

A számítástudomány alapjai

ÖSSZEFOGLALÓ JEGYZET

Készítette: Illyés Dávid

Ez a jegyzet nagyon hasonlóan van struktúrálva az előadás jegyzetekhez és fő célja, hogy olyan módon adja át a "A Programozás Alapjai 1" nevű tárgy anyagát, hogy az teljesen kezdők számára is könnyen megérthető és megtanulható legyen.

Tartalomjegyzék

Oldal

| | | |
|----------|---|-----------|
| 1 | A gráfelmélet alapjai | 2 |
| 1.1 | Mi a gráf? | 2 |
| 1.2 | Multigráfok és irányított gráfok | 2 |
| 1.3 | Handshaking lemma | 2 |
| 1.4 | Komplementer és izomorfia | 3 |
| 1.5 | Gráfoperációk | 3 |
| 1.6 | Háromféle elérhetőség, összefüggőség | 3 |
| 1.7 | Gráfok összefüggősége a gyakorlatban | 4 |
| 1.8 | Fák és erdők | 4 |
| 1.9 | Fák további tulajdonságai | 5 |
| 1.10 | Feszítőfák | 5 |
| 2 | Minimális költségű feszítőfák | 7 |
| 2.1 | Alapkörrendszer, alap vágás rendszer | 7 |
| 2.2 | Minimális költségű feszítőfa | 7 |
| 2.3 | Minimális költségű feszítőfák struktúrája | 7 |
| 2.4 | Az ötödik elem | 8 |
| 3 | Gráfbejárások és legrövidebb utak | 9 |
| 3.1 | Általános gráfbejárás & BFS | 9 |
| 3.2 | A BFS tulajdonságai | 9 |
| 3.3 | Legrövidebb utak | 10 |
| 3.4 | Az elméleti javítás | 10 |
| 3.5 | Dijkstra, egy példán | 11 |
| 3.6 | Dijkstra helyessége | 11 |
| 4 | Legrövidebb utak, DFS, PERT | 13 |
| 4.1 | Legrövidebb utak konzervatív hosszfüggvény esetén 1 | 13 |
| 4.2 | Legrövidebb utak konzervatív hosszfüggvény esetén 2 | 13 |
| 4.3 | Depth First Search (DFS) | 14 |
| 4.4 | Direct Acyclic Graphs | 14 |
| 4.5 | Leghosszabb út keresése | 15 |
| 4.6 | A PERT probléma | 15 |
| 5 | Euler-séták és Hamilton-körök | 17 |
| 5.1 | Euler-séták | 17 |
| 5.2 | Hamilton-körök és -utak | 18 |
| 5.3 | A Chvátal-lezárt | 20 |
| 6 | Síkgráfok | 21 |
| 6.1 | Síkbarajzolhatóság | 21 |
| 6.2 | Az Ezler-féle poliéderformula, síkgráfok karakterizációja | 21 |
| 6.3 | Síkgráfok duálisa | 22 |
| 6.4 | Whitney | 23 |
| 7 | Lineáris egyenletrendszerek | 24 |

1 A gráfelmélet alapjai

1.1 Mi a gráf?

Def: $G = (V, E)$ egyszerű, irányítatlan gráf

Példa: Ha $V \neq \emptyset$ és $E \subseteq \binom{V}{2}$, ahol $\binom{V}{2} = \{\{u, v\} : u, v \in V, u \neq v\}$. V a G csúcsainak (vagy (szög)pontjainak), E pedig G éleinek halmaza.

Példa: $G = (\{a, b, c, d\}, \{a, b\}, \{a, c\}, \{b, c\}, \{b, d\})$

Def: A $G = (V, E)$ gráf diagramja a G egy olyan lerajzolása, amiben V -nek a sík különböző pontjai felelnek meg, és G minden $\{u, v\}$ élének egy u -t és v -t összekötő görbe felel meg.

Terminológia & konvenciók: Gráf alatt rendszerint egyszerű, irányítatlan gráfot értünk. Ha G egy gráf, akkor $V(G)$ a G csúcshalmazát, $E(G)$ pedig G élhalmazát jelöli, azaz $G = (V(G), E(G))$. Az $e = \{u, v\}$ élt röviden uv -vel jelöljük.

Ekkor e az u és v csúcsokat köti össze. Továbbá u és v az e végpontjai, amelyek az e élre illeszkednek, és e mentén szomszédosak.

1.2 Multigráfok és irányított gráfok

Megj: Ha egy gráf nem egyszerű, akkor lehetnek párhuzamos élei, hurokélei vagy akár párhuzamos hurokélei is.

Def: Az irányított gráf olyan gráf, aminek minden éle irányított.

Def: $G = (V, E)$ véges gráf, ha V és E is véges halmazok.

Def: Az n -pontú út, n -pontú kör, ill. n -pontú teljes gráf jele rendre P_n , C_n , ill. K_n . (P_1, P_2, P_3 elfajulók.) **Megf:** $K_1 = P_1, P_2 = C_2, C_3 = K_3$

Def: $c \in V(G)$ esetén a v -re illeszkedő élek száma a v fokszáma. Jelölése $d_g(v)$ vagy $d(v)$, a hurokél kétszer számít. (Irányított gráf esetén $\delta(v)$ (Delta) ill. $\rho(v)$ (Rho) a v ki- ill. befokát jelöli.)

Def: A G gráf maximális ill. minimális fokszáma $\Delta(G)$ ill. $\delta(G)$. G reguláris, ha minden csúcsának foka ugyanannyi: $\Delta(G) = \delta(G)$, G pedig k -reguláris, ha minden csúcsának pontosan k a fokszáma.

Megf: Minden kör 2-reguláris, K_n pedig $(n - 1)$ -reguláris.

1.3 Handshaking lemma

Kézfogás-lemma (KFL): Ha $G = (V, E)$ véges, nem feltétlenül egyszerű gráf, akkor $\sum_{v \in V} d(v) = 2|E|$, azaz a csúcsok fokszámösszege az élszám kétszerese.

Általánosított kézfogás-lemma: Tetsz. $G = (V, E)$ véges irányított gráfra $\sum_{v \in V} \delta(v) = \sum_{v \in V} \rho(v) = |E|$, azaz a csúcsok ki- és befokainak összege is az élszámot adja meg.

Biz: Az egyes csúcsokból kilépő éleket megszámlálva G minden irányított élét pontosan egyszer számoljuk meg. Ezért a kifokok összege az élszám. A belépő éleket leszámolva hasonló igaz, ezért a befokok összege is az élszám. \square

A KFL bizonyítása: Készítsük a G' digráfot úgy, hogy G minden élét egy oda-vissza irányított élpárral helyettesítsük. Ekkor

$$\sum_{v \in V} d_G(V) = \sum_{v \in V} \delta_{G'}(v) = |E(G')| = 2|E(G)| \quad \square$$

Megj: Úgy is bizonyíthattuk volna az általánosított kézfogás-lemmát, hogy egyenként húzzuk be G -be az éleket. 0-elű (üres)gráfokra a lemma triviális, és minden egyes él behúzása pontosan 1-gyel növeli az élszámot is és a ki/befokok összegét is.

Megj: Úgy is bizonyíthatunk volna a kézfogás-lemmát, hogy egyenként húzzuk be G -be az éleket. Üresgráfokra a lemma triviális, és minden egyes él behúzása pontosan 2-vel növeli a kétszeres élszámot és a csúcsok fokszámösszeget is.

1.4 Komplementer és izomorfia

Def: A G egyszerű gráf **komplementere** $\overline{G} = (V, (G), \binom{V}{2} \setminus E(G))$.

Megj: G és \overline{G} csúcsai megegyeznek, és két csúcs pontosan akkor szomszédos \overline{G} -ben, ha nem szomszédosak G -ben.

Példa:

Megf: Ha $G = (V, E)$ egyszerű gráf és a $|V(G)| = n$, akkor $d_G(v) + d_{\overline{G}}(v) = n - 1$ teljesül G bármely v csúcsra.

Biz: A K_n teljes gráf minden éle a G és \overline{G} gráfok közül pontosan az egyikhez tartozik. Ezért $d_G(v) + d_{\overline{G}}(v)$ megegyezik a v csúcs K_n -beli fokszámával, ami $n - 1$. \square

Def: A G és G' gráfok akkor **izomorfak**, ha mindekét gráf csúcsai úgy számozhatók meg az 1-től n -ig terjedő egész számokkal (alkalmas n esetén), hogy G bármely két u, v csúcsa között pontosan annyi él fut G -ben, mint az u -nak és v -nek megfelelő sorszámú csúcsok között G' -ben. Jelölése: $G \cong G'$.

Példa:

Megf: Ha $G \cong G'$, akkor G és G' lényegében ugyanúgy néznek ki. Így például minden fokszám ugyanannyiszor lép fel G -ben mint G' -ben, ugyan annyi C_4 kör található G -ben, mint G' -ben, stb.

1.5 Gráfoperációk

Def: **Éltörlés, csúcs törlés, él hozzáadás.**

Def: **Feszítő részgráf:** éltörlésekkel kapható gráf.

Feszített részgráf: csúcs törlésekkel kapható gráf.

Részgráf: él- és csúcs törlésekkel kapható gráf.

Példa: H_1, H_2, H_3 : a G **feszítő, feszített, jelzőnélküli** részgráfjai.

Megf: H a G részgráfja $\iff V(H) \subseteq V(G)$ és $E(H) \subseteq E(G)$.

H a G feszítő részgráfja $\iff V(H) = V(G)$ és $E(H) \subseteq E(G)$.

H a G feszített részgráfja $\iff V(H) \subseteq V(G)$ és $E(H) = E(G) \cap E(H)$.

Megj: A gráf definíciója megengedi, hogy a gráf egyik részéből egyáltalán ne vezessen el a gráf maradék részébe, azaz a gráf egyik csúcsából ne lehessen eleken keresztül eljutni a gráf egy másik csúcsába. Ez történik pl. az üresgráf (alias $\overline{K_n}$) esetén.

1.6 Háromféle elérhetőség, összefüggőség

Def: Legyen $G = (V, E)$ (irányított vagy irányítatlan) gráf.

Élsorozat: $(v_1, e_1, v_2, e_2, \dots, v_k, e_k, v_{k+1})$, ahol $e_i = v_i v_{i+1} \forall i$. (Tkp egyik csúcsból eljutunk egy másik csúcsba mindig él mentén haladva.)

Séta: olyan élsorozat, amelyben nincsen ismétlődő él.

Út: olyan séta, amelyben nincs ismétlődő csúcs.

Terminológia: Ha a kezdőpont u , a végpont v , akkor **uv -élsorozatról, uv -sétáról, ill. uv -útról** beszélünk. Ha hangsúlyozni szeretnénk, hogy $u = v$, de a kezdő (és vég)pontot nem akarjuk megnevezni, akkor **zárt élsorozatról, körsétáról** ill. **körrel** beszélünk.

