Budapesti Műszaki és Gazdaságtudományi Egyetem

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.

September 30, 2023

Tartalomjegyzék

Oldal

1 A gráfelmélet alapjai

1.1 Mi a gráf?

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

Példa: Ha $V \neq 0$ 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 $\delta(v)$ ill. $\rho(v)$ 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 megszámolva G minden irányított élét pontosan egyszer számoljuk meg. Ezért a kifokok összege az élszám. A belépő éleket leszámlálva hasonló igaz, ezért a befokok összege is az élszám. $\square$

A KFL bizonyítása: Készítsükel a G' digráfot úgy, hogy G minden élét egy oda-vissza irányított élpárral helyettesítjük. 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 kéfogás-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 bizonyíthattuk 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 $\overline{G} = (V, (G), \binom{v}{2} \backslash E(G))$.
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 gárf és a $|V(G)| = n$, akkor $d_G(v) + d_{\overline{G}} = n - 1$ teljesül G bármely v csúcsra.
Biz: A K_n teljes frág minden éle a G és $\overline{G}$ gráfok közül pontosan az egyikhez tartozik. Ezért $d_G(v) + d_{\overline{G}}(v)$ megyegyezik a v csúcs K_n-beli fokszámával, ami $n - 1$. □
Def: A G és G' gráfok akkor izomorfak, ha mindekét gráf csúcsai őgy 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_{42} kör található G-ben, mint G'-ben, stb.

1.5 Gráfoperációk

Def: Éltörlés, csúcstörlés, élhozzáadás.
Def: Feszítő részgráf: éltörlésekkel kapható gráf.
Feszített részgráf: csúcstörlésekkel kapható gráf.
Részgráf: él- és csúcstörlésekkel 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 $E(H)$ a H.
Megj: A gráf definíciója megengedi, hogy a gráf egyik részéből egyáltalán ne vezessen él a gráf maradék részébe, azaz a gráf egyik csúcsából ne lehessen eleken kereszül 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: $(v_1, e_1, v_2, e_2, \ldots, v_k, e_k, v_{k+1})$, ahol $e_i = v_i v_{i+1} \forall i$. (Tkp egyik csúcsból eljutunk egy másik csúcsba mindig élek mentén haladva.)
Séta: olyan élsorozat, amelyikban nincsen ismétlődő él.
Út: olyan séta, amelyikben 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. körről beszélünk.
Megf: G-ben $\exists uv$-út $\Rightarrow$ G-ben $\exists uv$-séta $\Rightarrow$ G-ben $\exists uv$-élsorozat □
Állítás: G-ben $\exists uv$-élsorozat $\Rightarrow$ G-ben $\exists uv$-út □
Def: G ir.tatlan 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 öszefüggőnek, ha G bármely két csúcsa között vezet út G-ben.

Megj: (2) Az előző definíciót irányított fráfokra is kterjeszthető: 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 összefügő**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) $\forall u, v, w \in V(G) : u \sim v \sim w \Rightarrow u \sim w$. □
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 $\forall v, v' \in$ esetén $v \sim v'$.
(2) Minden G irányítatlan gráf csúcshalmaza egyértelműen bomlik fel G komponenseinek diszjunkt uniójára. □
Megj: A G komponense alatt sokszor nem csupán a G csúcsainak egy K részhalmazát, hanem a K által feszített részgráfot értjük.
Megf: G pontosan akkor összefüggő, ha egy komponense van. □
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 keveseb komponense van, mint G-nek.

1.8 Fák és erdők

Def: A körmentes irányítatlan gráfot erdőnek nevezzük. Az öszefüggő, 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 lé zöld: behúzásakor 1-gyel csökken a kmponensek száma. A $\overline{K_n}$ üresgráfnak n komponense van, G-nek pedig k. Ezért pontosan $n - k$ zöld élt kellett behúzni G felépítéséhez. □
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 él 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. □

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. □
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 nincsmás uv-út. □
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. □
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. □
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. □

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 kiszínezettekel, így nem leht 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. □
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. □
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 renszer

Adott egy G gráf és G-nek egy Frögzített 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 $e \in E(G) \backslash E(F)$ éléhez tarozó alapkör pedig az $F + e$ köre.
Megf: Tfh $f \in F$ és $e \in E(G) \backslash E(F)$. Ekkor ($F - f + e$ ffa) $\iff$ (f benne van e alapkörében) $\iff$ (e benne van f alap vágásában).
Köv: Az $e \in E(G) \backslash E(F)$ 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 $k : E \to \mathbb{R}_+$ 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 élk 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 építésekor költség szerint növekvő sorrendben döntsünk az élekről, hátha mkffát kapunk a végén.
Kruskal-algoritmus: Input: $G = (V, E)$ és $k : E \to \mathbb{R}_+$ költségfüggvény. Output: $F \subseteq E$
Működés: Tfh $k(e_1) \leq k(e_2) \leq \cdots \leq k(e_m)$, ahol $E = \{e_1, e_2, \dots, e_m\}$. Legyen $F_0 = 0$, és $i = 1, 2, \dots, m$-re

