu. Tikslesniu laikomas Edmondso ir Karpo algoritmas, kuriame nurodyta, kad ieškoma ne bet kokio kelio, o trumpiausio galimos talpos kelio. Jo paprastai ieškoma atliekant paiešką platyn ir laikant, kad kiekvienos briaunos svoris (talpa) yra 1. Ieškant papildomų, srautą didinančių, kelių jų ilgis (skaičiuojant lankais) vis didėja.



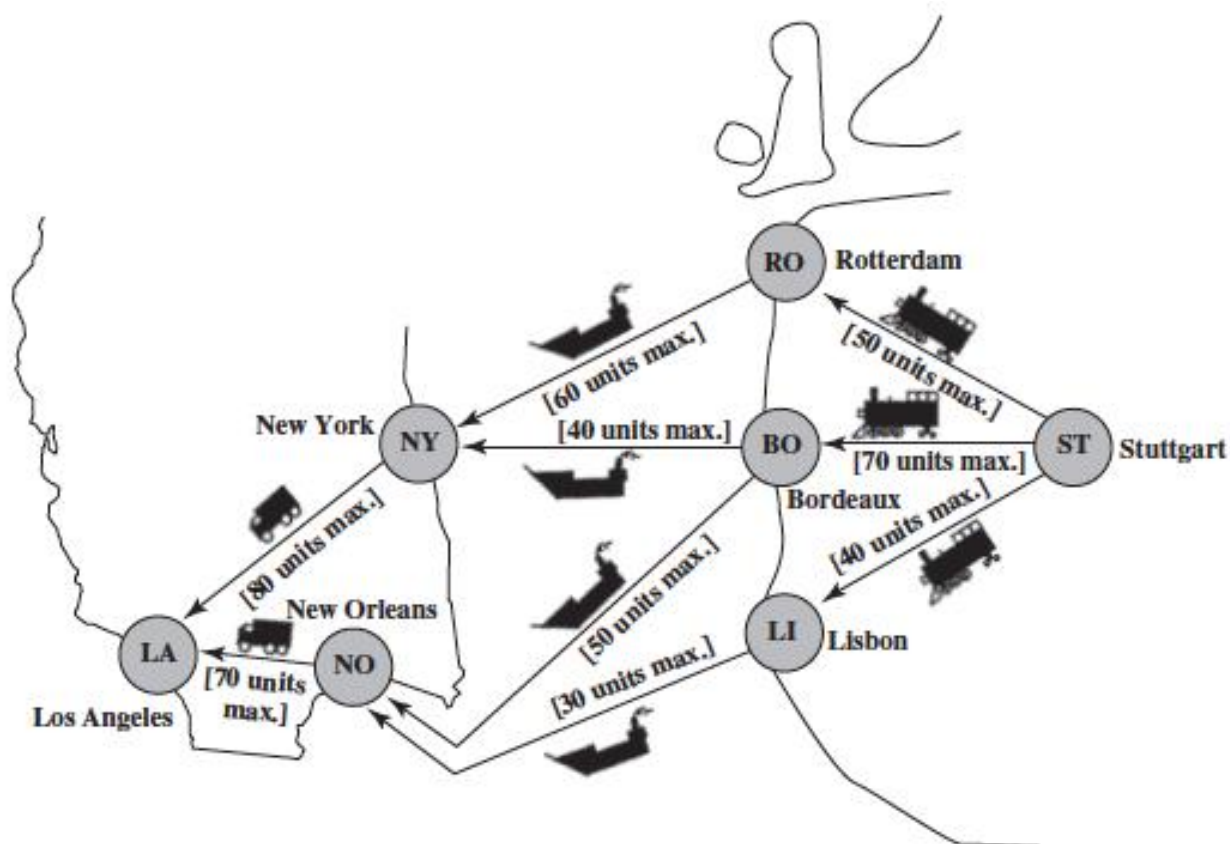
11 pav. Ričardas M. Karpas

https://en.wikipedia.org/wiki/Richard_M._Karp#/media/File:Karp_mg_7725-b.cr2.jpg

Įdomu tai, kad šis, jau pagerintas algoritmas kai kuriuose šaltiniuose vis tiek yra vadinamas Fordo ir Falkersono algoritmu.