Megf: G -ben $\exists uv$ -út $\implies G$ -ben $\exists uv$ -séta $\implies G$ -ben $\exists uv$ -élsorozat \square

Állítás: G -ben $\exists uv$ -élsorozat $\implies G$ -ben $\exists uv$ -út \square

Def: G irányítatlan gráf u -ból v **elérhető** ($u \sim v$), ha $\exists uv$ -út G -ben.

Def: A G irányítatlan gráf **összefüggő**, ha $u \sim v \forall u, v \in V(G)$.

Megj: (1) Az összefüggőség szokásos definíciója nem a \sim reláció segítségével történik, hanem valahogy így: a G irányítatlan gráfot akkor mondjuk összefüggőnek, ha G bármely két csúcsa között vezet út G -ben.

Megj: (2) Az előző definíciót irányított gráfokra is kiterjeszthetők: a G irányított gráfot akkor mondjuk **erősen összefüggő**nek, ha G bármely $u, v \in V(G)$ esetén van **irányított** uv -út G -ben.

Megj: (3) Irányított gráf másfajta összefüggősége is értelmezhető: a G irányított gráfot akkor mondjuk **gyengén összefüggő**nek, ha a G -nek megfelelő irányítatlan gráf összefüggő.

Köv: Ha G irányítatlan gráf, akkor \sim ekvivalenciareláció:

(1) $\forall u \in V(G) : u \sim u$, (2) $\forall u, v \in V(G) : u \sim v \Rightarrow v \sim u$, és (3) $\forall u, v, w \in V(G) : u \sim v \sim w \Rightarrow u \sim w$. \square

Def: A G gráf **(összefüggő) komponense** a \sim ekvivalenciaosztálya. Az egyelemű komponens neve **izolált pont**.

1.7 Gráfok összefüggősége a gyakorlatban

Lemma: (1) $K \subseteq V(G)$ pontosan akkor komponense G -nek, ha K -ból nem lép ki éle G -nek, de $\forall v, v' \in K$ esetén $v \sim v'$.

(2) Minden G irányítatlan gráf csúcshalmaza egyértelműen bomlik fel G komponenseinek diszjunkt uniójára. \square

Megj: A G komponense alatt sokszor nem csupán a G csúcsainak egy K részalmazát, hanem a K által feszített részgráfot értjük.

Megf: G pontosan akkor összefüggő, ha egy komponense van. \square

Élhozzáadási lemma (ÉHL): Legyen G irányítatlan gráf és $G' = G + e$. Ekkor az alábbi két esetből pontosan egy valósul meg.

(1) G és G' komponensei megegyeznek, de G' -nek több köre van, mint G -nek.

(2) G és G' körei megegyeznek, de G' -nek eggyel kevesebb komponense van, mint G -nek.

1.8 Fák és erdők

Def: A körmentes irányítatlan gráfot **erdőnek** nevezzük. Az összefüggő, körmentes irányítatlan gráf neve **fa**.

Megf: G erdő $\iff G$ minden komponense fa.

Példa:

Megf: (1) P_n fa minden $n \geq 1$ egész esetén. (2) Fához egy új csúcsot egy éllel bekötve fát kapunk:

Lemma: G n -csúcsú, k -komponensű erdő $\Rightarrow |E(G)| = n - k$.

Biz: Építsük fel G -t a $\overline{K_n}$ üresgráfból az élek egyenkénti behúzásával. G körmentes, ezért az ÉHL miatt minden lép zöld: behúzásakor 1-gyel csökken a komponensek száma. A $\overline{K_n}$ üresgráfnak n komponense van, G -nek pedig k . Ezért pontosan $n - k$ zöld élt kellett behúzni G felépítéséhez. \square

Köv: Ha F egy n -csúcsú fa, akkor élszáma $|E(F)| = n - 1$.

Biz: F egy 1-komponensű erdő, így az előző Lemma alkalmazható $k = 1$ helyettesítéssel.

Állítás: Tetsz. n -csúcsú G gráf esetén az alábbi három tulajdonság közül bármely kettőből következik a harmadik. (a) G körmentes. (b) G összefüggő. (c) $|E(G)| = n - 1$.

Biz: (a) + (b) \Rightarrow (c) : \checkmark

(a) + (c) \Rightarrow (b): Építsük fel G -t élek egyenkénti behúzásával. $n - 1$ él egyikének behúzása se hoz létre kört, ezért az ÉHL miatt minden lép zöld, és 1-gyel csökkenti a komponensszámot. Végül $n - (n - 1) = 1$ komponens marad, tehát G összefüggő.

$(b) + (c) \Rightarrow (a)$: Építsük fel G -t élek egyenkénti behúzásával. Mivel a komponensek száma végül 1 lesz, ezért $n - 1$ zöld élt kellett behúzni. (c) miatt G összes éle zöld, piros éle nincs. Az ÉHL miatt G körmentes. \square

1.9 Fák további tulajdonságai

Állítás: Legyen F egy tetszőleges fa n csúcson. Ekkor

- (1) $(F - e)$ -nek pontosan két komponense van $\forall e \in E(F)$ -re.
- (2) F -nek pontosan egy uv -útja van $\forall u, v \in V(F)$ -re.
- (3) $(F + e)$ -nek pontosan egy köre van $\forall e \notin E(F)$ -re.
- (4) Ha $n \geq 2$, akkor F -nek legalább két levele van.

Def: A G irányítatlan gráf v csúcsa **levél**, ha $d(v) = 1$.

Biz: (1): $F - e$ erdő, hisz körmentes. $F = (F - e) + e$, és mivel F is körmentes, e zöld az ÉHL miatt. Ezért F -nek 1-gyel kevesebb komponense van, mint $(F - e)$ -nek. Mivel F -nek 1 komponense van, $(F - e)$ -nek 2. \square

Biz: (2): F összefüggő, ezért van (legalább egy) uv -útja, mnodjuk P . Ezen P út bármely e élét elhagyva, a kapott $F - e$ grágnak (1) miatt két komponense van, melyek közül az egyik u -t, a másik v -t tartalmazza. Ezért $(F - e)$ -ben nincs uv -út. Azt kaptuk, hogy P minden éle benne van F minden uv -útjában, ezért F -ben P -n kívül nincs más uv -út. \square

Biz: (3): Tfh $e = uv$. Minden F körmentes, ezért $F + e$ minden köre e -ből és F egy uv -útjából tevődik össze. Ezért $F + e$ köreinek száma megegyezik az F fa uv -útjainka számával, ami (2) miatt pontosan 1. \square

Biz: (4): (Algebrai út) A KFL miatt $\sum_{v \in V(G)} (d(v) - 2) = \sum_{v \in V(G)} d(v) - 2n = 2(n - 1) - 2n = -2$. F minden v csúcsára $d(v) \geq 1$ teljesül, ezért $d(v) - 2 \geq -1$. A fenti összeg csak úgy lehet -2 , ha F -nek legalább 2 levele van. \square

Biz: (4): (Kombinatorikus út) Induljunk el F egy tetszőleges v csúcsából egy sétán, és haladjunk, amíg tununk. Ha sosem akadunk el, akkor előbb-utóbb ismétlődik egy csúcs, és kört találunk. Ezért elakadunk, és az csakis egy v -től különböző u levélben történhet. Ha $d(v) = 1$, akkor v egy u -tól különböző levél. Ha $d(v) \geq 2$, akkor sétát indulhatjuk v -ből egy másik él mentén. Ekkor egy u -tól különböző levélben akadunk el. \square

1.10 Feszítőfák

Építsük fel a G gráfot az élek egymás utái behúzásával, és az ÉHL szerinti kiszínezésével! Legyen G' a G gráf piros élei törlésével keletkező feszítő részgráf! G' biztosan körmentes lesz, hiszen a zöld élek sosem alkottak kört a korábbi élekkel. G' minden K' komponense részhalmaza G egy K komponensének. Ha $K' \neq K$, akkor G -nek van olyan éle, ami kilép K' -ből. Ezen élek mind pirosak K' definíciója miatt. Legyen e ezek közül az elsőnek kiszínezett. Az e él nem tudott kört alkotni a korábbn kiszínezettekkel, így nem lehet piros: ellentmondás. Ezek szerint G egy G' komponensei megegyeznek.

Köv: A G gráf zöld élei olyan G' feszítő részgráfot alkotnak, ami erdő, és komponensei megegyeznek G komponenseivel. \square

Def: F a G gráf **feszítőfája** (**ffája**), ha F egy G -ből éltörlésekkel kapható fa.

Állítás: $(G$ -nek van feszítőfája) $\iff (G$ összefüggő)

Biz: \Rightarrow : Legyen F a G feszítőfája. F összefüggő, és $V(F) = V(G)$, tehát G bármely két csúcsa között vezet F -beli út.

\Leftarrow : Építsük fel G -t az álek egyenkénti behúzásával és kiszínezésével. Láttuk, hogy a zöld élek egy F erdőt alkotnak, aminek egyetlen komponense van, hiszen G is egykomponensű. Ezek szerint F olyan fa, ami G -ből éltörlésekkel kapható. \square

Megj: Ha egy nem feltétlenül összefüggő G gráf éleit a fenti módon kiszínezzük, akkor a zöld élek G minden komponensének egy F feszítőfáját alkotják. Nem összefüggő G esetén a zöld élek alkotta feszítő részgráf neve a G **feszítő erdeje**.

2 Minimális költségű feszítőfák

2.1 Alapkörrendszer, alap vágás rendszer

Adott egy G gráf és G -nek egy F rögzített feszítőfája. Ekkor G minden éléhez F meghatározza G éleinek egy fontos részhalmazát. Attól függően, hogy az adott él F -hez tartozik-e vagy sem, különböző fajta részhalmazról van szó.

Def: A G gráf F feszítőfájának f éléhez tartozó **alap vágást** G azon élei alkotják, amik az $F - f$ két komponense között futnak. Az $e \in E(G) \setminus E(F)$ éléhez tartozó **alapkör** pedig az $F + e$ köre.

Megf: Tfh $f \in F$ és $e \in E(G) \setminus E(F)$. Ekkor $(F - f + e \text{ ffa}) \iff (f \text{ benne van } e \text{ alapkörében}) \iff (e \text{ benne van } f \text{ alap vágásában}).$

Köv: Az $e \in E(G) \setminus E(F)$ alapkörét e mellett azon F -beli élek alkotják, amelyek alapvágása e -t tartalmazza. Az $f \in F$ alapvágást f mellett a G azon élei alkotják, amelyek alapköre f -t tartalmazza.

