EKSĀMENA MATERIĀLI

Grafu teorijas elementu

Autors
Grafu cienītājs

Saturs

1. Pamatjēdieni

1.1. Grafs

Def. Grafs G ir kopu pāris (V, E), kur V ir netukša kopa, kas sastāv no elementiem virsotnēm, un E ir kopa, kas sastāv no šķautnēm, kas savieno dažādu atsevišķu virsotņu pārus. Šī definīcija neiekļauj cilpas.

1.2. Incidence

Def. Ja $u \in E(x_1, x_2)$, tad saka, ka šķautne ved no virsotnes x_1 uz virsotni x_2 , bet trijnieku (x_1, u, x_2) sauc par grafa incidenci. Ja u ir šķautne, kas ved no x_1 uz x_2 , tad saka, ka x_1 un x_2 indicē jeb pieder šķautnei. Un otrādi - ka u indicē jeb pieder virsotnēm x_1 un x_2.

Def. Patvaļīgā grafā virsotni y sauc par blakusvirsotni virsotnei x, ja no x ved šķautne uz y. Līdzīgi divu vai vairāk dažādas šķautnes sauc par blakusšķautnēm, ja tām ir kopīga virsotne.

Def. Neorientēta šķautne ved gan no x_1 uz x_2 , gan otrādi. Orientēta šķautne ved vienā virzienā, piem., no x_1 uz x_2. Ja šķautne, kas ved no x_1 uz x_2 ir orientēta, tad x_1 sauc par sākuma, bet x_2 par beigu virsotni.

Def. Šķautni, kuras gala virsotnes sakrīt sauc par cilpu. Pretējā gadījumā to dažkārt sauc par īstu šķautni.

2. Grafu klases

2.1. Galīgs grafs

Def. Grafu sauc par galīgu, ja tam ir galīgs skaits šķautņu un virsotņu. Ja u indicē x_1 un x_2 , tad x_1 un x_2 ir gala virsotnes.

Def. Grafu, kam nav šķautņu sauc par bezšķautņu grafu. Grafu, kam nav īstu šķautņu (tikai varbūt cilpas) sauc par triviālu grafu.

2.2. Orientēts grafs

Def. Grafu sauc par orientētu, ja tam nav šķautņu, kas nebūtu orientētas, un par neorientētu, ja tam nav šķautņu, kas nebūtu neorientētas. Jaukts ir tāds grafs, kam ir gan orientētas, gan neorientētas šķautnes, neskaitot cilpas.

Def. Ja ir dots patvaļīgs orientēts grafs $G = (V, E)$ tad $\breve{E}(x, y) = E(y, x)$ G apvērstais grafs - apvērš škautņu orientācijas. $\overline{E}(x, y) = E(y, x) \cup \breve{E}(x, y) G$ piesaistītais grafs - atmet šķautņu orientācijas.

2.3. Vienkāršs jeb unigrafs

Def. Grafu sauc par vienkāršu jeb unigrafu, ja no katras tā virsotnes uz katru citu ved ne vairāk kā viena šķautne. Pretējā gadījumā (ja ir paralēlas šķautnes) - to sauc par daudzkāršu jeb multigrafu.

2.4. Berža grafs

Def. Par Berža grafu sauc ikvienu vienkāršu orientētu grafu, bet par parastu grafu - ikvienu neorientētu grafu bez cilpām.

2.5. Pilns grafs

Def. Parastu grafu, kam jebkuras divas dažādas virsotnes ir blakusvirsotnes sauc par pilnu grafu. Katram fiksētam virsotņu skaitam tāds ir tikai viens. Pilnu grafu ar p virsotnēm parasti apzīmē ar K_p .

2.6. Zvaigznes grafs

Def. Par zvaigznes grafu sauc parastu grafu, kurā ir tāda virsotne, kas ir. Vienīgā blakusvirsotne katrai no pārējām. Šāds grafs katrai fiksētai virsotnei ir tikai viens. Zvaigznes grafu ar p virsotnēm parasti apzīmē ar S_p−1 (p-1 ir staru skaits).
Def. Parastu grafu, kam jebkuras divas dažādas virsotnes ir blakusvirsotnes sauc par pilnu grafu (complete graph).

- Katram fiksētam virsotņu skaitam tāds ir tikai viens.
- Pilnu grafu ar n virsotnēm parasti apzīmē ar K_n
- $|E| =_n C_2$

Def. Par brīvu sauc tādu virsotni, kurai pieder viena īsta šķautne. Par izolētu sauc tādu virsotni, kurai nepieder neviena īsta šķautne. Ja virsotnei nav neviena šķautne, pat ne cilpa, tad to sauc par bezšķautņu virsotni.

2.7. divdaļīgs grafs

Def. Grafs ir divdaļīgs (bipartite), ja tā virsotnes var sadalīt 2 kopās K un L tā, lai katrai šķautnei $uv \in E$ viena no virsotnēm pieder K un otra L, t.i., $u \in K$, $v \in L$. Pilnam divdaļu (complete bipartite) grafam ir visas iespējamās šķautnes starp abu virsotņu kopu K un L virsotnēm. Pilnu grafu ar n virsotnēm parasti apzīmē ar K_n,m. Divdaļu grafam raksturīgi, ka visi cilki ir ar pāra skaita garumu (taču var arī nebūt divdaļu grafā cikli).

2.8. Rats

Def. Par rata (wheel) grafu ar kārtu n sauc grafu kas sastāv no $n - 1$ gara cikla, kura ikviena no virsotnēm ir savienota ar vienu kopīgu virsotni. Rata grafu ar n virsotnēm parasti apzīmē ar Wn.

2.9. Sakarīgs grafs

2.9.1. Definīcijas

Def. Grafs ir sakarīgs, ja jebkurām virsotnēm u, v eksistē ceļš no u uz v.
Def. Virsotnes atvērta apkārtne $N_G(v) = \{u|uv \in E\}$ - visas virsotnes, kas ir blakusvirsotnes $v \in G$.
Def. Virsotnes slēgta apkārtne $N_G[v] = \{u|uv \in E\} \cup \{v \in V\}$- visas virsotnes, kas ir blakusvirsotnes $v \in G$ un ietver arī pašu virsotni v.

2.9.2. Sakarīga komponente

Def. Grafa sakarīgā komponente $G' = (V', E')$, kur V' ir maksimālā virsotņu kopa ar īpašību, ka katrām divām virsotnēm u,$v \in V0$ eksistē ceļš $u \rightarrow v$, $E' : uv \in E$, $u \in V0, v \in V0$.

Def. Par grafa G sakarīgo komponenti jeb komponenti sauc jebkuru maksimālu sakarīgu apakšgrafu. Maksimāls nozīmē, ka apakšgrafam zūd sakarības īpašība, ja tam pievieno vēl kādu no grafa G virsotnēm. Tas nozīmē, ja apakšgrafu var papildināt līdz lielākam, bet tomēr vēl sakarīgam, tad vēl nav bijusi aplūkota komponente. Sakarīgu komponenšu skaitu apzīmē ar s. Grafu G sauc apr viensakarīgu, ja tas ir sakarīgs, bet atmetot vienu virsotni v līdz ar tai incidentām šķautnēm, rodas nesakarīgs grafs. Šādu virsotni v sauc par sasaistes virsotni.

2.9.3. k-sakarība

Def. Grafu G sauc par divaskarīgu, ja tas nav viensakarīgs, bet atmetot 2 virsotnes v_i un v_j un tām incidentās šķautnes - rodas nesakarīgs grafs. Trīssakarīgs ir tāds grafs, kas nav divsakarīgs, bet atmetot 3 virsotnes ar tām incidentām šķautnēm, rodas nesakarīgs grafs.

Daži fakti:

- Koks ir 1-sakarīgs, bet nav 2-sakarīgs.
- Ja grafs ir nesakarīgs, tad tas ir 0-sakarīgs un nav 1-sakarīgs.
- Grafs ir sakarīgs, tad un tikai tad, ja tas ir 1-sakarīgs.
- Pilns grafs K_n ir (n – 1)-sakarīgs.
- Ja grafs ir k-sakarīgs, tad $k \leq$ min($d(v)$), kur $d(v)$ ir virsotnes v pakāpe.

2.9.4. Sadalošā un atdalošā kopa

Def. Sakarīga grafa $G = (V, E)$ virsotņu apakškopu S sauc par sadalošo kopu, ja grafs $G(V \backslash S)$ ir nesakarīgs. Sakarīga grafa sadalošā virsotne veido sadalošo kopu. Grafam K_n nav nevienas sadalošās kopas.

Def. Grafa $G = (V, E)$ virsotņu apakškopu S sauc par virsotnes x un y atdalošo kopu, ja x un y pieder dažādām $G(V \backslash S)$ sakarīgajām komponentēm.

Def. Šķēlums ir šķautņu kopa, kuru izmetot grafs sadalās divās daļās.

3. Grafa raksturlielumi

3.1. Virsotnes pakāpe

3.1.1. Definīcijas

Def. Par neorientēta grafa virsotnes pakāpi $d_v(degv)$ sauc skaitli, ko iegūst saskaitot šai virsotnei piederošo īsto šķautņu skaitu ar divkāršotu tai piederošo cilpu skaitu. Vizuāli tas ir virsotnei piederošķautņu galu skaits.

Def. Par neorientēta grafa virsotnes pakāpi $d_v(degv)$ sauc skaitli, ko iegūst saskaitot šai virsotnei piederošo īsto šķautņu skaitu ar divkāršotu tai piederošo cilpu skaitu. Vizuāli tas ir virsotnei piederošķautņu galu skaits, vienāds ar atvērtas apkārtnes virsotņu skaitu min pakāpe $\delta(G)$ max pakāpe $\Delta(G)$.

3.1.2. Havela–Hakimi teorēma (Rokasspiediena lemma)

T. Neorientēta grafa visu virsotņu pakāpju summa ir vienāda ar divreiz šķautņu skaitu grafā.

$$\sum_{v \in V(G)} degv = 2|E(G)| \tag{1}$$

S. Visu virsotņu pakāpju summai jābūt pāra skaitlim.
S. Grafā ir pāra skaits virsotņu ar nepāra pakāpēm.
P. Katrā sakarīgā komponentē ir pāra skaits virsotņu ar nepāra pakāpi.
Havela-Hakimi teorēma nodrošina algoritmu, lai pārbaudītu, vai noteiktā pakāpju secība ir atbilst grafam, citiem vārdiem, noskaidro, vai tāds grafs eksistē.
Algoritms:

- Sakārtojiet grādu secību, kas nav pieaugoša.
- Noņemiet pirmo skaitli n un samaziniet nākamos n skaitļus katru par 1.
- Atkārtojiet šo procesu, līdz:
  - secība ir tukša, tādā gadījumā sākotnējā secība ir grafiska vai
  - jūs saskaraties ar negatīvu skaitli vai jums ir jāsamazina vairāk skaitļu, nekā jums ir atlicis, tādā gadījumā secība nav grafiska.

3.1.3. Homogēns grafs

Def. Par homogēnu (regulāru, r-regulāru) grafu sauc tādu grafu, kam visu virsotņu pakāpes ir vienādas.
S. Ja grafā ir tieši 2 virsotnes ar nepāra pakāpi, tad eksistē ceļš no vienas uz otru.