1-oji veikla

Atidžiai išnagrinėkite maksimalaus srauto radimo algoritmus ir atlikite juos su šiuo tinklu:



Šaltinis: <https://henrywang.nl/wp-content/uploads/2018/12/Screenshot-2018-12-31-at-20.39.37.png>

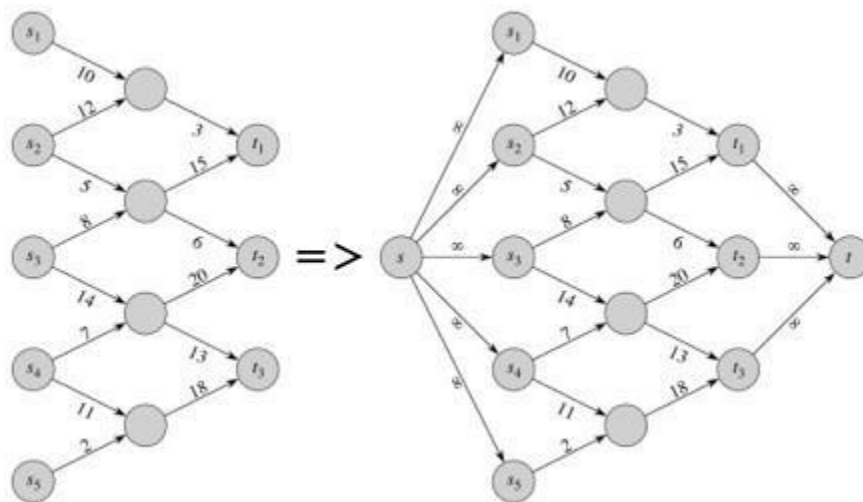
2-oji veikla

Mokyklos (ar kitos įstaigos) pastato plano pagrindu sudarykite tinklą, kurio lankais būtų koridoriai, laiptai, liftai, o viršūnėmis – jų susikirtimo taškai ir klasės. Įėjimą į mokyklą laikykite šaltiniu, o aktų salę – imtuvu. Surašę apytiksles kiekvieno lanko talpas pabandykite išspręsti uždavinį kiek daugiausia asmenų gali patekti į mokyklos organizuojamą renginį.

Bendrasis maksimalaus srauto uždavinys

Uždavinys atrodo sudėtingesnis, jei tinkle yra daugiau nei vienas šaltinis ir (arba) daugiau nei vienas imtuvas. Ypač jei dar keliamos sąlygos, kad, pavyzdžiui, visi imtuvai būtų „pamaitinti“ ar pan.

Vienas iš būdų supaprastinti uždavinį, kurio sąlygoje minimi keli šaltiniai – pertvarkyti tinklą taip, kad atsirastų dar viena viršūnė (menamas šaltinis, dar vadinamas **superšaltiniu**) ir sujungti ją su tikraisiais šaltiniais begalinės talpos lankais (12 pav.). Po to jau galima uždavinį spręsti taikant klasikinius maksimalaus srauto paieškos algoritmus. Jei tinkle yra keletas imtuvų, jis atitinkamai supaprastinamas įvedant **superimtuvą** viršūnę.



12 pav. Daugelio šaltinių tinklo pertvarkymas į klasikinį

<http://www2.hawaii.edu/~suthers/courses/ics311f20/Notes/Topic-20/Fig-26-3-Conversion-to-Single-Source-Sink.jpg>

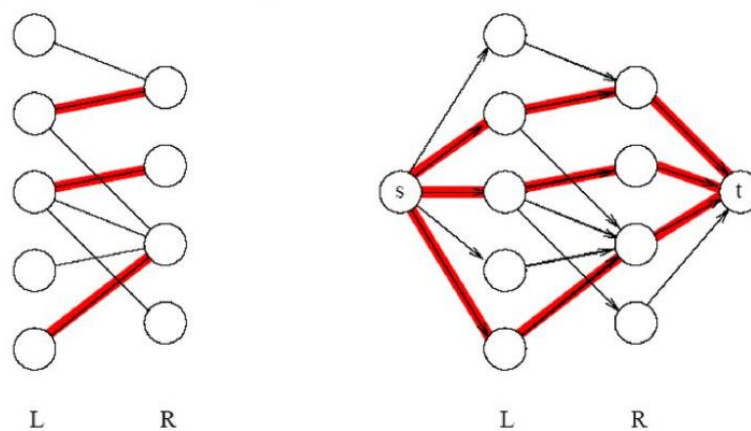
3-oji veikla

2-oje veikloje numatytu būdu sudarykite mokyklos (ar kitos įstaigos) pastato tinklą. Pasirinkite kabinetus, kuriuos laikysite viršūnėmis-šaltiniais, o mokyklos evakuacinius išėjimus – imtuvais. Taikydami maksimalaus srauto radimo algoritmus apskaičiuokite, ar saugu visiems laiku evakuotis iš mokyklos gaisro ar kt. nelaimės atveju. Jei ne, ką reikėtų keisti?