2.2 Minimális költségű feszítőfa

Def: Adott a $G = (V, E)$ irányítatlan gráf élein a $k : E \rightarrow \mathbb{R}_+$ költségfüggvény. Az $F \subseteq E$ élhalmaz **költsége** az F -beli élek összköltsége: $k(F) = \sum_{f \in F} k(f)$.

Az $F \subseteq E$ élhalmaz G -ben **minimális költségű feszítőfa (mkffa)**, ha

- (1) (V, F) a G feszítőfája, és
- (2) $k(F) \leq k(F')$ teljesül a G bármely (V, F') feszítőfájára.

Az $F \subseteq E$ élhalmaz G -ben **minimális költségű feszítő erdeje**, ha

- (1) (V, F) a G feszítő erdeje, és
- (2) $k(F) \leq k(F')$ teljesül a G bármely (V, F') feszítő erdejére.

Cél: Hatékony eljárás mkffa keresésére.

Ötlet: Keressük a feszítőfát a tanult módon, az élek egyenkénti behúzásával, az ÉHL szerint zöldre színezett élek halmazaként.

Zöld él: olyan él, ami nem alkot kört a korábban felépített élekkel.

Mohó stratégia: A feszítőfa építéskor költség szerint növekvő sorrendben döntünk az élekről, hátha mkffát kapunk a végén.

Kruskal-algoritmus: Input: $G = (V, E)$ és $k : E \rightarrow \mathbb{R}_+$ költségfüggvény. Output: $F \subseteq E$
Működés: Tfh $k(e_1) \leq k(e_2) \leq \dots \leq k(e_m)$, ahol $E = \{e_1, e_2, \dots, e_m\}$. Legyen $F_0 = \emptyset$, és $i = 1, 2, \dots, m$ -re

$$F_i := \begin{cases} F_{i-1} \cup \{e_i\} & \text{ha } F_{i-1} \cup \{e_i\} \text{ körmentes.} \\ F_{i-1} & \text{ha } F_{i-1} \cup \{e_i\} \text{ tartalmaz kört.} \end{cases} \quad F := F_m$$

2.3 Minimális költségű feszítőfák struktúrája

$G = (V, E)$ gráf és $k : E \rightarrow \mathbb{R}_+$ költségfüggvény esetén legyen G_c a legfeljebb c költségű élek alkotta feszítő részgráfja G -nak: $G_c = (V, E_c)$, ahol $E_c := \{e \in E : k(e) \leq c\}$.

Megf: A G gráfon futtatott Kruskal-algoritmus outputja tartalmazza G_c egy feszítő erdejét minden $c \geq 0$ esetén.

Biz: A Kruskal-algoritmus a legfeljebb c költségű (E_c -beli) éleket hamarabb dolgozza fel, mint a c -nél drágábbakat. Ezért E_c összes élének feldolgozása után pontosan azt az állapotot érjük el, mintha a Kruskal-algoritmust a G_c frágon futtattunk volna. Korábban (az ÉHL előtt) láttuk, hogy az utóbbi algoritmus outputja G_c egye feszítő erdeje. \square

Lemma: Tfh $F = \{f_1, f_2, \dots, f_l\}$, $k(f_1) \leq k(f_2) \leq \dots \leq k(f_l)$ és $F \cap E_c$ a G_c egy feszítő erdeje $\forall c \geq 0$ -ra. Tfh $F' = \{f'_1, f'_2, \dots, f'_l\}$ a G egy feszítő erdejének élei, és $k(f'_1) \leq k(f'_2) \leq \dots \leq k(f'_l)$. Ekkor $k(f_i) \leq k(f'_i)$ teljesül $\forall 1 \leq i \leq l$ esetén, így $k(F) \leq k(F')$.

Biz: Indirekt: tfh $k(f_i) > k(f'_i) = c$. Ekkor $|E_c \cap F| < i$, így a feltevés miatt $E_c \cap F$ a G_c egy i -nél kevesebb élű feszítő erdeje. Az f'_1, f'_2, \dots, f'_i élek is mind E_c -beliek, és többen vannak az $E_c \cap F$ feszítő erdő élszámánál. Tehát f'_1, f'_2, \dots, f'_i nem lehet körmentes, így f'_1, f'_2, \dots, f'_l sem. Ez ellentmondás. Tehát $k(f_i) \leq k(f'_i) \forall i$. Ezért $k(F) = \sum_{i=1}^l k(f_i) \leq \sum_{i=1}^l k(f'_i) = k(F')$. \square

Köv: (1) A Kruskal-algoritmus outputja a G gráf egy minimális költségű feszítő erdeje.

Biz: Legyen F a Kruskal-algoritmus outputja. A megfigyelés miatt $F \cap E_c$ a G_c feszítő erdeje $\forall c \geq 0$ -ra, ezért a Lemma szerint $k(F) \leq k(F')$ teljesül G tetszőleges F' feszítő erdejére. \square

Köv: (2) Az F' élhalmaz pontosan akkor minimális költségű feszítő erdeje G -nek, ha $F' \cap E_c$ a G_c egy feszítő erdeje minden $c \leq 0$ -ra.

Biz: A Lemma bizonyítja az elégfégességet.

Biz: A szükségességhez tfh $F' \cap E_c$ nem feszítő erdeje G_c -nek, és legyen F a Kruskal-algoritmus outputja. Mivel $F \cap E_c$ a G_c feszítő erdeje, ezért $|F \cap E_c| > |F' \cap E_c|$, így $k(f_i) < k(f'_i)$ teljesül legalább egy i -re, és minden j -re $k(f_j) \leq k(f'_j)$. Innen $k(F) < k(F')$. \square

Köv: (3) Ha a G gráf összefüggő, akkor G feszítő erdeje a G feszítő fája, így a Kruskal-algoritmus mkffát talál. A (2) következmény pedig G mkffáit karakterizálja.

2.4 Az ötödik elem

Algoritmusok megadásakor öt dologra figyelünk:

Input \checkmark , Output \checkmark , Működés \checkmark , Helyesség \checkmark , Lépésszám.

Utóbbiról nem volt szó a Kruskal-algoritmus esetében.

Tfh n ill. m a G csúcsai ill. élei száma.

A Kruskal-algoritmus két részből áll:

1. Élek költség szerinti sorbarendezése
2. Döntés az egyes élekről a fenti sorrendben.

1. m szám sorbarendezéséhez a buborékredezés legfeljebb $\binom{m}{2}$ összehasonlítást használ.

1. n csúcsú G gráf esetén egy élről döntés megoldható $\text{konst} \cdot \log_2 n$ lépésben az adatstruktúra karbantartásával együtt is. Az összes döntéshez tehát elegendő $\text{konst} \cdot n \cdot \log_2 n$ lépés. A Kruskal-algoritmus lépésszáma ezért felülről becsülhető $\text{konst} \cdot (n + m) \cdot \log_2(n + m)$ -mel.

3 Gráfbejárások és legrövidebb utak

3.1 Általános gráfbejárás & BFS

A gráfbejárási algoritmus az inputgráf csúcsait és éleit fedezi fel. Minden csúcs az eléretlen \rightarrow elért \rightarrow befejezett állapotokat veszi fel. A bejárás akkor ér véget, amint minden csúcs befejezetté vált.

1. Van elért csúcs. Választunk egyet, mondjuk u -t.
 - (1a) Ha van olyan uv él, amire v eléretlen, akkor v elértté válik.
 - (1b) Ha nincs ilyen uv él, akkor u befejezetté válik.
2. Nincs elért csúcs.
 - (2a) Ha van eléretlen u csúcs, akkor u -t elértté tesszük.
 - (2b) Ha nincs eléretlen csúcs (azaz \forall csúcs fejezett), akkor END.

Szélességi bejárás (BFS) szabálya:

Az 1. esetben mindeg a legkorábban elért u -t választjuk.

Input: $G = (V, E)$ (ir/ir.tatlan) gráf, ($v \in V$ gyökérpont¹).

Output: (1) A csúcsok elérési és befejezési sorrendje. (2) Az élek osztályozása:

faél: Olyan él, ami mentén egy csúcs elértté vált.

uv **előreél:** nem faél, de u -ból v -be faélekből irányított út vezet.

uv **visszaél:** v -ből u -ba faélekből irányított út vezet.

keresztél: minden más él (u és v közt nincs leszármazott viszony).

(3) A **bejárás fája:** a faélek alkotta részgráf. (A bejárás fája valójában egy gyökereiből kifelé irányított erdő.)

Megf: Irányítatlan esetben az előreél és a visszaél ugyanazt jelenti.

Terminológia: Ha a bejárás fájában u -ból v -be irányított út vezet, akkor u a v őse és v az u leszármazottja. A faél és az előreél tehát ősből leszármazottba, a visszaél leszármazottból ősbé vezet.

3.2 A BFS tulajdonságai

Nézzük meg egy **irányított** gráf BFS bejárását is.

Állítás: Tfh $G = (V, E)$ BFS bejárása után a csúcsok elérési sorrendje v_1, v_2, \dots, v_n . Ekkor az alábbiak teljesülnek.

(1) Ha $i < j$, akkor v_i -t hamarabb fejezzük be, mint v_j -t, továbbá v_i gyerekei megelőzik v_j gyerekeit az elérési sorrendben.

Biz: A v_i -t befejezésének pillanatában v_i minden gyereke elért, de v_j -nek még egy gyereke sem az. Ezért v_j gyerekeit a v_i csúcs befejezése után érjük el, majd ezt követően fejezzük be v_j -t. \square

(2) **Az elérési és befejezési sorrend (BFS esetén) megegyezik.**

Biz: Ha v_i -t korábban érjük el, mint v_j -t, akkor (1) miatt v_i -t korábban is fejezzük be v_j -nél. Ezért bármely két csúcs sorrendje ugyanaz az elérési sorrendben mint befejezési sorrendben. Tehát az elérési sorrendnek meg kell egyeznie a befejezési sorrenddel. \square

(3) **Gréfél nem ugorhat át falét:** ha $k < i < j \leq l$ és $v_i v_j$ faél, akkor $v_k v_l$ nem lehet gráfél.

Biz: Ha $v_k v_l \in E(G)$, akkor v_l szülője v_k vagy egy v_k -t megelőző csúcs. (1) miatt v_j szülője sem következhet v_k után, vagyis v_i nem lehet v_j szülője.

(4) **Nincs előreél.** (Irányítatlan eset: csak faél és keresztél van.)

Biz: Indirekt: ha $v_i v_j$ előreél lenne, akkor v_i -ből v_j -be irányított út vezetne a BFS-fában, és $v_i v_j$ ennek a faélekből álló útnak az utolsó élét átugraná. \square

¹A gyökérben kezdetben elért állapotú, ezért kivétel az általános szabály alól.

(5) Ha a BFS-fában k -élű irányított út vezet u -ból v -be, akkor G -ben nincs k -nál kevesebb élű uv -út.