4. Virsotņu virknes

Def. Par maršrutu (walk) grafā sauc tādu šī grafa virsotņu un šķautņu virkni $x_0u1x_2u_2x_3...u_lx_l$, kur $l \geq 0$ un katrs trijnkieks (x_{i-1}, ui, x_i) ir grafa incidence.Tas ir, katra šķautne ui ved no virsotnes x_{i-1} uz xi.
Def. Noslēgts maršruts - sākuma un beigu virsotnes sakrīt ($x_0 = x_l$)
Def. Virsotni x_0 sauc par maršruta sākuma virsotni un virsotni x_l sauc par maršruta beigu virsotni, bet l sauc par maršruta garumu.
Def. 0-maršruts sastāv no vienas virsotnes. Tas ir triviāls jeb tukšs maršruts.
Def. Ķēde - maršruts, kurā nav šķautnes, kas atkārtotos.
Def. Elementāra/vienkārša ķēde - nav virsotnes, kas atkārtojas, izņemot varbūt gala un sākuma virsotni.
Def. Cikls - noslēgta, netukša ķēde.
T. Ja ir maršruts $a \to b$, tad ir arī ceļš $a \to b$.

4.1. Divdaļīga grafa ciklu definīcija

T. Grafs ir divdaļīgs $\Leftrightarrow$ katram ciklam ir pāra garums.

4.2. Divdaļu grafa sakārtojuma konstruēšanas algoritms

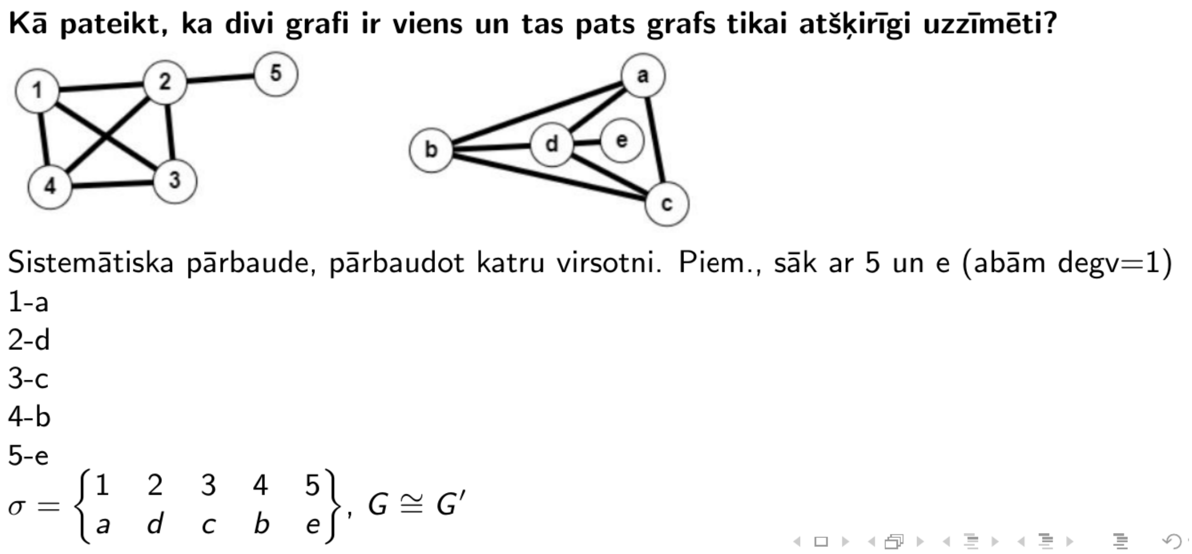
- ievieto kādu virsotni u virsotņu kopā K (iekrāso vienā krāsā)
- ievieto visas virsotnes v, kas ir incidentas u, virsotņu kopā L (iekrāso otrā krāsā)
- ieveto visas virsotnes v, kur īsākais ceļš no u uz v sastāv no 2, 4, ... šķautnēm virsotņu kopā K (turpina krāsot iepriekšējā solī iekrāsoto virsotņu blakusvirsotnes pretējā krāsā)
- ieveto visas virsotnes v, kur īsākais ceļš no u uz v sastāv no 3, 5, ... šķautnēm virsotņu kopā L (turpina krāsot iepriekšējā solī iekrāsoto virsotņu blakusvirsotnes pretējā krāsā)

5. Grafu izomorfisms

Def. Divi grafi G un G' ir izomorfi, ja pastāv savstarpēji viennozīmīgs attēlojums (bijekcija) starp grafa G un grafa G' virsotnēm, tāda ka virsotnes ir incidentas grafā G $\Leftrightarrow$ ja atbilstošās virsotnes ir incidentas grafā G'. Šādu savstarpēji viennozīmīgu atbilstību, kas saglabā virsotņu incidenci sauc par izomorfismu. Izomorfos grafos ir: vienāds skaits virsotņu; vienāds skaits šķautņu; vienāds skaits virsotņu ar noteiktām virsotņu pakāpēm, turklāt; izomorfisms saglabā virsotņu pakāpes, tas ir, atbilstošajām virsotnēm; abos grafos ir vienāda pakāpe strukturālas līdzības (piem., divdaļīgs grafs, regulārs grafs).

5.1. Sistemātiskā pārbaude

Sistemātiska pārbaude, pārbaudot katru virsotni. Piem., sāk ar 5 un e (abām $degv = 1$).



1. zīm.Sistemātiskās pārbaudes piemērs

5.2. Papildgrafs

Def. Par grafa $G = (V, E)$ papildgrafu $G = (V, E)$ sauc grafu, kurš satur visas grafa G virsotnes, bet šķautņu kopu veido tās virsotnes, kas G nav incidentas. Grafa G apvienojums ar $\bar{G}$ veido pilnu grafu divi grafi G_1 un G_2 ir izomorfi $\Leftrightarrow \overline{G}_1$ un $\overline{G}_2$ ir izomorfi papildgrafu var izmantot grafu izomorfisma noteikšanā.

6. Planāri grafi

Dabisks grafu piemērs ir ceļu kartes. Šādas kartes raksturo īpašība - tos var uzzīmēt ar šķautnēm (ceļiem) krustojoties tikai krustpunktos virsotnēs (krustcelēs).

Def. Grafs ir planārs, ja to var uzzīmēt plaknē tā, ka šķautnes nekrustojas.

6.1. Apakšgrafs, parciālgrafs, daļgrafs

Def. Grafa G apakšgrafs G' ir grafs, ko veido grafa G virsotņu un šķautņu kopu apakškopas. Par grafa G apakšgrafu sauc tādu grafu G', ko iegūst atmetot grafa G jebkuru skaitu virsotņu kopā ar tām piederošajām šķautnēm un tāpat, jebkuru skaitu šķautņu bez to gala virsotnēm Izmetot šķautnes no planāra/plakana grafa → iegūst planāru/plakanu grafu pievienojot jaunas šķautnes planāram/plakanam grafam → var grafu padarīt neplanāru/neplakanu.

Def. Maksimāli planārs grafs - ja tas pārstāj būt parasts un planārs pēc jaunas šķautnes pievienošanas vienalga starp kurām divām tā virsotnēm.

Jebkurš grafs ar n virsotnēm ir pilna grafa K_n apakšgrafs ja divi grafi ir izomorfi, tad apakšgrafi, kas veidojušies no atbilstošām virsotnēm un šķautnēm arī ir izomorfi apakšgrafu var izmantot grafu izomorfisma noteikšanā

Def. Par grafa $G = (V, E)$ parciālgrafu sauc grafu $G0 = (V, E*)$, $E* \subset E$, kura virsotņu kopa ir vienāda ar grafa G virsotņu kopu, bet šķautņu kopa ir daļa no G šķautņu kopas. Grafa $G = (V, E)$ daļgrafs $G_0 = (V_0, E_0)$ satur gan daļu no grafa virsotnēm, gan šķautnēm, tomēr katrai šķautnei $e \in E0$ virsotnes $e = (v_i, v_j)$ pieder kopai $V0 \subset V$.

Def. Plakans grafs ir tāds planārs grafs, kas ir uzzīmēts tā, ka šķautnes nekrustojas. Plakans grafs sadala plakni regionos.

Def. Cilku, kas ierobežo plaknes daļu, kura nesatur ne virsotnes, ne šķautnes sauc par robežcilku.

Def. Par plakana grafa skaldni (face) sauc ikvienu plaknes daļu (arī neierobežoto, bezgalīgo), ko norobežo robežcikls. Ja grafā ciklu nav - par skaldni uzskata visu planki. Ārējo robežciklu veido šķautnes, kas atdala grafu no bezgalīgās skaldnes. Skaldņu skaitu apzīmē ar r. Ja skatās uz jebkuru skaldni maksimālā planārā grafā - visi ir trīsstūri jebkuras skaldnes robeža var tikt pārveidota kā ārējā robeža. Skaldņu skaits paliek nemainīgs. Skaldnes pakāpe df ir šķautņu skaits, kas ierobežo skaldni.

6.2. Apļa-hordu metode (Circle-chord method)

→ no pretējā

- pieņem, ka ir plakans/planārs
- ja pretruna - tad pieņēmums ir aplams

- atrod cilku, kas satur visas grafa virsotnes (var gadīties, ka nav tāds cikls, bet šobrīd skatamies gadījumus, kas ir)
- uzzīmē ciklu kā lielu apli

- atlikušās šķautnes-hordas jāzīmē iekšpus vai ārpus apļa plaknē
- pirmo hordu izvēlas zīmēt piemēram, ārpus apļa.

  Šī horda liks citām būt iezīmētām iekšpusē (ja zīmētu ārpusē, tad krustotos) Iekšpuses hordas savukārt liks citas zīmēt ārpusē.

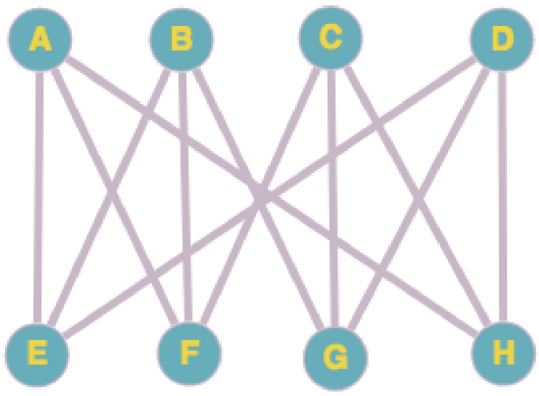
  Ja nonāk pie situācijas, kad nav iespēja ievilkt šķautni bez krustošanās, tad nav planārs.

  Ja var visas šķautnes savilkt bez krustošanās, tad ir plakans.

Nav nozīme, vai pirmo šķautni velk iekšpus vai ārpus apļa. Pastāv efektīvi algoritmi, kā noteikt, vai grafs ir planārs, atšķirībā no izomorfisma.