Dvišalio atitikimo uždavinys

Yra grupė uždavinių, susijusių su pasirinkimu. Grupė asmenų gali rinktis knygas, patiekalus, transporto priemones ir pan. Darbdaviai gali rinktis darbuotojus pagal jų kvalifikaciją ir pageidavimus. Vieni žmonės renkasi draugus (pvz., *Tinder* platformoje). Tokie uždaviniai vadinami **dvišalio atitikimo** (angl. *bipartite matching*) **uždaviniais**. Pirmasis toks uždavinys suformuluotas britų matematiko F. Halo ir vėliau pavadintas Halo vedybų uždaviniu.

Įdomu, kad tokius uždavinius galima suvesti į maksimalaus srauto paieškos uždavinį ir pastarąjį spręsti Fordo ir Falkersono metodu. Panagrinėkime pavyzdį (13 pav.). Kairėje pavaizduoti du viršūnių stulpeliai su objektais. Tarkime, L stulpelio objektai renkasi R stulpelio objektus. Kai kurie gali turėti daugiau kaip po vieną pageidavimą. Dabar L objektus sujunkime su vienu menamu šaltiniu, o R objektus – su menamu imtuvu. Gausime tinklą (13 pav.), kuriame galime ieškoti maksimalaus srauto (panaudoti keliai pažymėti raudona spalva). Tokiu būdu randame geriausius atitikmenis ir gauname dvišalio atitikimo uždavinio sprendinį.



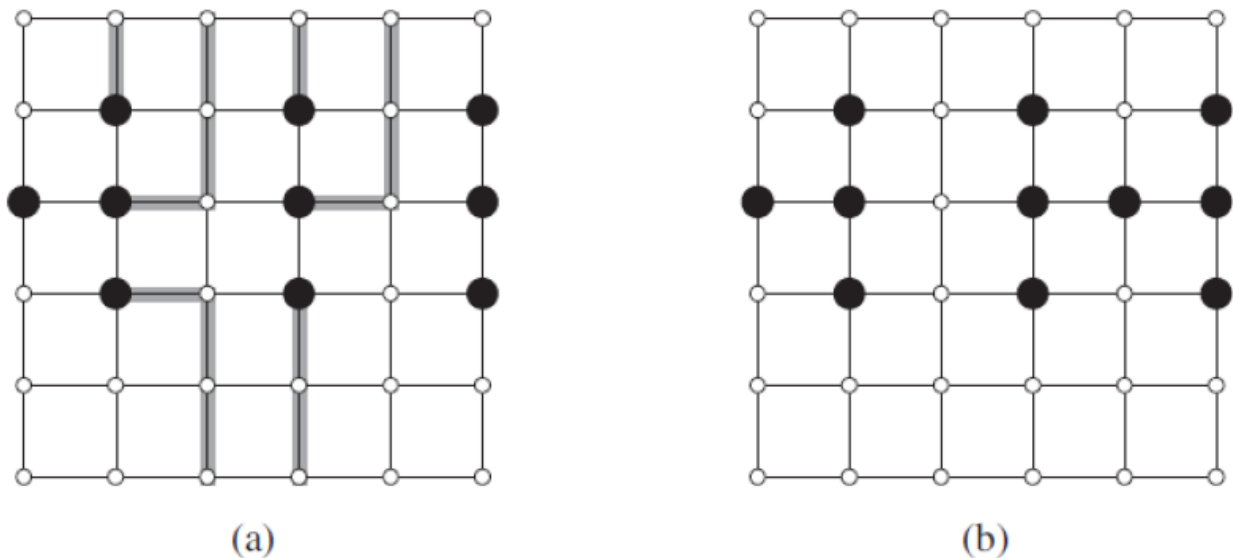
13 pav. Dvišalio atitikimo uždavinio pertvarkymas į maksimalaus srauto paieškos uždavinį

3-oji užduotis

Šią užduotį reikia atlikti dirbant 4–5 asmenų grupėje. Sudarykite knygų sąrašą (remiantis rekomenduojama perskaityti literatūra, knygų prekybos tinklalapio pirmuoju siūlomu sąrašu, knygyno „top 10“ eile ar pan.). Kiekvienas grupės narys turi pareikšti pageidavimus, kurias knygas norėtų perskaityti. Sudarykite pasirinkimų tinklą ir jame spręsdami maksimalaus srauto uždavinį pasidalinkite, kas ką skaitys.