Biz: Ha lenne a BFS fa-beli útnál kevesebb élű út G -ben, akkor lenne olyan gráfél, ami faélt ugrik át. \square

(6) **A BFS-fa egy legrövidebb utak fája:** a BFS-fa v_1 gyökeréből bármely v_i csúcsba vezető faút a G egy legkevesebb élű v_1v_i -útja.

3.3 Legrövidebb utak

Def: Adott G (ir) gráf és $l : E(G) \rightarrow \mathbb{R}$ hosszfüggvény esetén egy P út hossza a P éleinek összhossza: $l(P) = \sum_{e \in E(P)} l(e)$.

Az u és v csúcsok **távolsága** a legrövidebb uv -út hossza: $dist_l(u, v) := \min\{l(P) : P \text{ } uv\text{-út}\}$ ($\nexists uv\text{-út} \Rightarrow dist_l(u, v) = \infty$.) Az l hosszfüggvénye **nemnegatív**, ha $l(e) \geq 0$ teljesül minden e élre. Az l hosszfüggvény **konzervatív**, ha G -ben \nexists negatív összhosszú ir. kör.

Cél: Legrövidebb út keresése irányított/irányítatlan gráfban.

Megf: Ha $l(e) = 1$ a G minden e élére, akkor $l(P)$ a P élszáma. Ezért a BFS-fa minden gyökérből elérhető csúcsba tartalmaz egy legrövidebb utat a gyökérből elérhető csúcsba tartalmaz egy legrövidebb utat a gyökérből, azaz a szélességi bejárás tekinthető egy legrövidebb utat kereső algoritmusnak is.

Def: Adott G (ir) gráf, $l : E(G) \rightarrow \mathbb{R}$ hosszfüggvény és $r \in V(G)$. **(r, l) -felső becslés** olyan $f : V(G) \rightarrow \mathbb{R}$ függvény, ami felülről becsli minden csúcs r -től mért távolságát: $dist_l(r, v) \geq f(v) \forall v \in V(G)$.

Triviális (r, l) -felső becslés: $f(v) = \begin{cases} 0 & v = r \\ \infty & v \neq r \end{cases}$

Pontos (r, l) -felső becslés: $f(v) = dist_l(r, v) \forall v \in V(G)$.

3.4 Az elméleti javítás

Def: Tfh f egy (r, l) -felső becslés és $uv \in E(G)$. Az f **uv -elméleti javítása** az az f' , amire $f'(z) = \begin{cases} f(z) & z \neq v \\ \min\{f(v), f(u) + l(uv)\} & z = v \end{cases}$

Megf: Tfh az $l : E(G) \rightarrow \mathbb{R}$ hosszfüggvény konzervatív és $f(r) = 0$.

Ekkor (1) Az $f(r, l)$ -felső becslés élmenti javítása mindig (r, l) -felső becslést ad.

Biz: Azt kell megmutatni, hogy van olyan rv -út, aminek a hossza legfeljebb $f(u) + l(uv)$. Ha egy legrövidebb ru -utat kiegészítünk az uv éllel, akkor olyan rv -élsorozatot kapunk, aminek az összhossza $dist_l(r, u) + l(uv) \leq f(u) + l(uv)$. „Könnyen” látható, hogy az élhosszfűggvény konzervativitása miatt ha van x összhosszúságú rv -élsorozat, akkor van legfeljebb x összhosszúságú rv -út is. Ezek szerint van legfeljebb $f(u) + l(u, v)$ hosszúságú uv -út is, azaz az érdemi élmenti javítás után szintén (r, l) -felső becslést kapunk. \square

(2) $f(r, l)$ -felső becslés (pontosan) $\iff (f\text{-en } \nexists \text{ érdemi élmenti javítás})$.

Biz: \Rightarrow : Ha f pontos, akkor biztosan nincs rajta érdemi élmenti javítás: ha volna, akkor egy felső becslés a pontos érték alá csökkenne, így az élmenti javítás nem (r, l) -felső becslést eredményezne. \Leftarrow : Legyen $v \in V(G)$ tetsz, és legyen P egy legrövidebb rv -út. A P egyik éle mentén sincs érdemi élmenti javítás, ezért P minden u csúcsára pontos a felső becslés: $f(u) = dist_l(r, u)$. Ez igaz az út utolsó csúcsára, a tetszőlegesen választott v -re is. \square

Köv: Adott G , konzervatív l és $r \in V(G)$ esetén ha kiindulunk a triviális (r, l) -felső becslésből, és addig végzünk émj-kat, amíg lehet, akkor a végén megkapjuk minden csúcs r -től való távolságát.

Itt a jegyzet 17. oldaláról az utolsó kettő pont hiányzik, mivel nem tudom, hogy mennyire lényegesek.

Def: Tfh f egy (r, l) -felső becslés és $uv \in E(G)$. Az f uv -élmenti javítása az az f' , amire

$$f'(z) = \begin{cases} f(z) & z \neq v \\ \min\{f(v), f(u) + l(uv)\} & z = v \end{cases}$$

Megf: Tfh az $l : E(G) \rightarrow \mathbb{R}$ hosszfüggvény konzervatív és $f(r) = 0$. Ekkor (1) Az $f(f, l)$ -felső becslés élmenti javítása mindig (r, l) -felső becslést ad. (2) $f(r, l)$ -felső becslés (pontosan) $\Leftrightarrow (f\text{-en } \nexists \text{ érdemi élmenti javítás})$.

Dijkstra-algoritmus: Input: $G = (V, E), l : E \rightarrow \mathbb{R}_+, r \in V$. Output: $\text{dist}_l(r, v) \forall v \in V$
Működés: $U_0 := \emptyset, f_0$ a triviális. (r, l) -felső becslés.

Az i -dik fázis:

1. Legyen $U_i := U_{i-1} \cup \{u_i\}$, ahol u_i olyan csúcs a $V \setminus U_{i-1}$ halmazból, amelyre $f_{i-1}(v)$ minimális.
2. $f_i : f_{i-1}$ élmenti javítása minden U_i -ből kivezető $u_i x$ élen. Output: $f_{|V|}$. Megjelöljük a végső $f_{|V|}(V)$ értékeket beállító éleket.

3.5 Dijkstra, egy példán

Dijkstra-algoritmus: Input: $G = (V, E), l : E \rightarrow \mathbb{R}_+, r \in V$. Output: $\text{dist}_l(r, v) \forall v \in V$
Működés: $U_0 := \emptyset, f_0$ a triviális. (r, l) -felső becslés.

Az i -dik fázis:

1. Legyen $U_i := U_{i-1} \cup \{u_i\}$, ahol u_i olyan csúcs a $V \setminus U_{i-1}$ halmazból, amelyre $f_{i-1}(v)$ minimális.
2. $f_i : f_{i-1}$ élmenti javítása minden U_i -ből kivezető $u_i x$ élen. Output: $f_{|V|}$. Megjelöljük a végső $f_{|V|}(V)$ értékeket beállító éleket.

Megf: Ha a v -be vezet megjelölt él, akkor vezet r -ből v -be megjelölt éleken út, és ennek hozzá megegyezik $f_{|V|}(v)$ -vel.

Biz: $f_{|V|}(r) = 0$, és a megjelölt élek mentén haladva az $f_{|V|}$ érték az élhosszal növekszik. \square

Köv: Ha a Dijkstra-algoritmus helyes, akkor az algoritmus végén a megjelölt élek egy legrövidebb utak fáját alkotják r gyökérrel.

3.6 Dijkstra helyessége

Megf: Tfh u_1, u_2, \dots, u_n a G csúcsainak sorrendje a Dijkstra-algoritmus végrehajtása után.

(1) Ekkor $f_{|V|}(u_i) \leq f_{|V|}(u_{i+1})$ teljesül $\forall 1 \leq i \leq n$.

Biz: Az i -dik fázisban $f_i(u_i) \leq f_i(u_{i+1})$ teljesült az u_i választása miatt. Ezek után $f_i(u_i)$ már nem változott: $f_{|V|}(u_i) = f_i(u_i)$. Ugyan $f_i(u_{i+1})$ még csökkenhetett, de csak az $u_i u_{i+1}$ él mentén történt javítás miatt, hiszen az $(i+1)$ -dik fázisban u_{i+1} bekerült az U_i halmazba, és a hozzá tartozó (r, l) -fb már nem csökken tovább. Ekkor $f_{i+1}(u_{i+1}) = \min\{f_i(u_{i+1}), f_i(u_i) + l(u_i u_{i+1})\} \geq f_i(u_i)$, mivel $l(u_i u_{i+1}) > 0$. Ezért $f_{|V|}(u_i) = f_i(u_i) \leq f_{i+1}(u_{i+1}) = f_{|V|}(u_{i+1})$ \square

(2) $f_{|V|}(u_1) \leq f_{|V|}(u_2) \leq \dots \leq f_{|V|}(u_n)$

(3) A Dijkstra-algoritmus outputjaként kaptt $f_{|V|}$ -n élmenti javítás nem tud változtatni.

Biz: Tegyük fel, hogy $u_i u_j \in E(G)$ a G egy tetszőleges éle. Ha $i > j$, akkor (2) miatt $f_{|V|}(u_i) \geq f_{|V|}(u_j)$, ezért az $u_i u_j$ mentén történő javítás nem tudja $f_{|V|}(u_j)$ -t csökkenteni, hisz $l(u_i u_j)$ pozitív. Ha pedig $i < j$, akkor az i -dik fázisban megrörtént az $u_i u_j$ mentén történő javítás, és ezt követően $f(u_i)$ nem változott, azaz $f_{|V|}(u_i) = f_i(u_i)$. A másik (r, l) -felső becslés pedig csak tovább csökkenhetett a későbbi émj-ok során $f_{|V|}(u_j) \leq f_i(u_j)$. Ezért az $u_i u_j$ él mentén sem az i -dik fázisban, sem később nincs érdemi javítás. \square

Tétel: A Dijkstra-algoritmus helyesen működik, azaz G minden csúcsára igaz, hogy $\text{dist}(r, v) = f_{|V|}(v)$.

Biz: A Dijkstra-algoritmus az f_0 triviális (r, l) -felső becslésből indul ki, és élmenti javításokat alkalmaz. Így minden f_i (speciálisan $f_{|V|}$ is) (r, l) -felső becslés lesz. A fenti (3)-as megfigyelés miatt $f_{|V|}$ -n nem végezhető érdemi élmenti javítás. Ezért egy korábbi (2)-es megfigyelés miatt $f_{|V|}$ pontos (r, l) -felső becslés, azaz $f_{|V|}(v) = \text{dist}_l(r, v) \forall v \in V(G)$. \square

„Lépésszámanalízis”: Ha a G gráfnak n csúcsa és m éle van, akkor a Dijkstra-algoritmus n -szer keresi meg legfeljebb n szám minimumát, ami összességében legfeljebb $\text{konst} \cdot n^2$ lépést igényel. Ezen kívül legfeljebb m élmenti javítást végez, ami $\text{konst}' \cdot m$ lépés. Összességében tehát legfeljebb $\text{konst}'' \cdot (n^2 + m)$ lépésre van szükség, az algoritmus hatékony.