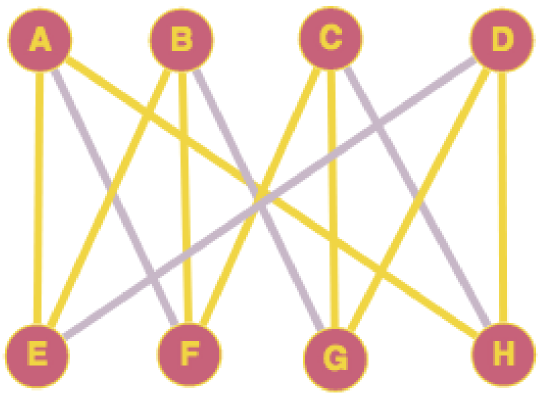
6.2.1. Piemērs

Sāksim ar divdaļīgu grafu (grafs var būt jebkura veida).



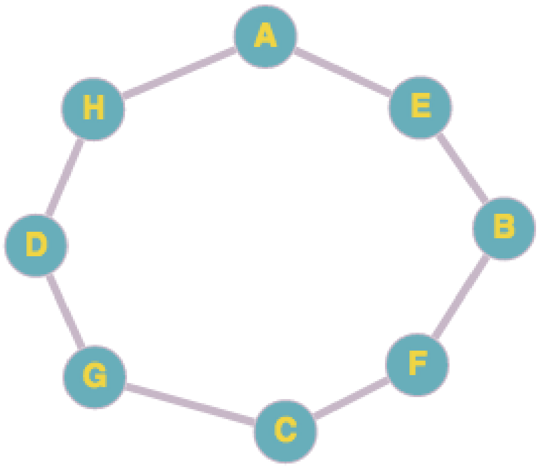
2. zīm.Planārs vai neplanārs grafs

1. darbība: Atrodiet ķēdi, kurā ir visas grafika virsotnes.



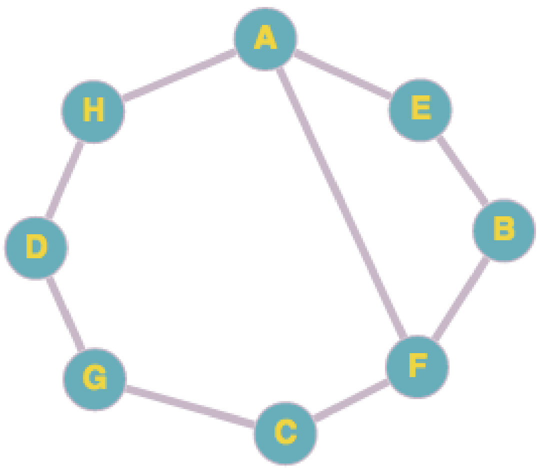
3. zīm.Ķēde ar visām virsotnēm (dzeltenā krāsā)

2. darbība. Izvietojiet atrasto ķēdi kā lielu apli. Iezīmējiet virsotnes, apzīmējot tās tā pat kā oriģinālajā grafā.



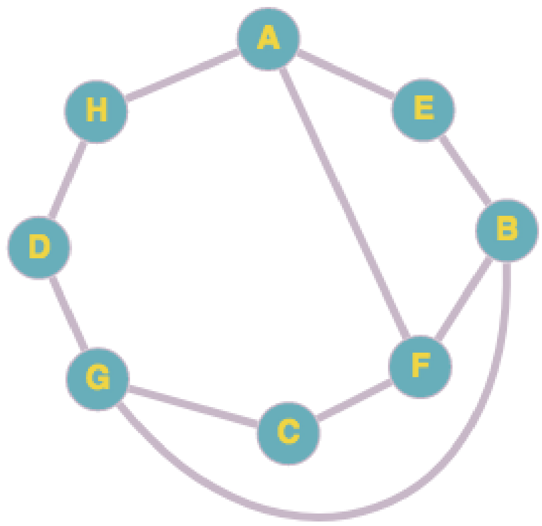
4. zīm.Ķēde izvietota apļa veidā

3. darbība. Pievienojiet hordas, no sākotnējā grafa. Piemēram A virsotnei. Tā kā virsotne A bija savienota ar virsotni E un virsotni H ar apli, man bija jāsavieno tikai virsotne A un virsotne F. – Varat to uzzīmēt apļa iekšpusē vai ārpusē.



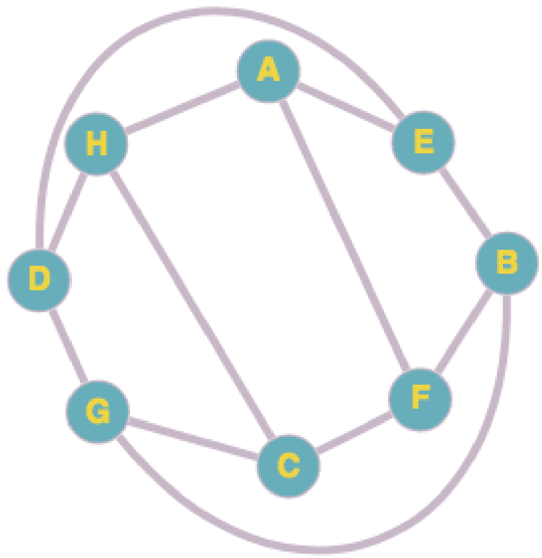
5. zīm.Hordas ir pievienotas virsotnei A

Tālāk ir jāsavieno virsotne B ar virsotni G. Mēs cenšamies izvairīties no krustojumiem, tāpēc mums ir jānovelk horda tā, lai tā krustotos ar AF. Tas nozīmē, ka mums ir jāiziet ārpus apļa.



6. zīm.Hordas ir pievienotas virsotnei B

Turpinam šo procesu citām virsotnēm.



7. zīm.Visas hordas ir pievienotas

4. darbība. Tā kā mēs varējām uzzīmēt visas hordas bez krustojumiem (vienalga, apļa iekšpusē vai ārpusē), tad grafiks ir planārs un mēs ieguvām attiecīgo plakano grafu.

6.3. Eilera teorēma

T. Eilera teorēma par plakaniem grafiem - Ja G ir sakarīgs un plakans grafs ar p virsotnēm, q šķautnēm un r skaldnēm, tad $p - q + r = 2$. p - virsotņu skaits, q - šķautņu skaits, r - skaldņu skaits.

S. Jebkurā sakarīgā plakanā grafā izpildās $p - q + r - s = 1$. p - virsotņu skaits, q - šķautņu skaits, r - skaldņu skaits, s - sakarīgo komponenšu skaits.

T. Ja G ir maksimāls plakans grafs ar $p \geq 3$ virsotnēm un m šķautnēm, tad $q = 3p - 6$. p - virsotņu skaits, q - šķautņu skaits

S. Ja grafs G ir sakarīgs plakans grafs ar $p \geq 3$ virsotnēm un q šķautnēm, tad $q \leq 3p - 6$. p - virsotņu skaits, q - šķautņu skaits

Tomēr arī neplanāri grafi var apmierināt šīs sakarības. Ja neizpildās ⇒ nav plakans; Ja izpildās ⇒ vēl nevar droši secināt, ka ir plakans. L. Ja G ir plakans sakarīgs grafs ar p virsotnēm un q šķautnēm un bez trīsstūru skaldnēm, tad $q \leq 2p - 4$

6.4. Planāru grafu atpazīšana

Kā atpazīt neplanārus grafus? Šie 3 apgalvojumi ir ekvivalenti:

- Grafs ir planārs
- Grafs nesatur K_5 vai K_{33} kā konfigurāciju
- Grafs nesatur K_5 vai K_{33} kā minoru

Teiksim, ka grafs G ir reducējams uz grafu G', ja tas sakrīt ar G' vai arī G' var iegūt no G veicot vienu vai vairākas reizes patvaļīgā secībā šādas redukcijas procedūras: no grafa izmet vienu (jebkuru) šķautni, atstājot galavirsotnes no grafa izmet jebkuru virsotni un tai incidentās škautnes ja x u y v z ir divu šķautņu ceļš grafā un virsotnei y ir pakāpe 2, bet $x \neq z$, tad abas šķautnes u un v un to kopīgo virsotni izņem, bet vietā ielik jaunu šķautni ar gala virsotnēm x un z.

Piezīme. Izmantojot pirmos divus punktus, jebkuru grafu var reducēt uz vienu virsotni.

6.5. Kuratovska un Pontrjagina teorēma

T. Grafs nav planārs ⇔ to var reducēt uz K_5 vai K_{33} Iepriekš esam pierādījuši, ka K_5 vai K_{33} nav planāri. Redukcijas ceļā neplanāru grafu var iegūt tikai no neplanāra grafa.

S. Grafs ir planārs ⇔ ja to nevar reducēt ne uz K_5 , ne K_{33}

Kuratovska T. Grafs G ir planārs 1 ja tas neveido ne K_5 , ne K_{33} daļgrafu, nedz arī tiem līdzīgus (homomorfus) grafus

Def. Plakanam grafam var piekārtot duālo grafu, kurā skaldnes=virsotnes un robežškautnes=šķautnes (t.i., šķautnes saglabājas).

6.6. Konfigurācija

Def. Grafu ir iespējams pārveidot, pievienojot virsotnes šķautņu vidū. Rezultātā tiek iegūts grafs, kas vairs nav, piem., K_{33} , un neietver K_{33} kā apakšgrafu, tomēr tas joprojām ir neplanārs. Pievienojot neplanāram grafam virsotnes nevar iegūt planāru grafu.

Def. Elementāra konfigurācija tiek iegūta grafam G noņemot šķautni e=uv un pievienojot jaunu virsotni w un divas šķautnes uw un vw. Grafa konfigurācija ir grafs, kas iegūts no G secīgi veicot 0 vai vairākas elementāras konfigurācijas.

6.6.1. K_{33} konfigurācija

T. Jebkura grafa G konfigurācija H ir planāra 1 G ir planārs (Šķautņu sadalīšana nemaina planaritāti.)

Def. Saka, ka grafs ir K_{33} konfigurācija, ja tas var tikt iegūts no K_{33}, pievienojot virsotnes šķautņu vidū.

Def. Saka, ka grafs ir K_5 konfigurācija, ja tas var tikt iegūts no K_5 , pievienojot virsotnes šķautņu vidū.

K_5 pievienotas virsotnes ar šķautnēm. K_5 ir izveidotā grafa apakšgrafs. K_5 šķautnes sadalītas ar jaunām virsotnēm. K_5 nav apakšgrafs, bet nav arī planārs. Ja izņemtu jaunas virsotnes, tad arī nav planārs.

Kuratovska T. Grafs G ir planārs 1 ja tas nesatur K_5 vai K_{33} konfigurāciju

6.7. Grafu reducēšana (minors)

Reducēšanas operācijas:

- Šķautņu dzēšana $E\backslash\{e\}$
- Virsotņu dzēšana $V\backslash\{v\}$
- Šķautņu savilkšana - abas šķautnes galavirsotnes savelk kopā (šķautnes e gala virsotnes u un v izņem un aizvieto ar w, w ir incidenta ar visām u un v blakusvirsotnēm) (multigrafs, parasts grafs). Citiem vārdiem, šķautnes džēšana ar ar virsotņu saplūšanu.

Def. Grafs H ir grafa G minors, ja to var iegūt no G veicot šādas redukcijas: dzēš šķautni savelk šķautni dzēš izolētas virsotnes (Grafs G ir pats sev minors, pielietojot 0 redukcijas)

6.8. Vāgnera teorēma

Vāgnera teorēma (Wagner's theorem) - Katrs grafs, kas ir izomorfs G minoram, arī ir G minors.

Def. Grafs G satur grafu H kā minoru, ja G satur apakšgrafu G', ko var savilkt uz H.