Ištrūkimo uždavinys

3-oji veikla mus nukreipia ir į savitą uždavinių grupę, vadinamuosius ištrūkimo uždavinius. Tarkime, languoto $n \times n$ dydžio sąsiuvinio lape pavaizduotas grafas. Jame yra pažymėta nepriklausomų $m \leq n^2$ viršūnių, kuriose tupi kiškiai (14 pav.). Visiems kiškiams leidžiama keliauti grafo briaunomis sudarant skirtingus takus, tačiau bet kurie du takai negali turėti bendrų viršūnių. Reikia nustatyti, ar gali visi kiškiai pabėgti į grafo pakraštį. Esant mažam kiškių skaičiui ir palyginti nedideliame lauke šią užduotį galima išspręsti empiriniu būdu (ką tikriausiai jau ir atlikote), tačiau su didesniais skaičiais taip jau nebepavyks. Tokiu atveju **ištrūkimo uždavinį** rekomenduojama suvesti į maksimalaus srauto paieškos uždavinį.



14 pav. Ištrūkimo uždavinys

Tinklas jau beveik sudarytas (laikoma, kad kiekvienos briaunos talpa lygi 1), tik dar reikia jį papildyti:

- Visas viršūnes, kuriose tupi kiškis, sujungti su menamo šaltinio viršūne;

- Visas kraštines viršūnes sujungti su menamo imtuvo viršūne.

Kaip ir Dvišalio atitikimo uždavinyje maksimalus srautas lygus nepriklausomų takų tarp šaltinio ir imtuvo skaičiui m . To užtenka m kiškių pabėgti. Jei kiškių yra daugiau nei m , tuomet pabėgimas negalimas.

4-oji užduotis

Šią užduotį reikia atlikti dirbant poroje (arba dvejose asmenų grupėse). Reikia atlikti tokius veiksmus:

- Nubraižyti gatvių žemėlapij ir perduoti jį kitam asmeniui (grupei);
- Gavus žemėlapij iš kito asmens (grupės) jo pagrindu spręsti uždavinį: „Mokytojas Adomonis turi du vaikus Simą ir Rimą, kurie taip nemėgsta vienas kito, kad net nenori eiti ta pačia gatve iš namų į mokyklą. Simas neis gatve, jeigu ja jau eina Rimas ir atvirkščiai, tačiau sankryžose jų keliai gali liestis. Nustatykite, ar abu vaikai tokiu būdu gali nueiti į mokyklą, spręsdami šį uždavinį, kaip maksimalaus srauto uždavinį.“

Tiesinis programavimas

Pažymėtina, kad maksimalaus srauto uždavinį galima spręsti ir naudojant tiesinio programavimo algoritmus.

Srautų uždavinių sprendimas kompiuteriu

Vienas iš būdų šiam uždaviniui spręsti kompiuteriu – panaudoti MS Excel ar analogiškos skaičiuoklės funkciją Solver. Jos naudojimo pavyzdį galite rasti čia: <https://www.excel-easy.com/examples/maximum-flow-problem.html>

Kompiuterinės matematikos sistemos grafų ir tinklų uždaviniams spręsti turi specialias funkcijas. Jų naudojimo pavyzdžiai:

- <https://www.wolfram.com/mathematica/new-in-9/enhanced-graphs-and-networks/maximum-flows-and-minimum-cost-flows.html>

- <https://henrywang.nl/maximum-flow-problem-with-r/>

Žinoma, labai smagu ieškoti sprendimo kuriant patiems programą pasirinkta programavimo kalba!

Daugiau grafų algoritmų ir jų programavimo pavyzdžių galima rasti ir „YouTube“ platformoje:

- https://www.youtube.com/watch?v=tWVWeAqZ0WU&ab_channel=freeCodeCamp.org
- <https://www.youtube.com/watch?v=v1VgJmkEJW0&t=63s>

Šaltiniai

Cormen, T.H. et al. Introduction to Algorithms. 4th Ed. MIT Press, 2022, 1676 p.

Heineman, G.T. et al. Algorithms in a Nutshell: A Practical Guide. O'Reilly, 2016, 390 p.

Sabaliauskas, M. Algoritmai ir duomenų struktūros: Maksimalaus srauto paieškos algoritmai. 2019, <http://web.vu.lt/mii/m.sabaliauskas/wp-content/uploads/2019/04/11paskaita.pdf>

Williamson, D.P. Network Flow Algorithms. 2001, Cambridge University Press. 328 p.