4 Legrövidebb utak, DFS, PERT

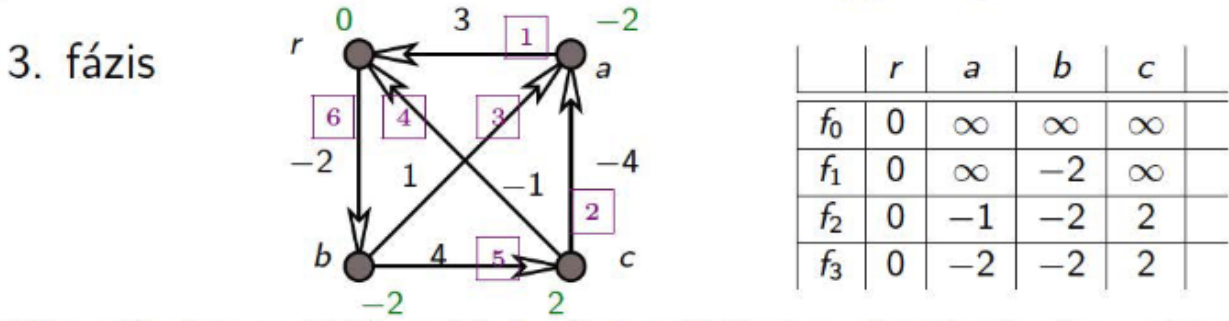
4.1 Legrövidebb utak konzervatív hosszfüggvény esetén 1

Könnyű olyan példát találni, ahol a Dijkstra-algoritmus konzervatív hosszfüggvény esetén hibás eredményt ad. Azonban konzervatív hosszfüggvény esetén is igaz, hogy

- (r, l) -fb élmenti javítása (r, l) -fb-t eredményez, ill.
- ha egy (r, l) -fb-ben nem végezhető érdemi élmenti javítás, akkor pontos.

konzervatív hosszfüggvény esetén is hasonló startégiát követünk: Élmenti javításokat végzünk a triviális (r, l) -fb-en, míg van érdemi javítás.

Ford-algoritmus: Input: $G = (V, E), l : E \rightarrow \mathbb{R}, r \in V$. Output: $dist_l(r, l) \forall v \in V$ Működés: f_0 a triviális (r, l) -fb, $|V| = n, E = \{e_1, e_2, \dots, e_m\}$. Az i -dik fázis $i = 1, 2, \dots, n - 1$ -re az alábbi. f_i -t f_{i-1} -ből kapjuk, az e_1, \dots, e_m élmenti javítások után. OUTPUT: $dist_l(r, v) = f_{n-1}(v) \forall v \in V$.



Állítás: Ha l konzervatív, akkor $dist_l(v) \forall v \in V$.

Biz: $f_1(v) = dist_l(r, v)$ ha $\exists \leq 1$ -élű legrövidebb rv -út. $f_2(v) = dist_l(r, v)$ ha $\exists \leq 2$ -élű legrövidebb rv -út. ... $f_{n-1}(v) = dist_l(r, v)$ ha $\exists \leq (n - 1)$ -élű legrövidebb rv -út. Tehát $f_{n-1}(v) = dist_l(r, v) \forall v \in V$. \square

Megf: Ha $f_i = f_{i-1}$, akkor a Ford-algoritmust az i -dik fázis után be lehet fejezni, hisz nincs érdemi élmenti javítás, így $f_{n-1} = f_i$.

Megj: Az $f_{n-1}(v)$ -t beállító élek legrövidebb utak fáját alkotják.

Biz: A Dijkstra esethez hasonló. Tetszőleges v csúsból visszafelé követve a végső értékeket beállító éleket $f_{n-1}(v)$ hosszúságú rv -utat találunk. \square

"Lépésszámanalízis": Ha a $|V(G)| = n$ és $|E(G)| = m$, akkor minden fázisban $\leq m$ élmenti javítás, ami $konst \cdot m$ lépés. Ez összesen $\leq konst \cdot (n - 1) \cdot m \leq konst \cdot n^3$ lépés, az algoritmus hatékony.

4.2 Legrövidebb utak konzervatív hosszfüggvény esetén 2

Tegyük fel, hogy $G = (V, E), l : E \rightarrow \mathbb{R}$ és $V = \{v_1, v_2, \dots, v_n\}$. Jelölje $d^{(k)}(i, j)$ a legrövidebb olyan $v_i v_j$ -út hosszát, aminek belső csúcsai csak v_1, v_2, \dots, v_k lehetnek.

Megf: (1) $d^{(n)}(i, j) = dist_l(v_i, v_j), v_i v_j \in E \Rightarrow d^{(0)}(i, j) = l(v_i, v_j)$ (2) $d^{(0)}(i, j) = 0$, különben $d^{(0)}(i, j) = \infty$. (3) Ha l konzervatív, akkor tetszőleges i, j ill. $k \leq n$ esetén $d^{(k+1)}(i, j) = \min\{d^{(k)}(i, j), d^{(k)}(i, k+1) + d^{(k)}(k+1, j)\}$ teljesül.

Biz: Tekintsünk egy $d^{(k+1)}(i, j)$ -t meghatározó P utat.

I. eset: $v_{k+1} \notin P$. Ekkor $d^{(k+1)}(i, j) = d^{(k)}(i, j)$, és $d^{(k+1)}(i, j) \leq d^{(k)}(i, k+1) + d^{(k)}(i, k+1) + d^{(k)}(k+1, j)$.

II. eset: $v_{k+1} \in P$. Ekkor $d^{(k+1)}(i, j) \leq d^{(k)}(i, j)$, és $d^{(k+1)}(i, j) = d^{(k)}(i, k+1) + d^{(k)}(k+1, j)$. Mindkét esetben helyes a képlet. \square

Floyd-algoritmus: Input: $G = (V, E)$, konzervatív $l : E \rightarrow \mathbb{R}$. Output: $dist_l(u, v) \forall u, v \in V$
Működés: $d^{(0)}$ felírása (2) alapján. Az i -dik fázis: $d^{(i-1)}$ -ből meghatározzuk $d^{(i)}$ -t (3) alapján.
OUTPUT: $d^{(n)}(u, v) = dist_l(u, v) \forall u, v \in V$.

”Lépésszámanalízis”: A $d^{(0)}$ felírása $konst \cdot n^2$ lépés. Minden fázis $konst' \cdot n^2$. Mivel összesen n fázis van, a lépésszám legfeljebb $konst'' \cdot n^3$ lépés, az algoritmus hatékony.

Ford vs Floyd: Konzervatív hosszfüggvényre működnek helyesen. Mindkét algoritmus talál bizonyítékot, ha l nem konzervatív. (!!)

A Ford csak egy gyökérből, a Floyd bármely két csúcs között talál legrövidebb utat. (!!)

A Ford ritka gráfokra jelentősen olcsóbb, sok él esetén a Floyd nem sokkal drágább.

4.3 Depth First Search (DFS)

”Mélységi bejárás (DFS): A bejárás során mindig a legutolsónak elért csúcsot választjuk az [1.] esetben.

Mélységi és befejezési számozás: DFS után $m(v)$ ill. $b(v)$ a v csúcs elérési ill. befejezési sorrendben kapott sorszáma.

Megj: A BFS konkrét megvalósításában szükség van arra, hogy az **elért** csúcsokat úgy tároljuk, hogy könnyű legyen kiválasztani az **elért** csúcsok közül a legkorábban elértet. Erre egy célszerű adatstruktúra a *sor* (avagy *FIFO lista*). Ha a BFS megvalósításában ezt az adatstruktúrát *veremre* (más néven *FIFO listára*) cseréljük, akkor a DFS egy megvalósítása adódik.

Megf: Tegyük fel, hogy a G gráf éleit DFS után osztályoztuk.

(1) Ha uv **faél**, akkor $m(u) < m(v)$ és $b(u) > b(v)$.

Biz: v -t u -ból értük el, ezért $m(u) < m(v)$. A v elérésekor u és v elért állapotúak. A DFS szerint v -t u elptt fejezzük be. \square

(2) Ha uv **előreél**, akkor $m(u) < m(v)$ és $b(u) > b(v)$.

Biz: u -ból v -be faéleken keresztül vezet irányított út. (1) miatt az út mentén a mélységi szám növekszik, befejezési csökken. \square

(3) Ha uv **visszaél**, akkor $m(u) > m(v)$ és $b(u) < b(v)$.

Biz: v -ből u -ba faéleken keresztül vezet irányított út. (1) miatt az út mentén a mélységi szám növekszik, a befejezési csökken. \square

Biz: $m(u) < m(v)$ esetén a DFS miatt v az u leszármazottja lenne. Ezért $m(u) > m(v)$. Ha u -t a v befejezése előtt értenék el, akkor u a v leszármazottja lenne. Ezért az alábbi sorrendben történik u és v evolúciója: **v** **elérése**, **v** **befejezése**, **u** **elérése**, **u** **befejezése**. \square

(4) Ha uv **keresztél**, akkor $m(u) > m(v)$ és $b(u) > b(v)$.

(5) Irányítatlan gráf DFS bejárása után nincs keresztél.

Biz: Indirekt. Ha uv keresztél, akkor (4) miatt $m(u) > m(v)$, továbbá vu is keresztél, ezért $m(v) > m(u)$. Ellentmondás. \square

(6) Ha DFS után van visszaél, akkor G tartalmaz irányított kört.

Biz: A DFS fa visszaélhez tartozó alapköre a G egy irányított köre. \square

(7) Ha DFS után nincs visszaél, akkor G -ben nincs irányított kör.

Biz: Bmely irányított körnek van olyan uv éle, amire $b(u) < b(v)$. Ez az él csak visszaél lehet. \square

4.4 Direct Acyclic Graphs

Def: A $G = (V, E)$ irányított gráf **aciklikus** (más néven **DAG**), ha G nem tartalmaz irányított kört.

Példa: DAG-ot úgy kaphatunk, hogy egy G irányítatlan gráf csúcsait csupa különbözőszámmal megszámozzuk, és minden élt a kisebb számot viselő csúcsból a nagyobbba irányítunk.

Ha ugyanis lenne az így megírányított gráfban írányított kör, akkor az élei mentén a számok végig növekednének, ami lehetetlen. Azt fogjuk ihazolni, hogy a fenti példa minden DAG-ot leír.