Vāgnera **T.** Grafs G ir planārs 1 tam nav vai K_{33} minora (G nesatur K_5 vai K_{33} kā minoru)

6.9. Kuratovska T. vs Vāgnera T.

Kāda saistība starp abām teorēmām? G minoru ne vienmēr var pārveidot G konfigurācijā. Ja G satur ≥ 1 K_5 vai K_{33} kā minoru, tad tas satur ≥ 1 K_5 vai K_{33} kā konfigurāciju.

7. Grafu krāsošana

Def. Pareizi nokrāsots grafs - piešķirt grafa virsotnēm krāsas tā, lai incidentām virsotnēm būtu atšķirīgas krāsas. Hromatiskais skaitlis χ ir mazākais skaits krāsu, ar ko var pareizi nokrāsot grafu. k-krāsojums - grafs G ir nokrāsots k krāsās. Ja ir nokrāsots ⇒ grafs G ir k-nokrāsojams, $\chi = k$, grafs G ir k-hromatisks.

Def. Dots grafs $G = (V, E)$ K-krāsojums sadala virsotņu kopu V kopās $V_1, V_2, ...V_k$, kur katra kopa V_i neatkarīgu virsotņu kopu (neviena no virsotnēm nav savā starpā incidentas). $V = V_1 u V_2 \cup \ldots \cup V_k$; $V_i \cap V_j =$ visiem $i \neq j$; Neatkarīgās kopas $V_1, V_2, \ldots V_k$ sauc par krāsu klasēm.

Ar cik krāsām var nokrāsot dotu grafu? Grafam ar $\leq$ 15 virsotnēm parasti nav grūti atrast hromatisko skaitli. Lai pārbaudītu, ka grafa hromatiskias skaitlis ir k, ir jāpārbauda arī, ka grafs nevar tikt nokrāsots k − 1 krāsās. Mērķis parādīt, ka jebkurš k − 1 krāsojums noved pie tā, ka divas incidentas virsotnes ir vienā krāsā.

7.1. Daži fakti

- Ja G ir n virsotnes,;
- tad $\chi(G) \leq n$;
- $\chi(G) = 1 \Leftrightarrow G$ ir tukšs grafs;
- $\chi(Kn_2) = n$⇒G ir pilns grafs;
- Cikls ar pāra šķautnēm $\chi(C_{2n}) = 2$;
- Cikls ar nepāra šķautnēm $\chi(C_{2n=1}) = 3$;
- Rats ar pāra zariem $\chi = 3$;
- Rats ar nepāra zariem $\chi = 4$;
- Ja H ir apakšgrafs $\chi(G) \geq \chi(H)$.

7.2. 2-hromatiski grafi

Fakts: Katrs grafs-koks ar vismaz 2 virsotnēm ir 2-hromatisks. **T.** Grafa G hromatiskais skaitlis $\chi(G)$ ir 2⇔G ir divdaļīgs. **T.** Grafa G hromatiskais skaitlis $\chi(G)$ ir 2⇔G ir divdaļīgs.

7.3. Hromatiskā skaitļa augšājā robeza

T. Katru grafu var nokrāsot ar $d + 1$ krāsām, kur d ir grafa maksimālā pakāpe, $\chi(G) \leq 4(G) + 1$.

7.4. Planāru grafu krāsojums - Eilera formula

T. Planārā grafā $q \leq 3p$−6, $p \geq 3$, q - šķautņu sk., p - virsotņu sk.

S. (min pakāpe planārā grafā). Planārā grafā ir virsotne v, kam $dv \leq 5$. 6 krāsu **T.** Jebkuru planāru grafu var pareizi nokrāsot ar 6 krāsām. 5 krāsu **T.** Ja grafs ir planārs, sakarīgs un vienkāršs, tad $\chi(G) \leq 5$.

8. Eilera ķēdes un cikli

Def. Eilera ķēde grafā ir jebkurš maršruts, kas satur visas grafa šķautnes. Ja tāds maršruts ir cikls, tad to sauc par Eilera ciklu. (Satur katru grafa šķautni tieši 1 reizi) Grafu, kurā eksistē vismaz viens Eilera cikls sauc par Eilera grafu.

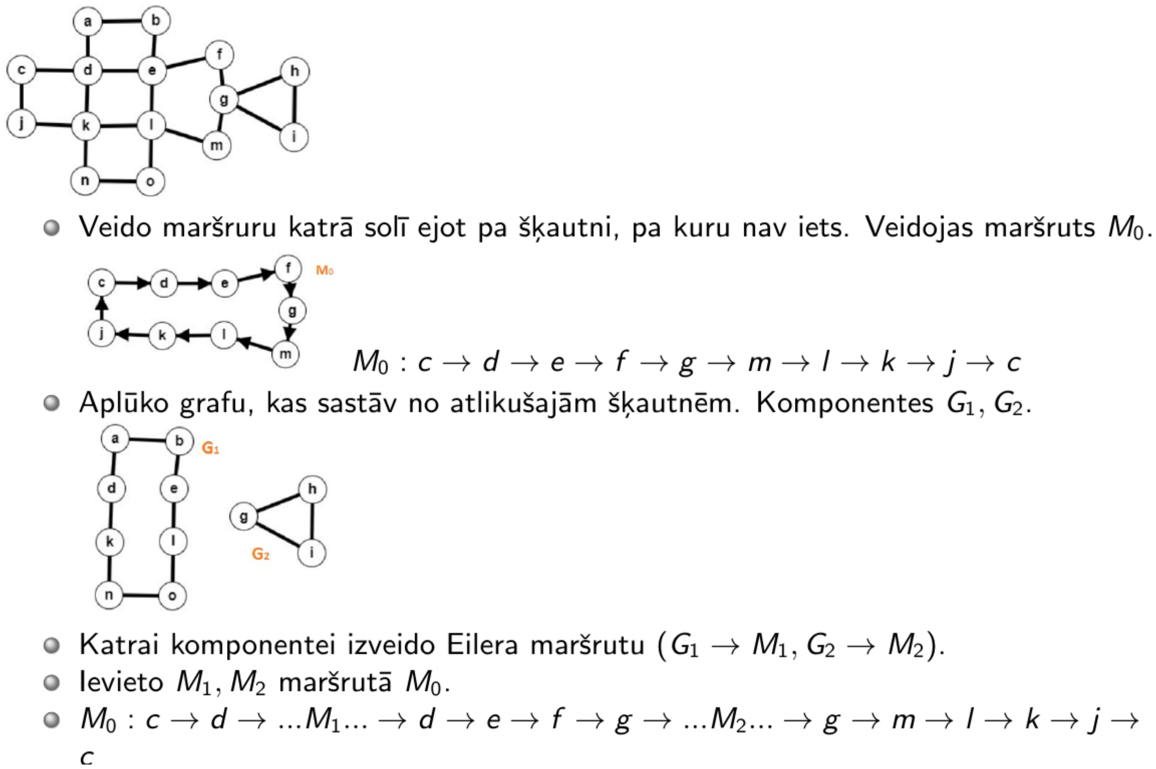
T. Ja grafā ir virsotne v ar nepāra pakāpi dv , tad tajā nav noslēgta Eilera maršruta (Eilera cikla).

8.1. Eilera cikls. Algoritms.

Kā grafā konstruēt Eilera ciklu?
Algoritms:

- Veido maršrutu no brīvi izvēlētas virsotnes katrā solī ejot pa šķautni, pa kuru iepriekš nav iets, līdz tas vairs nav iespējam

- (Veidojas maršruts M0)
- Aplūko grafu, kas sastāv no atlikušajām šķautnēm. (Komponentes G_1 , G_2 utt.)
- Katrai komponentei izveijo Eilera maršrutu ($G_1 \rightarrow M_1$, $G_2 \rightarrow M_2$ utt.)
- Ievieto M_1 , M_2 utt. maršrutā M0
- Algoritms vienmēr atrod noslēgtu Eilera maršrutu (Eilera ciklu), ja tāds eksistē.



8. zīm.Piemērs

8.2. Eilera grafs

T. Grafs bez izolētām virsotnēm ir Eilera grafs $\Leftrightarrow$ šis grafs ir sakarīgs un katrai tā virsotnei ir pāra pakāpe.

T. Ja grafā ir nenoslēgts Eilera maršruts, tad visām virsotnēm, izņemot sākuma un beigu, pakāpe dv ir pāra.

T. Grafā eksistē nenoslēgta Eilera ķēde ⇔ grafs ir sakarīgs un a $r \leq 2$ virsotnēm ar nepāra pakāpi.

8.2.1. Grafa sadalīšana

Grafu sadala tā, ka katrā izdalītajā apakšgrafā ir atšķirīgas šķautnes Aplūko grafu G Var atrast divus 3 šķautņu ciklus Nav noteikti jābūt cikliem vai vienādām daļām $F = F1, F2, E(F1) \cup E(F2) = E(G)$

Def. Grafa sadalījums ir saime F ar apakšgrafiem $F1, F2...Fn$, kas sastāv no atšķirīgām šķautnēm tā, ka $uE(F) = E(G)$. Ja katrs apakšgrafs saimē F ir cikls, tad sadalījumu sauc par cilku sadalījumu. Ja katrs apakšgrafs saimē F ir ceļš, tad sadalījumu sauc par ceļu sadalījumu. nevar sadalīt ciklos, bet var izveidot ceļu sadalījumu viena šķautne-triviāls ceļš.

Def. Pāra grafs - visām virsotnēm ir pāra pakāpe.

T. Grafa G šķautnes var sadalīt cikla sadalījumā $\Leftrightarrow$ grafs G ir pāra grafs

T. Eilera grafs $\Leftrightarrow$ pāra grafs $\Leftrightarrow$ cilku sadalījums.

9. Hamiltona grafs

Def. Hamiltona ceļš ir ceļš, kas satur visas grafa virsotnes. Hamiltona cikls ir cikls, kas satur katru grafa virsotni tieši vienu reizi.

Def. Ja grafs satur Hamiltona ciklu, tad tas ir Hamiltona grafs.

Hamiltona cikls var nebūt Eilera cikls un otrādi. Līdz šim nav atrasti pietiekamie un nepieciešamie nosacījumi tā eksistencei, Daudzi grafu teorijas jautājumi pierādāmi grafos ar Hamiltona ciklu. Hamiltona cikla noteikšanu var veikt ar pārlasi.

9.1. Hamiltona cikla noteikšana

Kā grafā konstruēt Hamiltona ciklu? Izmanto līdzīgu spriešanu kā nosakot izomorfismu vai planaritāti fokusējas, lai noteiktu, ka grafā neeksistē Hamiltona cikls lai pierādītu - konstruē grafā potenciālu Hamiltona ciklu, līdz nonāk līdz pretrunai.

Likumi, kam jāizpildās, konstruējot Hamiltona ciklu:

- Pieņem, ka grafā ir Hamiltona cikls. Šādam ciklam jābūt tieši 2 šķautnēm incidentām katrai virsotnei. No tā seko:
- Ja grafā ir virsotnes ar pakāpi 2, tad abas šķautnes pieder Hamiltona ciklam
- Neveido apakšciklu, tas ir, ciklu, kas nesatur visas virsotnes
- Tā kā katrai virsotnei var būt tikai 2 šķautnes, tad pārējās (tās, kas nav izmantotas, lai konstruētu Hamiltona ciklu) jānoņem

9.2. Kā grafā konstruēt Hamiltona ciklu?

Likumi, kam jāizpildās, konstruējot Hamiltona ciklu

- Ja grafā nav virsotne ar pakāpi 2, tad mēģinājumu ceļā meklē Hamiltona ciklu.
- Kā sākuma virsotni izvēlas tādu, lai pē iespējas vairāk šķautnes pazustu var būt jāaplūko vairāki gadījumi ņem vērā simetriju, ja var piemērot

9.3. Teorēmas

T. Dīraka teorēma. Ja grafā no katras virsotnes iziet $\geq \frac{n}{2}$ šķautnes tajā ir Hamiltona ceļš.

T. Grīnberga teorēma. (Var tikt izmantota, lai pierādītu, ka daži planāri grafi nesatur Hamiltona ciklu) Pieņem, ka planāram grafam G ir Hamiltona cikls. Grafu G uzzīmē jebkurā plakanā izkārtojumā. r_i - skaldņu skaits Hamiltona ciklā, ko ierobežo i šķautnes. r'_i - skaldņu skaits ārpus Hamiltona cikla, ko ierobežo i šķautnes.

Tad izpildās:

$$\sum_i (i-2)(r_i-r'_i) = 0 \tag{2}$$

9.4. Turnīrs

Def. Turnīrs ir orientēts grafs, kurā starp katrām 2 virsotnēm ir tieši viena šķautne. Pilns grafs ar n virsotnēm K_n katra šķautne orientēta vienā virzienā virsotnes - komandas, šķautne uv - v uzvar u katrā šādā grafā ir orientēts ceļš $(v_1 v_2)...(v_{n-1} v_n)$, kas ietver visas virsotnes.

T. Katrā turnīrā ir Hamiltona ceļš.

10. Koki

10.1. Definīcijas

Def. Koks ir sakarīgs grafs bez cilkiem.

Standarta veids, kā zīmēt koku ar sakni, ir novietojot sakni augšā un atbilstoši tālāk virsotnes kārtot pa līmeņiem Sakne ir 0. līmenis No saknes uz katru koka virsotni ved unikāls ceļš Virsotnes x līmenis kokā norāda unikālā ceļa garumu no saknes a. Ja grafs ir neorientēts, tad jebkura tā virsotne var būt sakne; orientētam grafam ir noteikta sakne.

Def. Lapa - virsotne kokā ar pakāpi 1. Mežš - grafs bez cikliem, ne obligāti sakarīgs (komponentes ir koki).

Def. Ja p ir i līmeņa virsotne, c ir $i + 1$ līmeņa virsotne un pc ir šķautne, tad c ir p bērns; p ir c vecāks. Ja ir virsotņu virkne $v_0 = p1, v_1, ..., v_k = d$, kur v_1 ir v_0 bērns, v_2 ir v_1 bērns, ..., v_k ir v_k-1 bērns, tad d ir p pēctecis.

Def. Šādi apgalvojumi ir ekvivalenti: (A) T ir koks; (B) T starp katrām 2 virsotnēm ir tieši viens ceļš; (C) T ir sakarīgs grafs, no kura izmetot jebkuru šķautni, T kļūst nesakarīgs.

Def. Lapas ir koka virsotnes ar pakāpi 1, tās ir virsotnes bez bērniem. Virsotnes ar bērniem ir iekšējās virsotnes. Ja katrai iekšējai virsotnei ir 2 bērni, tad tas ir binārs koks. Ja katrai iekšējai virsotnei ir m bērni, tad tas ir m-ārs koks.

Def. Koka (ar sakni) augstums ir garākais ceļš no saknes vai ekvivalenti jebkuras virsotnes augstākais/lielākais līmenis. Koks (ar sakni) ir līdzsvarā, ja visas lapas ir līmenī h un h-1. (Tie ir labi koki) M-āra koka līdzsvarošana minimizē tā augstumu.

10.2. Teorēmas

T. T ir koks $\Leftrightarrow$ T ir grafs bez cikliem, kurā pievienojot šķautni starp jebkurām divām nesavienotām virsotnēm rodas cikls.

S. Katrā kokā ir ≥ 2 lapas, $n \geq 2$.

T. Kokā ar n virsotnēm ir n-1 šķautne.

T. Mežā ar n virsotnēm un k komponentēm ir n-k šķautnes.

T. T ir koks ⇔ T ir sakarīgs grafs ar n virsotnēm un n-1 šķautnēm.

T. Ja T ir m-ārs koks ar n virsotnēm, no kurām i ir iekšējās virsotnes, tad $n = mi + 1$.

S. Ja T ir m-ārs koks ar n virsotnēm, i iekšējām virsotnēm un l lapām, tad zinot vienu no vērtībām (i, n vai l) abas pārējās attiecīgi var aprēķināt:

$$i : l = (m-1)i + 1, n = mi + 1 \tag{3}$$

$$l : i = (l-1)/(m-1), n = (ml-1)/(m-1) \tag{4}$$

$$n : i = (n-1)/m, l = [(m-1)n + 1]/m \tag{5}$$

T. Ja T ir m-ārs koks ar augstumu h un l lapām, tad: a) $l \leq mh$ un ja visas lapas ir augstumā h, $l = mh$; b) $h \geq \lceil \log nl \rceil$ un ja koks ir līdzsvarā, $h = \lceil \log nl \rceil$ $\lceil r \rceil$ noapaļo r uz nākošo lielāko veselo skaitli

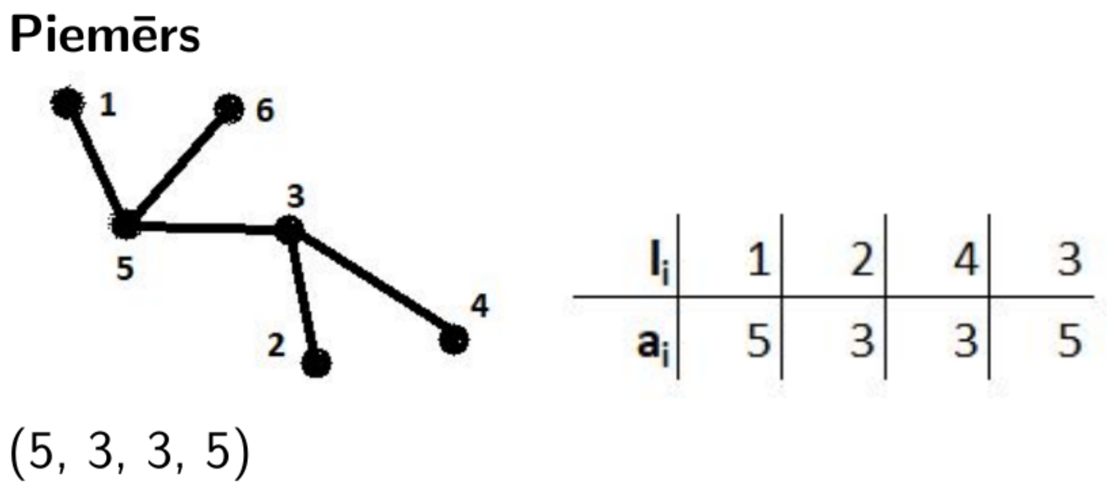
T. Kokā ar n virsotnēm ir n-1 šķautne.

S. Katrā kokā ir ≥ 2 lapas, $n \geq 2$.

10.3. Prūfera virkne

Koka virsotnes numurē. Katram kokam ar n numuriem izveido virkni $a_1, a_2, ..., a_{n-2}$ ar garumu n − 2 sekojoša veidā: izvēlas l_1 lapu ar mazāko numuru un ar a_1 apzīmē blakusvirsotni; izdzēš l_1 no koka un atkārto iepriekšējo punktu; apstājas, kad atlikušais koks ir reducēts uz 2 lapām, kas savienotas ar šķautni izveidojas virkne.

To sauc par Prufera kodu/virkni (Prüfer sequence). Tālāk var parādīt, ka katra šāda (n − 2) garuma virkne no n numuriem definē unikālu n numurētu koku: apvērš procedūru: var novērot, ka lapas (virsotnes ar pakāpi 1) nekad neparādās virknē pirmais numurs virknē ir kaimiņš lapai ar mazāko numuru mazākā numura lapa ir mazākais numurs, kas neparādās virknē, aplūkojot visu virsotņu numuru sarakstu atbilstoši ievieto tabulā li un tālāk vairs neaplūko atkārto procedūru palikušajām (n − 1) virsotnēm numurētajā kokā atbilstoši palikušajai (n − 3) virknei utt..



9. zīm.Piemērs

Apgalvojums Katra virsotne ar pakāpi ≥ 2 parādās virknē $a1, a2, ...a_{n-2}$.; Kopā tiek nodzēstas n-2 no n-1 šķautnēm; Tiek nodzēsta ≥ 1 šķautne, kas ietver virsotni v; kad tiek nodzēsta pirmā no šķautnēm, v nav lapa; u ir lapa, $li = u$, $ai = v$.

10.4. Pārklājošais koks

Koki piedāvā ietvaru problēmu risināšanā, kas ietver dažādu variantu secīgu aplūkošanu/izskatīšanu/izvēlēšanos. Daudzu no iepriekš aplūkotām problēmām (izomorfisms, Hamiltona cikls, hromatiskais skaitlis u.c.) datorizētā risināšana balstās uz kokiem.