Def: A $G = (V, E)$ írányított gráf csúcsainak **topologikus sorrendje** alatt a csúcsok olyan sorrendjét értjük, amire igaz, hogy minden írányított él a sorban előbb álló csúcsból vezet a sorban későbbi csúcsba. ($V = \{v_1, v_2, \dots, v_n\}, v_i v_j \in E \Rightarrow i < j$)

Tétel: (G írányított gráf DAG) $\Leftrightarrow (V(G)$ -nek \exists topologikus sorrendje).

Biz: Tegyük fel, hogy \exists topologikus sorrend. Láttuk, hogy G ekkor DAG. \checkmark

Biz: Most tegyük fel, hogy G DAG, és futtassunk rajra egy DFS-t. Láttuk, hogy a DFS után nem lesz visszaél, ezért minden uv írányított élre $b(u) > b(v)$ teljesül. Ezért a csúcsok befejezési sorrendjének megfordítása a G csúcsainak egy topologikus sorrendje. \square

Köv: Írányított gráf aciklikussága DFS-sel gyorsan eldönthető: ha van visszaél, akkor a visszaél DFS-fabeli alapköre G egy írányított köre, így G nem DAG. Ha pedig nincs visszaél, akkor a fordított befejezési sorrend a G egy topologikus sorrendje, G tehát DAG.

Megj: DAG-ban topologikus sorrendet forráskeresések és forrástörlések alkalmazásával is találhatunk.

4.5 Leghosszabb út keresése

Ötlet: Az $l'(uv) = -l(uv)$ élhosszokkal a leghosszabb utak legrövidebbekké válnak. Olyanokat pedig tudunk keresni.

Gond: A módszerünk csak konzervatív élhosszokra működik. Írányítatlan gráfon ez nemnegatív élhosszokat jelent, ezért ez az ötlet itt nem segít. Írányított esetben nem baj a negatív élhossz, feltéve, hogy G DAG. Ekkor Ford, Floyd bármelyike használható.

Jó hír: Van egy még gyorsabb módszer: a dinamikus programozás. Ennek segítségével tetszőleges G DAG minden v csúcsához ki tudjuk számítani a v -be vezető leghosszabb utat. (Sőt! ...)

Leghosszabb út DAG-ban: Input: $G = (V, E)$ DAG, $l : E \rightarrow \mathbb{R}$. Output: $\max\{l(P) : P v\text{-be vezető út}\}$ minden $v \in V$ csúcsra. Működés: $[1] V = \{v_1, v_2, \dots, v_n\}$ topologikus sorrend meghatározása. $[2] i = 1, 2, \dots, n : f(v_i) = \max\{\max\{f(v_j) + l(v_j v_i) : v_j v_i \in E\}, 0\}$ Output: $f(v) \forall v \in V$

Helyesség: Ha a v_i -be vezető leghosszabb út utolsó előtti csúcsa v_j , akkor $f(v_i) = f(v_j) + l(v_j v_i)$.

Megj: Ha a fenti algoritmusban minden csúcsra megjelöljük az $f(v)$ értéket beállító élt (éleket), akkor a megjelölt élek minden v csúcsba megadnak egy leghosszabb utat. Sőt: minden v -be vezető leghosszabb megkapható így.

4.6 A PERT probléma

Egy a, b, \dots tevékenységekből álló projektet kell végrehajtunk.

Precedenciafeltételek: bizonyos (u, v) párok esetén előírás, hogy az u tevékenységet a v előtt kell elvégezni, ezért v az u kezdetét követően $c(uv)$ időkorlát elteltével kezdhető.

Cél: minden v tevékenységhez olyan $k(v) \geq 0$ kezdési időpont meghatározása, ami nem sérti a preferenciafeltételeket, és a projekt végrehajtási ideje (a legnagyobb $k(v)$ érték) minimális.

G **írányított gráf** csúcsai a tevékenységek, élei pedig a precedenciafeltételek, az uv él hossza $c(uv)$.

Megf: (1) Ha G nem DAG, akkor a projekt nem hajtható végre. (2) Ha G DAG, akkor minden v tevékenység legkorábbi kezdési időpontja a v -be vezető leghosszabb út hossza.

Köv: A PERT probléma megoldása nem más, mint a G DAG minden csúcsára az oda vezető leghosszabb út meghatározása.

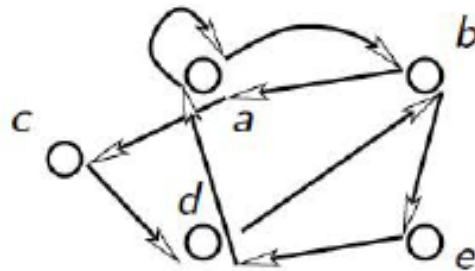
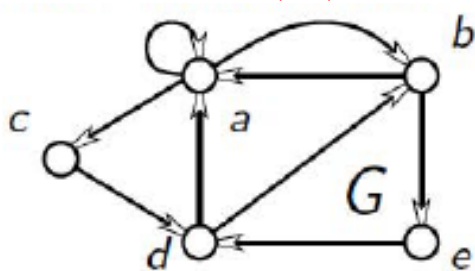
Terminológia: G leghosszabb útja **kritikus út**, amivől több is lehet. Kritikus út csúcsai a **kritikus tevékenységek**.

Megf: Ha egy kritikus tevékenység nem kezdődik el a lehető legkorábbi időpontban, akkor az egész projekt végrehajtása csúszik.

5 Euler-séták és Hamilton-körök

5.1 Euler-séták

Def: A G gráf **Euler-(kör)sétája** a G egy olyan (kör)sétája, ami G minden élét tartalmazza.



Megj: (1) A fenti definíció 2×2 fogalmat definiál: az Euler-sétát és az Euler-körsétát irányítatlan és irányított gráfra is. (2) Szokás a definíciót abban a formában kimondani, hogy az Euler-(kör)séta G minden élét **pontosan** egyszer tartalmazza. Tekintettel arra, hogy egy séta nem mehet át kétszer ugyanazon az élen, ez redundáns kiváncságnak, hiszen következménye az általunk használt definíciónak. Használatos ezen kívül az Euler-kör ill. Euler-út megnevezés is a fenti fogalmakra. (3) Irányítatlan Euler-séta: " G egy vonallal lerajzolható".

Cél: Gyors módszer az Euler-(kör)séta megtalálására, létezésének ellenőrzésére.

Megf: (1) Ha a G irányított gráfnak van Euler-körsétája, akkor

- (a) G izolált pontoktól eltekintve gyengén összefüggő, és
- (b) minden v csúcsára $\rho(v) = \delta(v)$ teljesül.

Biz: (a) Ha G két különböző gyenge komponense is tartalmaz élt, akkor G -nek nem lehet Euler-körsétája, hisz egyetlen séta sem tartalmazhat élt két különböző gyenge komponensből. ✓

(b) Ha végighaladunk az Euler-körsétán, akkor a v csúcsba pontosan annyiszor lépünk be, mint ahányszor kilépünk onnan. A körséta G minden élét pontosan egyszer érinti: $\rho(v) = \delta(v)$

□

Megf: (2) Ha a G irányítatlan gráfnak van Euler-körsétája, akkor

- (a) G izolált pontoktól eltekintve összefüggő, és
- (b) G -ben minden fokszám páros.

Biz: Az irányított esethez hasonló. (a) Egy (kör)séta nem tartalmazhatja két különböző komponensnek is $1-1$ élét, és (b) az Euler-körsétát követve tetszőleges v csúcsba ugyanannyiszor lépünk be, mint ahányszor kilépünk belőle. Ezért $d(v)$ páros. □

Megf: (3) Ha a G irányítatlan gráfnak van Euler-sétája, akkor

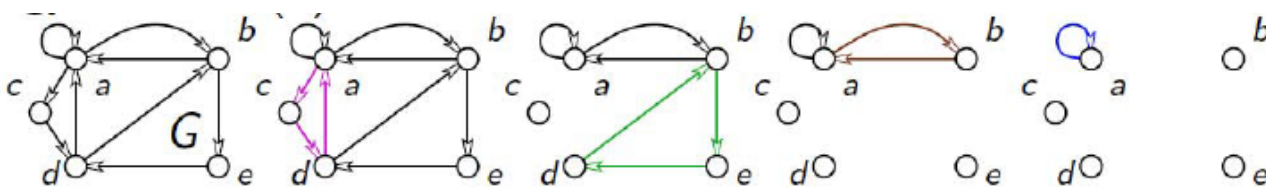
- (a) G izolált pontoktól eltekintve összefüggő, és
- (b) G -nek 0 vagy 2 páratlan fokú csúcsa van.

Biz: (a) ✓. (b): Tegyük fel, hogy G Euler-sétája egy uv -séta. Ekkor minden $w \neq u, v$ csúcsra $d(w)$ kétszer annyi, mint ahányszor az Euler-séta w -n áthalad, vagyis $d(w)$ páros. Ha $u = v$, akkor az Euler-séta körséta, így $d(u)$ is páros (2b) miatt. Ha pedig $u \neq v$, akkor u -ból 1-gyel többször lépünk ki, mint be, v -be 1-gyel többször lépünk be, mint ki, vagyis $d(u)$ és $d(v)$ páratlanok. □

Megj: A fenti Megfigyelés segítségével bizonyos esetekben azonnal látszik, hogy G -nek nincs Euler-sétája ill. -körsétája.

G **irányítatlan Euler-gráf**, ha G minden v csúcsra $d(v)$ páros.

Lemma: Ha G Euler-gráf, akkor G élei kiszínezhetők úgy, hogy az egyszínű élek (irányított) kört alkossanak minden színre.

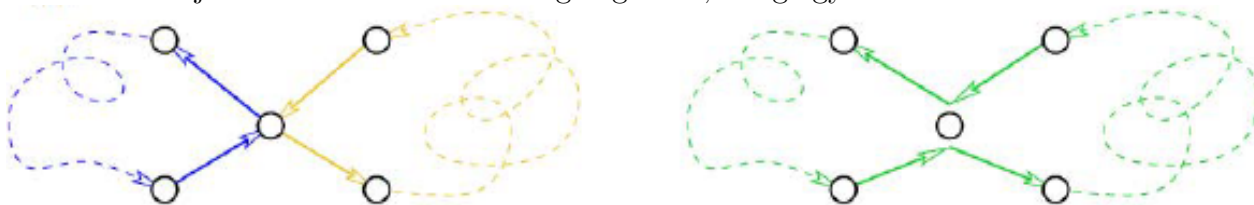


Biz: Induljunk el G egy éle mentén, és haladjunk tovább az (irányított) élek mentén. Mivel G Euler, ezért sosem adaunk el: előbb-utóbb ismétlődik egy csúcs, így találunk egy C_1 kört. C_1 éleit törölve $G - C_1$ Euler-gráf marad. Ismételjük meg ezt a $G - C_1$ gráfon. Így G minden éle előbb-utóbb sorra kerül és megkapja a C_2, C_3, \dots köröket. Ezért $E(G) = C_1 \cup C_2 \cup \dots$ diszjunkt körök uniójára bomlik fel. Színezzük ki a C_1 kör éleit az i -dik színnel. \square

Tétel: (1) (G irányított gráfnak van Euler-körsétája) \iff (G Euler-gráf és G izolált pontoktól eltekintve gyengén összefüggő)

(2) (G irányítatlan gráfnak van Euler-körsétája) \iff (G Euler-gráf és G izolált pontoktól eltekintve összefüggő)

Biz: \Rightarrow : Láttuk. $\checkmark \Leftarrow$: A Lemma miatt $E(G)$ felbontható körökre, tehát körsétákra is. Ha a körséták száma legalább 2, akkor választunk két körsétát, aminek van közös csúcsa és e csúcs mentén "összevarjuk" azokat. Mindezt addig végezzük, amíg egyetlen körséta marad. \square



Tétel: (3) (G irányítatlan gráfnak van Euler-sétája) \iff (G izolált pontoktól eltekintve összefüggő és 0 vagy 2 páratlan fokú csúcsa van.)