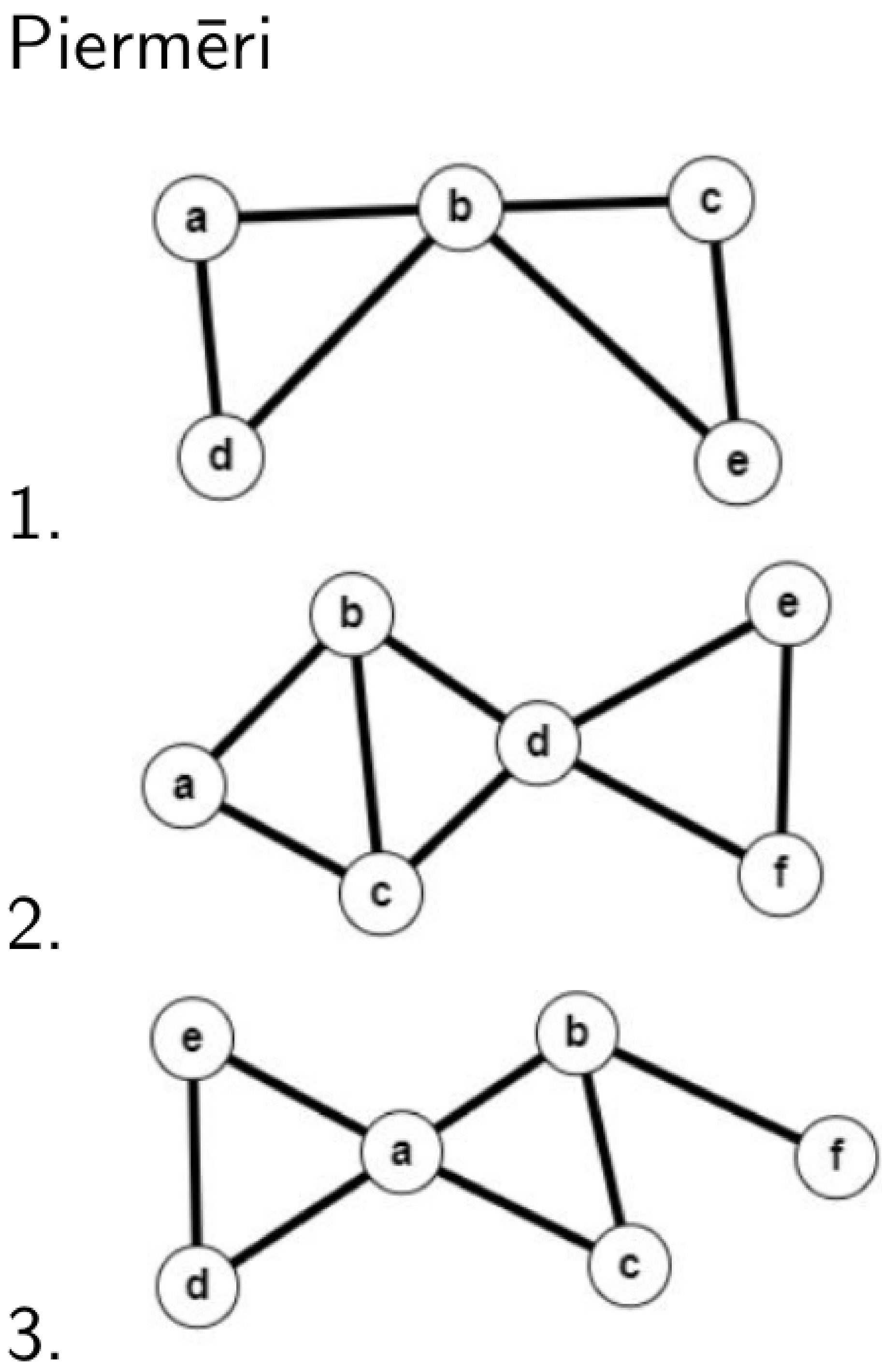
Def. Grafa G pārklājošais koks T (spanning tree) ir koks , kura visas šķautnes ir arī G šķautnes un tas ietver visas G virsotnes.

10.5. Dažāda veida meklēšanas algoritmi

10.5.1. Meklēšana dziļumā (Depth-first spanning tree DFS)

Algoritms

- Izvēlas kādu virsotni par sakni un sāk veidot ceļu no saknes, kas sastāv no grafa šķautnēm ‘
- katrā solī mēgina turpināt konstruēt koku no tekošās virsotnes
- ceļš tiek turpināts, kamēr to nevar turpināt neatkārtojot jau kokā esošu virsotni
- virsotne, kur šis ceļš apstājas, ir lapa
- ja nav iespējams turpināt ceļu, tad atkāpjas uz iepriekšējo virsotni, t.i., dodas atpakaļ uz vecāka virsotni y šai lapai un cenšas turpināt ceļu citā virzienā
- kad visi iespējamie ceļi no šī vecāka y un tā citiem bērniem ir izveidoti, tad kāpjas atpakaļ uz konkrētā vecāka y vecāka virsotni utt. līdz nonāk atpakaļ līdz saknei un ir pārbaudīti visi iespējamie ceļi no saknes



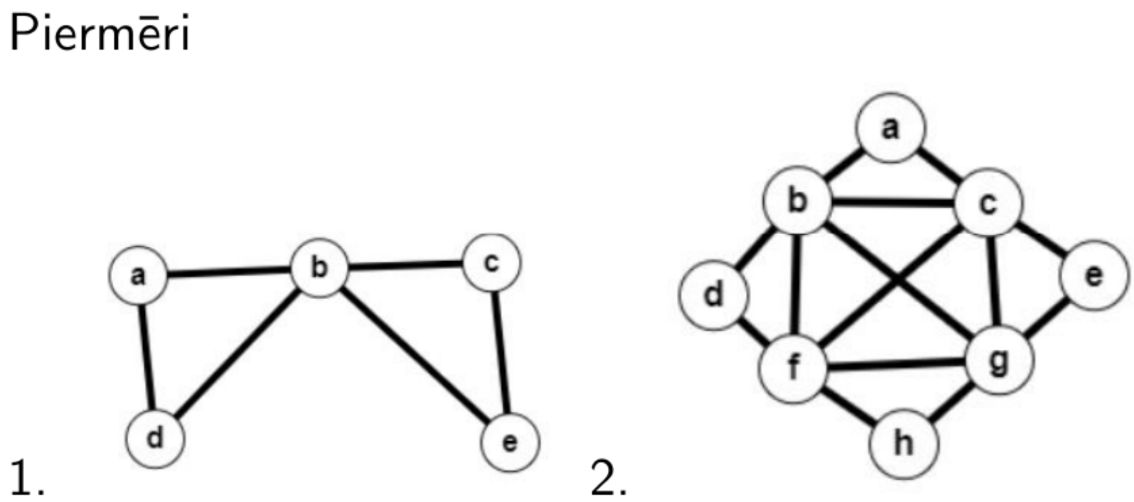
10. zīm.Piemērs

10.5.2. Meklēšana plašumā (Breadth-first spanning tree BFS)

Algoritms

- Izvēlas kādu virsotni kā sakni, kas ir 0-līmenis un novieto visas izejošās šķautnes 1-līmenī kokā.

- Sekojoši nākošajā līmenī pievieno šķautnes, kas ir incidentas saknes blakusvirsotnēm, ja vien nav savienota ar kokā jau izmantotu virsotni, utt.



11. zīm.Piemērs

10.6. Minimālais pārklājošais koks (Minimal Spanning Tree (MST))

10.6.1. Definīcija

Def. Minimālais pārklājošais koks (MST) vai minimālā svara aptverošais koks ir savienota, ar malām svērta nevirzīta grafika malu apakškopa, kas savieno visas virsotnes kopā, bez cikliem un ar minimālo iespējamo kopējo malas svaru.

Dots grafs, kurā šķautnēm ir garumi. Uzdevums atrast pārklājošo koku ar minimālo garumu (tiek summēts šķautņu garums).

Tā ir sarežgītāka problēma nekā atrast īsāko ceļu tīklā. Aplūkosim 2 algoritmus: Prima algoritmu un Kruskal algoritmu.

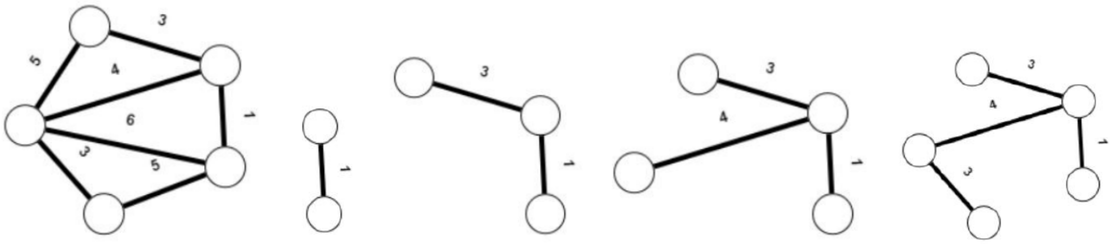
10.6.2. Prima algoritms

n - virsotņu skaits tīklā

Atkārtot sekojošo soli, kamēr kokā T ir n-1 šķautnes: pievieno T īsāko šķautni, starp virsotni, kas ir kokā T un virsotni, kas nav T (sākotnēji izvēlas jebkuru šķautni ar īsāko garumu).

Piemērs.

Īsākā šķautne ir 1. Nākošā īsākā šķautne, ko var savienot ar koku ir 3. Nākošā ir 4, kurai seko 3. Visas virsotnes ir iekļautas.

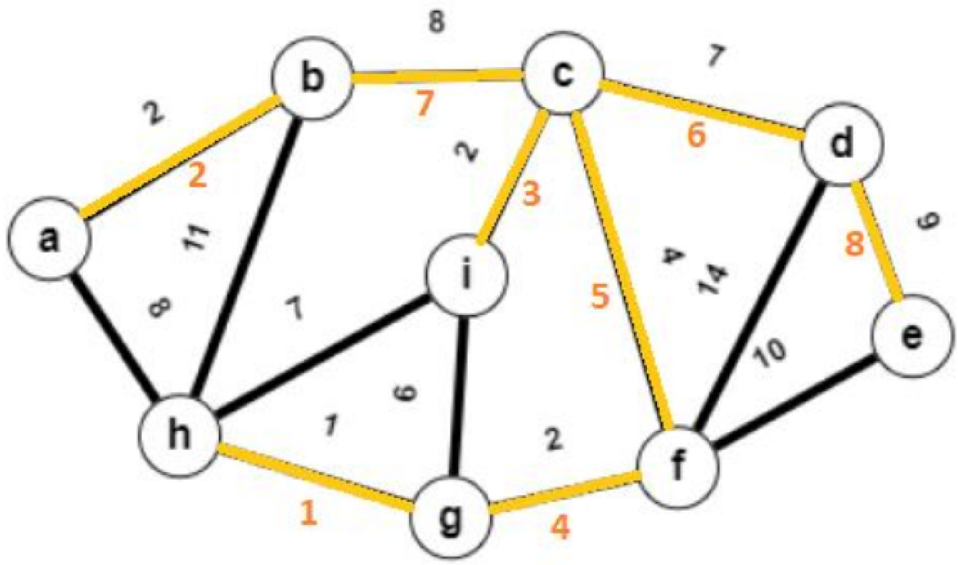


12. zīm.Piemērs

10.6.3. Kruskal algoritms

Atkārtot sekojošo soli, kamēr kopā T ir n-1 šķautnes (sākumā kopa T ir tukša): pievieno T īsāko šķautni, kas neveido cilku ar šķautnēm, kas jau atrodas kopā **T** (sākotnēji izvēlas jebkuru šķautni ar īsāko garumu).

Piemērs



Dzeltenie cipari - secība, kādā izvēlas šķautnes

13. zīm.Piemērs

T. Prima algoritms vienmēr atrod (vienu no) minimālo pārklājošo koku.

10.7. Sakarīgo grafu pārklājošie koki

Ja grafs nav sakarīgs, tad tam neeksistē pārklājošs koks.

Ja r ir sakne un u ir patvaļīga virsotne, tad ceļš ru, kas ir meklēšanas plašumā kokā T, ir īsākais ceļš no r uz u sākotnējā grafā.

Īsākais ceļš:

$ru_1, u_1u_2, \ldots, u_{k-1}u$

$u_1 - 1$ līmenis

u_2− ≤ 2 līmenis

$\ldots$

uk−$1 \leq k - 1$ līmenis

$u \leq k$ līmenis

10.8. Pārklājošā koka pielietojumi

Vai grafs ir sakarīgs? Sākot ar virsotni r (sakni), konstruē pārklājošo koku T (ar DFS vai BFS) ja virsotņu skaits T = virsotņu skaitu G, tad G ir sakarīgs (citādi nē).

Grafa G sadalīšana sakarīgās komponentēs. Kamēr visas virsotnes nav ieliktas kādā komponentē: r - virsotne, kas nav nevienā komponentē K veido koku ar BFS K' - komponente, kas sastāv no visām $v \in T$ un uv , u, $v \in T$.

Vai grafā ir cikls? Katrai komponentei K_i uzkonstruē pārklājošo koku Ti. Ja ir šķautne uv, u, $v \in K_i$, $uv \in /Ti$, tad ir cikls.

Vai grafs ir divdaļīgs? Konstruē pārklājošo koku T ar BFS. $r \in K$⇒ 1.līm. $\in L$ ⇒ 2.līm. $\in K$ ⇒ 3.līm.$\in L$... $K = r\cup$ (2.līm.) $\cup$ (4.līm) $\cup$ ... L = (1.līm.) $\cup$ (3.līm) $\cup$ ... Pārbauda, vai šķautnes uv,$u \in K, v \in L$

Neorientētu grafu var raksturot ar kaimiņmatricu, kas satur $|V| = n$ rindas un kolonnas. Katrai rindai un kolonnai atbilst viena virsotne. Rindu un kolonnu secība ir vienāda. Neorientētam grafam $a_i j = 1$, ja virsotnes K_i un K_j ir kaimiņvirsotnes, t.i., tās abas in incidentas šķautnei $e = [v_i, v_j]$, pretējā gadījumā ai $j = 0$. Sakarīguma pārbaude ar kaimiņmatricu.

11. Tīkls

11.1. Definīcijas

Optimizācijas problēmas ir aktuālas gan zinātnē, gan pielietojumos. Nozīmīgas tīklu optimizācijas problēmas: īsākā ceļa atrašana minimālais pārklājošais koks maksimālā plūsma.

Def. Tīkls ir grafs ar katrai šķautnei piekārtotu ne-negatīvu veselu skaitli. Skaitlis var reprezentēt garumu, caurlaidību, izmaksas u.c. turpmāk aplūkosim tīklus, kas ir neorientēti un sakarīgi.

11.2. Īsākā ceļa atrašana

Algoritms salīdzinoši vienkāršai problēmai:

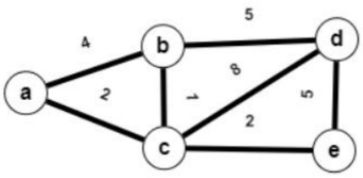
- Atrast īsāko ceļu tīklā no virsotnes u uz virsotni z. Var būt vairāk kā 1 īsākais ceļš
- Viens no veidiem īsākā ceļa atrašanai - izskatīt visus ceļus un noteikt īsāko. Taču tas nav optimāli, jau pie 100 virsotnēm pat skaitļojot datorā - problēmas.
- Ir algoritmi, kas atrisina šo problēmu efektīvi. Aplūkosim Daikstras algoritmu, kas tīklā atrod īsākos ceļus no dotas virsotnes a uz visām pārējām virsotnēm.

11.2.1. Daikstras algoritms

Ar $k(e)$ apzīmē šķautnes e garumu; ar m apzīmē attāluma skaitītāju; lai palielinātu m vērtību, algoritms apzīmē virsotnes, kuru minimālais attālums no virsotnes a ir m; pirmā vērtība virsotnei x būs iepriekšējā virsotne ar īsāko ceļu no a uz x; otrā vērtība virsotnei x būs īsākā ceļa garums no a uz x

Procedūra

- izvēlas virsotni a un apzīmē ar $(-, 0)$, kur '-' norāda tukšumu un 0 atbilst tam, ka nav šķautne starp virsotnēm
- pārbauda katru šķautni $e = (p, q)$ no kādas atzīmētas virsotnes p uz kādu neatzīmētu virsotni q. Pieņem, ka p apzīmējums ir $[r, d(p)]$. Ja $d(p) + k(e) = m$, virsotni q atzīmē ar (p, m)
- ja visas virsotnes nav vēl apzīmētas, palielina m par 1 un veic otro soli. Citādi iet uz soli 4. Ja ir interse tikai par īsāko ceļu uz z, tad dodas uz 4.soli, tiklīdz z ir atzīmēts
- katrai virsotnei y īsākais ceļš no a uz y ir garumā $d(y)$, otrā y vērtība. Šādu ceļu var atrast, izsekojot atpakaļ no y, izmantojot pirmo vērtību;



1 Izvēlas virsotni a. No virsotnes a iziet 2 šķautnes (m=1). Īsākais ceļš ir $a \to c = 2$

m=1	ceļš a->x
b	4
c	2

2 palielina m (attālumu no a), aplūkojot virsotnes, kas ir 2 šķautņu attālunā no a (pamatojas uz iepriekš atrasto īsāko ceļu)

m=2	ceļš a->x	ceļš a->c->x
b	4	3
d	-	10
e	-	4

Īsākais ceļš ir $a \to c \to b$

m=3	ceļš a->x	ceļš a->c->x	ceļš a->c->b->x
d	-	10	8
e	-	4	-

3 Īsākais ceļš ir $a \to c \to e$

m=3	ceļš a->x	ceļš a->c->x	ceļš a->c->b->x	ceļš a->c->e->x
d	-	10	8	9

4 Īsākais ceļš ir

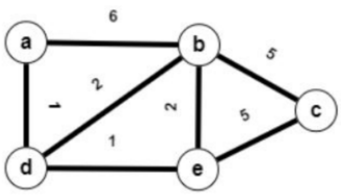
14. zīm.Piemērs

Procedūra (Cits pieraksta veids)

- Ceļa meklēšanu uzsāk no kādas izvēlētas virsotnes. Sākumā ceļa garums ir 0 un ceļa garums līdz pārējām virsotnēm nav zināms (pieņem, ka liels un apzīmē ar ∞).
- Aplūko visas sākuma virsotnes blakusvirsotnes un nosaka attālumu līdz tām. Atbilstoši aizstāj ∞ ar šo noteikto attālumu;

- izvēlas kā nākošo aplūkot virsotni, kurai ir mazākā vērtība un atkārto procedūru virsotnēm, kas nav jau kā mazāko izvēlēto virsotņu kopā. Ja ir īsāks attālums līdz jau apzīmetai virsotnei, tad nomaina uz mazāko, citādi atstāj to, kas atrasta iepriekš.

Cits pieraksta veids



S	a	b	c	d	e
-	0	∞	∞	∞	∞
a	0	6	∞	1	∞
ad	0	**3**	∞	1	2
ade	0	3	**7**	1	2
adeb	0	3	7	1	2

- ceļa meklēšanu uzsāk no kādas izvēlētas virsotnes. Sākumā ceļa garums ir 0 un ceļa garums līdz pārējām virsotnēm nav zināms (pieņem, ka liels un apzīmē ar ∞)
- aplūko visas sākuma virsotnes blakusvirsotnes un nosaka attālumu līdz tām. Atbilstoši aizstāj ∞ ar šo noteikto attālumu
- izvēlas kā nākošo aplūkot virsotni, kurai ir mazākā vērtība un atkārto procedūru virsotnēm, kas nav jau kā mazāko izvēlēto virsotņu kopā
- ja ir īsāks attālums līdz jau apzīmetai virsotnei, tad nomaina uz mazāko, citādi atstāj to, kas atrasta iepriekš

15. zīm.Piemērs

11.3. Plūsma tīklā

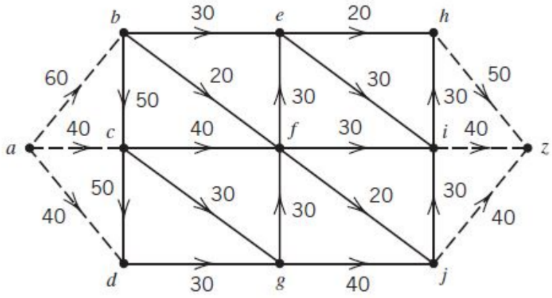
Def. Plūsma tīklā - orientēts grafs (tīkls), katrai šķautnei uv atbilst caurlaidība $c(uv)$ (piekārtots vesels ne-negatīvs skaitlis) sākuma un beigu virsotne (a un z).

Kādu daudzumu iespējams nosūtīt no a uz z, pa nevienu šķautni, nesūtot vairāk kā $c(uv)$? (interesē maksimalizēt plūsmu).

Def. $a - z$ plūsma $f(x)$ orientētā tīklā ir (vesela) skaitļa funkcija, kas definēta katrai šķautnei, kas kopā ar sākuma un beigu virsotni apmierina nosacījumus:

- katrai šķautnei $0 \leq f(uv) \leq c(uv)$, kur $f(uv)$ ir plūsma caur uv un $c(uv)$ ir uv caurlaidība
- ja $u \neq a$, $u \neq z$, tad $\sum_v f(uv) = \sum_v f(v_u)$ (ienākošās un izejošās plūsmas ir vienādas)
- $\sum_u f(au)$ = maksimālā iespējamā plūsma tīklā

Ja nav zināma sākuma un beigu virsotne, tad ir iespējas pārveidot tīklu atbilstoši, lai iegūtu vienu sākuma un vienu beigu virsotni.



11.3.1. Šķēlums

Def. Šķēlums - šķautņu kopa E, ka jebkurš ceļš (a,z) iet caur E.

Def. Šķēluma caurlaidība -

$$c(E) = \sum_{uv \in E} c(uv) \tag{6}$$

(plūsmas virziens no kopas, kas satur a uz kopu, kas satur z).

Šķēlums $(\{a, b, c\}, \{d, e, z\})$ sastāv no šķautnēm bd, be un K_e, bet ne no dc, jo vērsta pretējā virzienā.

T. Plūsma $a \to z$ nevar būt lielāka par (Forda-Folkensona algoritms/procedūra, z) - šķēluma caurlaidību.

11.3.2. Forda-Folkensona T.

T. (Max plūsma $A \to B$) = (Min šķēlums, kas atdala A un B) Apgalvojums: (Max ceļu skaits bez kopīgām virsotnēm) = (Max plūsma)

11.3.3. Forda-Folkensona teorēma

Kāds ir mazākais šķautņu skaits šķēlumā, kas atdala A no B?

Mengera **T.** Ja mazākajā šķēlumā, kas atdala A no B ir k šķautnes, tad var atrast k ceļus no A uz B, kuriem nav kopīgu šķautņu. (Min šķēlums) = (Max ceļu skaits no A uz B bez kopīgām virsotnēm). Lielākais iespējamais nekaimiņu virsotnes x un y savienojošu iekšēji nešķeļošos ceļu skaits vienāds ar mazākās x un y atdalošās kopas apjomu.

T. Jebkurai $a - z$ plūsmai tīklā plūsma no a ir vienāda ar plūsmu, kas nonāk z.

Def. Plūsmas vērtība $|f|$ ir vienāda ar plūsmu, kas iziet no sākuma virsotnes, jeb saskaņā ar teorēmu plūsmu, kas nonāk z.

T. Jebkurai $a - z$ plūsmai un $a - z$ šķēlumam tīklā $|f| \leq c(E)$.

Iepriekšējā teorēma saka, ka $a - z$ plūsmas vērtība nevar pārsniegt jebkura $a - z$ šķēluma caurlaidību. No tā seko, ka kāda $a - z$ plūsma $f*$ ir vienāda ar kādu $a - z$ šķēlumu. Tādā gadījumā $f*$ ir jābūt maksimālās plūsmas lielumam. Pārliecināsimies, ka jebkuram plūsmas tīklam var konstruēt $a - z$ plūsmu, kuras vērtība ir vienāda ar kādu $a - z$ šķēlumu un tā ir plūsma ar maksimālo vērtību.