Biz: \Rightarrow : Láttuk. $\checkmark \Leftarrow$: Ha G Euler-gráf, akkor (2) miatt van Euler-körsétája, ami Euler-séta is egyúttal. Ha G nem Euler-gráf, akkor legyenek u és v a G páratlan fokú csúcsai. Ekkor $G + uv$ Euler-gráf, és (2) miatt van Euler-körsétája. Feltehető, hogy e körséta utolsó éle uv . Ezt az uv élt elhagyva a körsétából, G Euler-sétáját kapjuk. \square

Euler-körséta keresése Euler-gráfban: $E(G)$ -t felbontjuk körsétákra, amiket összevarrunk. Körsétát a felbontáshoz pl. úgy is kereshetünk, hogy addig követünk egy sétát, amíg tudunk. Előbb-utóbb elakadunk, de ez csakis a séta kiindulási pontjában történhet meg. Ezért a bejárt séta egy körséta, amit a felbontásban felhasználunk.

5.2 Hamilton-körök és -utak

Def: A G gráf **Hamilton-köre** (**Hamilton-útja**) a G olyan köre (útja), ami G minden csúcsát tartalmazza.

Megj: A célunk hasonló, mint az Euler-(kör)séta esetén, azaz gyors módszer, amivel el lehet dönteni egy gráfról, hogy van-e Hamilton-köre ill. -útja. Sajnos jól használható szükséges és elégséges feltételt nem tudunk adni erre a problémára, és jó oka van annak, hogy nem is számítunk ilyen feltétel létezésére. Tudunk viszont jól használható szükséges, és jól használható elégséges feltételt adni, de ezek csak bizonyos gráfok esetén hasznosak.

Szükséges feltétel Hamilton-kör és -út létezésére

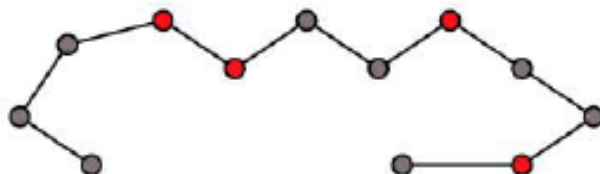
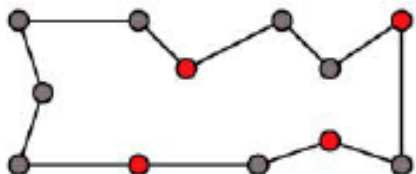
(1) Ha a G gráfnak van Hamilton-köre, akkor $\forall U \subseteq V(G)$ esetén $G - U$ komponenseinek száma legfeljebb $|U|$.

(2) Ha a G gráfnak van Hamilton-útja, akkor $\forall U \subseteq V(G)$ esetén $G - U$ komponenseinek száma legfeljebb $|U| + 1$.

Megj: A fenti feltétel, miszerint k csúcs törlésétől a gráf legfeljebb k (ill. $k + 1$) komponensre eshet szét feltétlenül **szükséges** ahhoz, hogy G -nek legyen Hamilton-köre ill. -útja. Abból

azonban, hogy G teljesíti a fenti feltételt, nem következik, hogy G -nek csakugyan van Hamilton-köre vagy útja. A szükséges feltételt úgy tudjuk alkalmazni, hogy a segítségével igazoljuk egy konkrét gráfról, hogy nincs Hamilton-köre (vagy -útja). Ha pl. azt látjuk, hogy G -ből 42 csúcsot elhagyva 43 komponens keletkezik, akkor G -nek nincs Hamilton-köre. Ha a komponensszám legalább 44, akkor G -nek Hamilton-útja sincs.

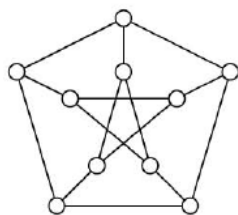
Biz: (1,2) G -t tekinthetjük úgy, mint egy kör (ill. út), amihez még további éleket adunk hozzá. Könnyű látni, hogy egy kör (ill. út) k pont elhagyásától legfeljebb k ($k+1$) komponensre eshet szét. A további élek (amit a körhöz ill. úthoz hozzá kell adni, hogy G -t kapjunk) csak csökkenteni tudják a komponensszámot, növelni nem. Ezért G -ből k csúcsot törölve legfeljebb k ($k+1$) komponens keletkezik. \square



Megj: Az ábrán látható Petersen-gráfnak (sok más mellett) két érdekes tulajdonsága van.

1. Teljesíti a fenti szükséges feltételt.

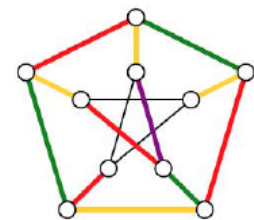
- (a) Tegyük fel, hogy külső körből k_1 , a belsőből k_2 csúcsot hagytunk el. Ha $k_1 = 0$ vagy $k_2 = 0$, akkor a gráf összefüggő marad. Különben a külső kör legfeljebb k_1 , a belső pedig legfeljebb k_2 részre esik szét, vagyis összesen legfeljebb $k_1 + k_2$ komponens



létezik.

2. Nincs Hamilton-köre.

- (a) Ha lenne Hamilton-köre, akkor a Hamilton-kör éleit felváltva pirosra és zöldre tudnánk színezní. Ha a körön kívüli élek sárgák, akkor a 3-regularitás miatt minden csúcsból pontosan egy piros, sárga ill. zöld él indulna. Ha megpróbáljuk az éleket így kiszínezní,



kiderül, hogy nem lehet.

A továbbiakban elégséges feltételeket fogunk látni Hamilton-kör létezésére. Ezek segítségével (szerencsés esetben) gyorsan és kétséget kizáróan tudjuk bizonyítani, hogy egy adott gráfnak van Hamilton-köre. Az elégséges feltétel vizsgálata azonban nem alkalmas arra, hogy egy gráf a Hamilton-körének hiányát igazoljuk.

Def: Legyen G n -csúcsú, egyszerű gráf.

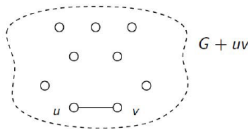
Az $u, v \in V(G)$ csúcspár **gazdag**, ha $d(u) + d(v) \geq n$. A G gráfra teljesül a **Dirac-feltétel**, ha $d(v) \geq \frac{n}{2} \forall v \in V(G)$ -re. G -re igaz az **Ore-feltétel**, ha G bármely két nem szomszédos csúcsa gazdag párt alkot.

Dirac tétele: G -re igaz a Dirac-feltétel $\Rightarrow G$ -nek van H-köre.

Ore tétele: G -re igaz az Ore-feltétel $\Rightarrow G$ -nek van H-köre.

Megj: A Dirac-feltétel erősebb (többet kíván), mint az Ore. Ezért az Ore-tétel erősebb, mint a Dirac: gyengébb feltételből igazolja ugyanazt. Ezért az Ore-tétel bizonyítása a Dirac-tételt is igazolja.

5.3 A Chvátal-lezárt



Hízlalási lemma: Tegyük fel, hogy G egyszerű gráf, és (u, v) gazdag pár. (G -nek van Hamilton-köre) $\iff (G + uv$ -nek van Hamilton köre).

Megj: A hízlalási lemma jelentősége az, hogy segít eldönteni azt, hogy van-e G -ben Hamilton-kör. Azt mondja ki ugyanin, hogy a gazdag párok közé G -be "ingyen" behúzhatunk éleket, u.i. ez nem változtat azon a tényen, hogy van vagy nincs Hamilton-kör a vizsgált gráfban. Megtehetjük tehát, hogy a lemma segítségével addig húzunk be éleket a gráfba, amíg lehet. Ha az így adódó \overline{G} Chvátal-lezártban találunk Hamilton-kört, akkor G -nek is bizonyosan van Hamilton-köre. Ha pedig \overline{G} nem tartalmaz Hamilton-kört, akkor persze G -nek nincs Hamilton-köre.

Biz: \Rightarrow : Láttuk. \Leftarrow : Legyen C a $G + uv$ Hamilton-köre. Ha $uv \notin C$, akkor C a G -nek is Hamilton-köre, kész vagyunk. Ha viszont $uv \in C$, akkor $C - uv$ a G egy Hamilton-útja. Legyen ez a Hamilton-út $u = v_1, v_2, \dots, v_n = v$. Legyen $A := N(v) = \{v_i : vv_i \in E(G)\}$ a v szomszédainak halmaza, és legyen $B := \{v_{i-1} : uv_i \in E(G)\}$ az u szomszédait a Hamilton-úton megelőző csúcsok halmaza.

Világos, hogy $v \notin A$ és $v \notin B$, így $|A \cup B| \leq n - 1$. Mivel (u, v) gazdag pár, ezért $|A| + |B| = d(u) + d(v) \geq n$. Ezek szerint $A \cap B \neq \emptyset$, legyen pl. $v_i \in A \cap B$. Ekkor $v_1, v_2, \dots, v_i, v_n, v_{n-1}, \dots, v_{i+1}, v_1$ a G egy Hamilton-köre. \square

Ore tétele: Ha G bármely két nemszomszédos csúcsa gazdag párt alkot, akkor G -nek van Hamilton-köre.

Biz: A hízlalási lemma alapján G bármely két nemszomszédos csúcsát "ingyen" összeköthetjük. Így G Chátal-lezártja a $\overline{G} = K_n$ teljes gráf. Mivel K_n -nek van H-köre, ezért G -nek is van. \square

Dirac-tétele: Ha $\delta(G) \geq \frac{|V(G)|}{2}$, akkor G -nek van Hamilton-köre.