Intuitīva pieeja, lai risinātu problēmu varētu būt plūsmas sadalīšana pa vienas vienības ceļiem. Sekojoši maksimālo plūsu varētu veidot skaitot kopā šādus $a - z$ vienas vienības ceļus, vienmēr pārliecinoties, ka nevienas šķautnes caurlaidība nav pārsniegta. Tādējādi plūsmas papildināšanai varētu izmantot tikai nepiesātinātas šķautnes, tas ir tādas, kur jau uzkonstruētā plūsma neaizņem visu caurlaidību.

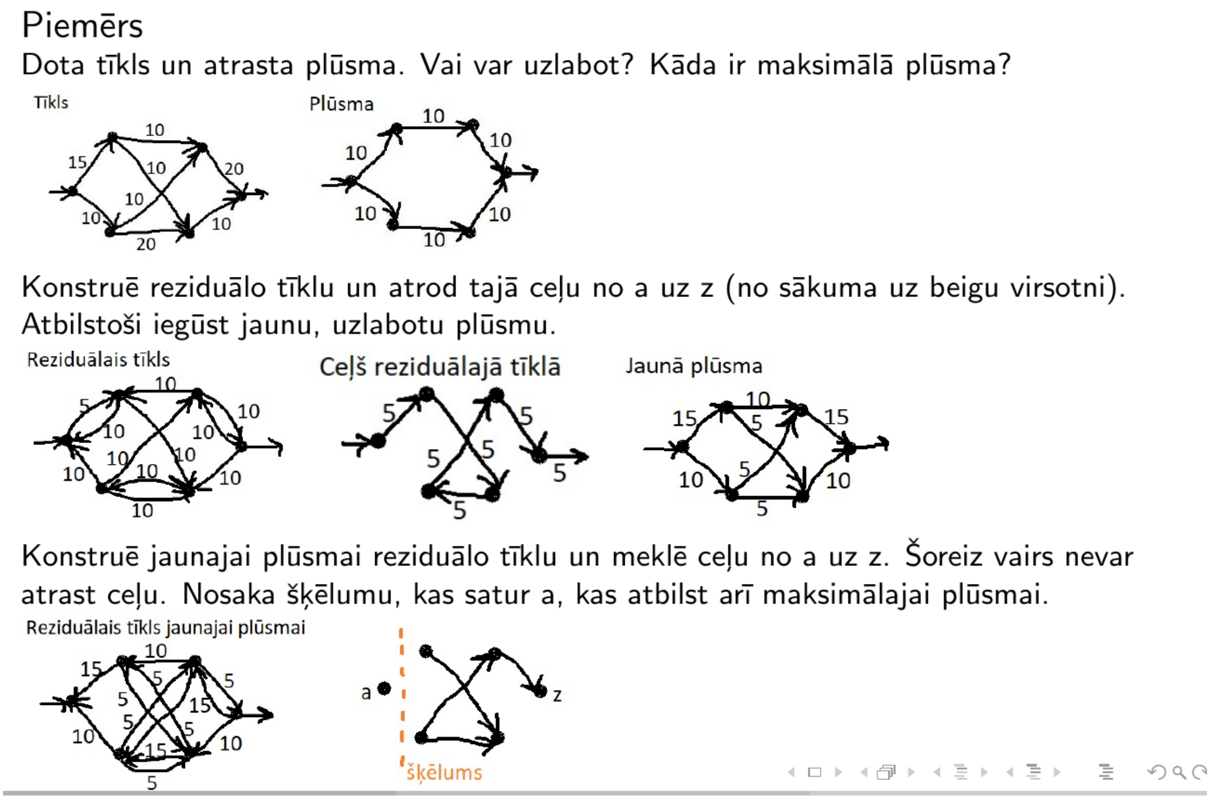
Kā izvairīties no šādām situācijām un vai būtu iespējams uzlabot plūsmas ceļu izvēles? Tas ir problemātiski piemērojot iepriekš aplūkoto pieeju, taču ir algoritms/procedūra, kas ļauj uzlabot plūsmu un palielināt plūsmu. Tīklu ar atrasto plūsmu 7 var pārzīmēt, lai uzskatāmāk redzams, ka caur šķautni bc plūsma virzās pretējāvirzienā ar vērtību 5 (no kopas, kas satur z un kopu, kas satur a). Par cik vienībām plūsma šķautnē be varētu tapt samazināta?

11.3.4. Forda-Folkensona algoritms/procedūra

Def. $c'(uv)$ - šķautnes uv caurlaidība reziduālajā tīklā

- Ja nav plūsma $u \to v$ vai $v \to u$: $c'(uv) = c(uv)$; Ja ir plūsma $u \to v$: $c'(uv) = c(uv) - f(uv)$
- Ja ir plūsma $v \to u$: $c'(uv) = c(uv) + f(uv)$
- Ja ir plūsma $f(uv)$ un tiek pievienota f'.

Var pārsūtīt plūsmu $f' \leq f$ pretējā virzienā samazinot plūsmu $u \to v$ pa f'.



16. zīm.Forda-Folkensona procedūras piemērs

12. Sapārojumi grafos

12.1. Definīcijas

Def. Sapārojums ir šķautņu kopa, kur katra virsotne pieder ≤ 1 no šķautnēm. Citiem vārdiem, grafa G sapārojums M (matching) ir G apakšgrafs, kurā nevienai šķautnei nav kopīga virsotne ne ar vienu citu šķautni. Katrai virsotnei sapārojumā ir virsotne 1, tas ir, katra virsotne ir galavirsotne vienai šķautnei. Šķautņu skaits sapārojumā raksturo sapārojuma izmēru $|M|$.

Def. Maksimāls sapārojums M grafā G ir tad, ja vairs nav iespējams pievienot sapārojumam M papildus citas šķautnes no G.

Def. Sapārojuma maksimums ir tad, ja sapārojumam ir lielākais iespējamais izmērs grafā. Tas ir lielākais maksimālais sapārojums. Vienam grafam var būt vairāki maksimumi.

Def. Neatkarīgas šķautnes - tām nav kopīgas virsotnes.

Def. Neatkarīga kopa - satur virsotnes starp kurām nav šķautnes.

Def. Divdaļīgs grafs - virsotnes var sadalīt divās neatkarīgās kopās.

Def. Divdaļīgs grafs ir līdzsvarā, ja tam abās kopās ir vienāds skaits virsotņu.

12.2. Holla teorēma

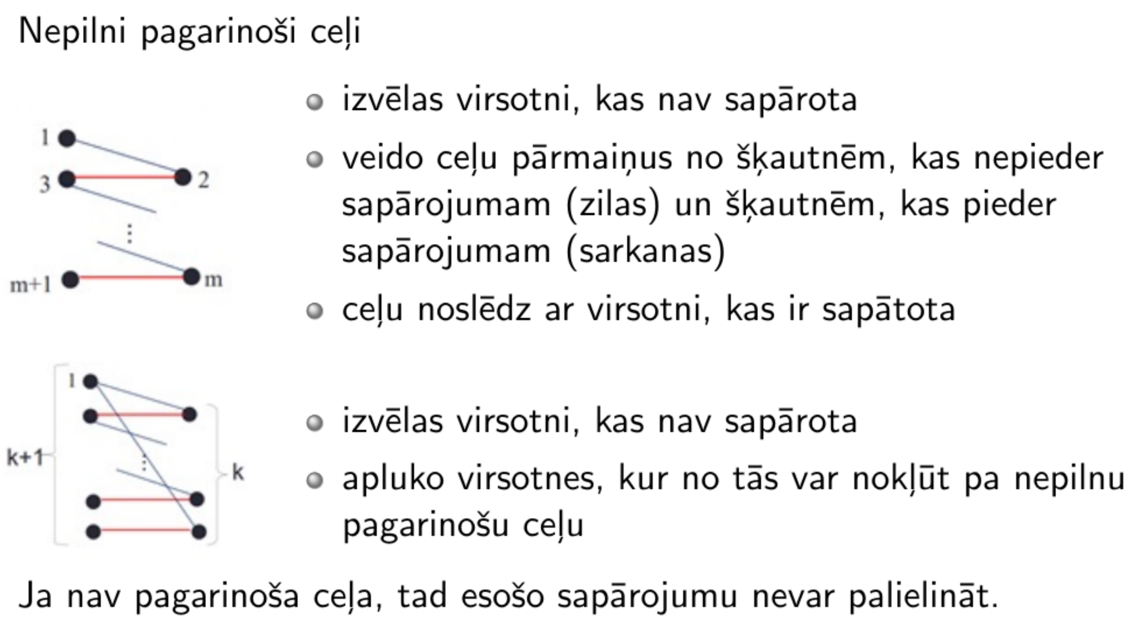
Interesē jautājums - kā atrast sapārojuma maksimumu? It īpaši interesē, kā atrast pilnu sapārojumu. Pilns (perfect) sapārojums ietver visas grafa virsotnes un katra virsotne pieder tieši vienai šķautnei. Katrs pilns sapārojums ir arī maksimāls sapārojums, jo nav iespēja papildus pievienot jaunas šķautnes. Taču maksimuma sapārojums var arī nebūt pilns. Ja grafam ir pilns sapārojums, tad tam ir pāra skaits virsotņu.

Holla **T.** Ja pilna sapārojuma nav ⇒ var atrast k virsotnes kreisajā pusē, kas savienotas ar $\leq k{-}1$ virsotnēm labajā pusē. Ja ir šāda situācija, tad sapārojuma nav.

12.3. Pagarinošie ceļi

Kā konstruēt sapārojumu? Grafā izvēlas šķautnes saskaņā ar sapārojuma definīciju. Kā uzlabot šo sapārojumu?

Nepilni pagarinoši ceļi: izvēlas virsotni, kas nav sapārota; veido ceļu pārmaiņus no šķautnēm, kas nepieder sapārojumam (zilas) un šķautnēm, kas pieder sapārojumam (sarkanas); ceļu noslēdz ar virsotni, kas ir sapātota, izvēlas virsotni, kas nav sapārota; apluko virsotnes, kur no tās var nokļūt pa nepilnu pagarinošu ceļu; ja nav pagarinoša ceļa, tad esošo sapārojumu nevar palielināt.



17. zīm.Pagarinošo ceļu konstruēšana

T. Ja sapārojums M nav maksimāls, tad esksitē pagarinošs ceļš.

12.4. Plūsmas tīklā metode

Plūsmas grafa izmantošana, lai risinātu divdaļīga grafa sapārošanas uzdevumu.

- Dotam divdaļu grafam $G = (A \cup B, E)$ orientē šķautnes no A uz B,
- pievieno jaunas virsotnes: sākuma virsotni a un beigu virsotni z.
- Pievieno papildus orientētas šķautnes no a uz katru virsotni A.
- Pievieno papildus orientētas šķautnes no virsotnēm B uz beigu virsotni z.
- Visas šķautņu caurlaides spējas pieņem kā 1.
- Atrisina maksimālās plūsmas problēmu jaunizveidotajā grafā G'.
- Šķautnes, kas tiek izmantotas maksimālās plūsmas tīklā atbilsts maksimuma sapārojumam.

Priekšmetu rādītājs