Biz: G bármely két csúcsa gazdag párt alkot, ezért G -re teljesül az Ore-feltétel. Az Ore-tétel miatt G -nek van Hamilton-köre. \square

6 Síkgráfok

6.1 Síkbarajzolhatóság

Def: **Síkbarajzolt (SRt)** gráf alatt olyan gráfdiagramot értünk, amiben az élek nem keresztezik egymást.

A G gráf **síkbarajzolható (SRható)**, ha van SRt diagramja.

Síkbarajzolt gráf **tartománya (lapja)**: a diagram komplementerének összefüggő tartománya. A nem korlátos rész neve **külső tartomány**.

Megj: (1) A fentieket nem csak egyszerű gráfokra definiáltuk. (2) A SRt gráf nem csupán egy gráf, hanem egy **konkrét** diagram. (3) Ugyanannak a SRható gráfnak nagyon sok lényegesen különböző síkbarajzolt diagramja (lerajzolása) lehet. (4) A görbe (tóruszra) rajzolhatóság hasonlóan definiálható.

Állítás: $(A \text{ } G \text{ gráf SRható}) \iff (G \text{ gömbre rajzolható})$

Biz: A sztereografikus projekcióban az északi-sarkból történő vetítés kölcsönösen egyértelmű megfeleltetés a sík pontjai és a síkot a déli-sarkon érintő gömbfelszín pontjai (mínusz északi-sark) között. A síkbarajzolt diagram vetülete gömbre rajzolt lesz ($\Rightarrow \checkmark$), és az \bar{E} -t nem tartalmazó gömbre rajzolt diagram pedig síkbarajzoltává válik. A \Leftarrow irány igazolásához csupán annyi kell, hogy úgy rajzoljuk G -t a gömbre, hogy az \bar{E} -n ne menjen át él. \square

Köv: SRt gráf külső tartománynak nincs kitüntetett szerepe.

Biz: Bármely lerajzolás "kifordítható": a diagram átrajzolható úgy, hogy a kiválasztott tartomány legyen a külső tartomány.

1. Vetítsük fel a diagramot a gömbre.
2. Állítsuk az \bar{E} -t a kiválasztott tartománynak megfelelő gömbi tartomány belsejébe.
3. Vetítsük vissza a gömbre rajzolt gráfot a síkra. \square

Köv: Bármely konvex poliéder élhálójá SRható gráf.

Biz: A kx poliéder belső pontjából az élháló kivetíthető egy, a poliédert tartalmazó gömbre. Így az élhálóból gömbre rajzolt gráf lesz. Láttuk, hogy minden gömbre rajzolható gráf SRható. \square

Megj: A kx poliéder élgráfjának tartományai a poliéder lapjainak felelnek meg.

Terminológia: SRt G gráf esetén n, e, t ill. k jelöli rendre a G csúcsai, élei, tartományai és komponensei számát.

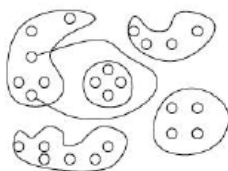
Duális kézfogáslemma (DKFL): Ha G SRt gráf, akkor $\sum_{i=1}^t l_i = 2e$ ahol l_i az i -dik lapot határoló élek számát jelöli.

Biz: Minden él vagy két különböző lapot határol, vagy ugyanazt a lapot 2-szer. Így minden él 2-vel járul a BO-hoz és a JO-hoz is. \square

Megj: A DKFL akkor hasznos, ha a SRt gráf lapjairól, a KFL pedig akkor, ha a foksámkokról van információnk.

Fáry-Wagner-tétel: Ha G egyerű SRható gráf, akkor olyan síkbarajzolása is van, amiben minden él egyenes szakasz.

6.2 Az Ezler-féle poliéderformula, síkgráfok karakterizációja



Tétel: Ha G SRt gráf, akkor $n + t = e + k + 1$.

Biz: Rajzoljuk meg G -t az n csúcsból kiindulva, az élek egyenkénti behúzásával. Kezdetben $t = 1, e = 0$ és $k = n$, így a bizonyítandó összefüggés fennáll. Tegyük fel, hogy már néhány élt berajzoltunk, még mindig fennáll az összefüggés, és egy éppen az uv élt rajzolunk meg.

[1.] u és v különböző komponenshez tartoznak. Ekkor k értéke 1-gyel csökken, e -é pedig 1-gyel nő. Az ÉHL miatt nem keletkezik kör, tehát nem zárunk körül új tartományt, vagyis t nem változik. Az összefüggés fennmarad.

[2.] u és v ugyanahhoz a komponenshez tartoznak. Ekkor k nem változik, e viszont 1-gyel nő. Az ÉHL miatt keletkezik kör, tehát kettévágjuk az uv élt tartalmazó korábbi tartományt. Ezért t is 1-gyel nő, az összefüggés ismét fennmarad. \square

Köv: (1) Ha G SRható, akkor t nem függ a síkbarajzolástól.

Biz: $t = e + k + 1 - n$, és a JO nem függ a síkbarajzolástól. \square

(2) **(Euler-formula)** Ha G összefüggő SRt gráf, akkor $n + t = e + 2$

Biz: Mivel G összefüggő, ezért a fenti Tételben $k = 1$. \square

(3) Ha G egyszerű, SRható és $n \geq 3$, akkor $e \leq 3n - 6$.

Biz: Ilyenkor G minden lapját legalább 3 él határolja, így a DKFL miatt $2e = \sum_{i=1}^t l_i \geq 3t$. A Tétel alapján $3n + 2e \geq 3n + 3t = 3e + 3k \geq 3e + 3 + 3 = 3e + 6$, amit rendezve $e \leq 3n - 6$ adódik. \square

(4) G egyszerű, SRható, C_3 -mentes és $n \geq 3 \Rightarrow e \leq 2n - 4$.

Biz: Ilyenkor G minden lapját legalább 4 él határolja. A DKFL miatt $2e = \sum_{i=1}^t l_i \geq 4t$, így $e \geq 2t$. A Tétel miatt $2n + e \geq 2n + 2t = 2e + 2k + 2 \geq 2e + 2 + 2 = 2e + 4$ Ezt rendezve $e \leq 2n - 4$ adódik. \square

(5) Ha G egyszerű, SRható, akkor $\delta(G) \leq 5$ (azaz $\exists v : d(v) \leq 5$).

Biz: A KFL és (3) miatt $\sum_{v \in V(G)} d(v) = 2e \leq 6n - 12$. Ezért van olyan csúcs, amire $d(v) \leq \frac{6n-12}{n} < 6$. \square

(6) A K_5 és $K_{3,3}$ gráfok egyike sem SRható.

Biz: A K_5 gráf egyszerű, de nem teljesül (3), hiszen $|E(K_5)| = \binom{5}{2} = 10 \not\leq 9 = 3 \cdot 5 - 6$. Ezért K_5 nem SRható. A $K_{3,3}$ gráf egyszerű és C_3 -mentes, de nem teljesül rá (4), u.i. $|E(K_{3,3})| = 9 \not\leq 8 = 2 \cdot 6 - 4$. Ezért $K_{3,3}$ nem SRható. \square

Megj: Könnyen látható, hogy ha G SRható, akkor $G + e$ tóruszra rajzolható bármely e él behúzása esetén. Nem nehéz látni, hogy K_6 is tóruszra rajzolható. Sőt: még K_7 is az, de K_8 már nem.

Def: **Élfelosztás:** az élre egy új, másodfokú csúcs ültetése. **Élüsszehúzás:** az él törlése és két végpontjának azonosítása. **Topologikus G (soros bővítés):** G -ből élfelosztásokkal képzett gráf.

Megf: Az éltörlés, csúcsörlés, élfelosztás, élüsszehúzás operációk mindegyike megőrzi a gráf SRható tulajdonságát.

Köv: (1) Top. K_5 top. $K_{3,3}$ nem SRható. (2) Ha G SRható, akkor G -nek nincs se topologikus K_5 , se topologikus $K_{3,3}$ részgráfja.

Kuratowski tétele: (G SRható) \iff (G -nek nincs se topologikus K_5 , se topologikus $K_{3,3}$ részgráfja) **Példa:** Petersen-gráf

6.3 Síkgráfok duálisa

Def: A G SRt gráf **duálisa** a G^* gráf, ha G^* csúcsai G tartományainak, G^* élei G éleinek felelnek meg. Az $uv \in E(G)$ élnek megfelelő duális él az uv él által határolt két tartománynak megfelelő duális csúcsokat köti össze.

Megf: (1) A SRt G gráf G^* duálisa SRható. (n^*, e^*, t^*, k^*) (2) $n^* = t, e^* = e, k^* = 1$. (3) Ha v az i -dik laphoz tartozó duális csúcs, akkor $d_{G^*}(v) = l_i$.

Köv: KFL a duálisra $\sum_{i=1}^t l_i = \sum_{v \in V(G^*)} d_{G^*}(v) = 2e^* = 2e$.

Def: A $Q \subseteq E(G)$ élhalmaz a G gráf **vágása**, ha $G - Q$ szétesik (több komponense van, mint G -nek), de $Q' \subsetneq Q$ esetén $G - Q'$ nem esik szét. **Elvágó él:** egyélű vágás. **Soros él:** kétélű vágás.

Kör-vágás dualitása: Tegyük fel, hogy G^* a G SRt gráf duálisa. Ekkor $(C$ a G köre) $\iff (C^*$ a G^* vágása) ill. $(Q$ a G vágása) $\iff (Q^*$ a G^* köre).

Köv: Hurokél duálisa elvágó él, soros élpáré párhuzamos élpár.

6.4 Whitney

Whitney tétele: Tegyük fel, hogy G^* a G SRt gráf duálisa. Ekkor H pontosan akkor duálisa a G egy alkalmas síkbarajzolásának, ha H előáll G^* -ból a fenti Whitney-operációk alkalmas egymásutánjával.

Def: A $\varphi : E(G) \rightarrow E(H)$ kölcsönös egyenértékű leképezés **kör-vágás dualitás** G és H között, ha C pontosan akkor G köre, ha $\varphi(C)H$ vágása.

Whitney másik tétele: Tegyük fel, hogy G és H között kör-vágás dualitás van. Ekkor G SRható, és H a G egy alkalmas síkbarajzolásának duálisa.

Megj: Egy G gráf által leírt villamos hálózat viselkedését az Ohm-él Kirchhoff-törvények írják le. Ezek a G gráf éleire, köreire és vágásaira vonatkoznak. Ha G és H közt kör-vágás dualitás van, akkor H -n elkészíthető az előző hálózat duálisa. Az eredeti hálózat megoldásában ha az I és U értékeket felcseréljük, az utóbbi hálózat megoldását kapjuk. Whitney másik tétele miatt ez a különös szimmetria csak SRható gráfok által leírt hálózatokon lehetséges.

7 Lineáris egyenletrendszerek