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

2.3 Minimális költségű feszítőfák struktúrája

$G = (V, E)$ gráf és $k : E \to \mathbb{R}_+$ 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 futtotott 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 $F = \{f_1, f_2, \ldots, f_l\}, k(f_1) \leq k(f_2) \leq \cdots \leq k(f_l)$ és $F \cap E_c$ a G_c egy feszítő erdeje $\forall c \geq 0$-ra. Tfh $F' = \{f'_1, f'_2, \ldots, f'_l\}$ a G egy feszítő erdejének élei, és $k(f'_1) \leq k(f'_2) \leq \cdots \leq k(f'_l)$. 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 $f'_1, f'_2, \ldots, f'_i$ élek is mind E_c-beliek, és többen vannak az $E_c \cap F$ feszítő erdő élszámánál. Tehát $f'_1, f'_2, \ldots, f'_i$ nem lehet körmentes, így $f'_1, f'_2, \ldots, f'_l$ sem. Ez ellentmondás. Tehát $k(f_i) \leq k(f'_i)\, \forall i$. Ezért $k(F) = \sum_{i=1}^{l} k(f_i) \leq \sum_{i=1}^{l} k(f'_i) = k(F')$. □
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. □
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 $k(k_j) \leq k(f'_i)$. Innen $k(F) < k(F')$. □
Köv: (3) Ha a G gárf 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 ✓, Output ✓, Működés ✓, Helyesség ✓, 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 buborékrendezés legfeljebb $\binom{m}{2}$ összehasonlítást használ.
1. n csúcsú G gráf esetén egy élről döntés megoldható $konst \cdot \log_2 n$ lépésben az adatstruktúra karbantartásával együtt is. Az összes döntéshez tehát elegendő $konst \cdot n \cdot \log_2 n$ lépés. A Kruskal-algoritmus lépésszáma ezért felülről becsülhető $konst \cdot (n + m) \cdot \log_2(n + m)$-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 → elért → 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, ($v \in V$ gyökérpont[1]).
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 őszből leszármazottba, a visszaél leszármazottból ősbe 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 fejezük be, mint v_j-t, továbbá v_i gyerekei megleőzik 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_iv_j faél, akkor v_kv_l nem lehet gráfél.
Biz: Ha $v_kv_l \in E(G)$, 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_iv_j előreél lenne, akkor v_i-ből v_j-be irányított út vezetne a BFS-fában, és v_iv_j ennek a faélekből álló útnak az utolsó élét átugraná. $\square$
(5) Ha a BFS-fában k-élű irányított út vezet u-ból v-be, akkor G-ben nincs k-nál kevesebb élű uv-út.

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

Biz: Ha lenne a BFS fa-beli útnál kevesebb elű út G-ben, akkor lenne olyan gráfél, ami faélt ugrik át. □
(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 $l : E(G) \to \mathbb{R}$ 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: $dist_l(u, v) := \min\{l(P) : P\ uv\text{-út}\}$ ($\nexists uv$-út$\Rightarrow$ $dist_l(u, v) = \infty$.) Az l hosszfüggvénye nemnegatív, ha $l(e) \geq 0$ teljesül minden e élre. Az l hosszvüggvény 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, $l : E(G) \to \mathbb{R}$ hosszfüggvény és $r \in V(G)$. (r, l)-felső becslés olyan $f : V(G) \to \mathbb{R}$ 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: $f(v) = dist_l(r, l)\ \forall v \in V(G)$.

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 $l : E(G) \to \mathbb{R}$ 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 élhosszfüggvény 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. □
(2) $f(r, l)$-felső becslés (pontosan) $\iff$ (f-en $\nexists$ 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 legrövidebbb 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. □
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 $l: E(G) \to \mathbb{R}$ 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-en $\nexists$ érdemi élmenti javítás).
Dijkstra-algoritmus: Input: $G = (V,E), l: E \to \mathbb{R}_+, r \in V$. Output: $dist_l(r,v)\forall v \in V$
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: $G = (V,E), l: E \to \mathbb{R}_+, r \in V$. Output: $dist_l(r,v)\forall v \in V$
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 hozza 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 Dijsktra-algoritmus helyes, akkor az algoritmus végén a megjelölt élek egy legrövidebb utak fáját alkotják r gyökérrel.

3.6 Dijkstra helyessége

Megf: Tfh $u_1, u_2, \ldots, u_n$ 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) $f_{|V|}(u_1) \leq f_{|V|}(u_2) \leq \cdots \leq f_{|V|}(u_n)$
(3) A Dijsktra-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 váltorott, azaz $f_{|V|}(u_i) = f_i(u_i)$. A másik (r,l)-felső becslés pedig csak tovább csökkenhetett a késpbbi 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 Dijsktra-algoritmus helyesen működik, azaz G minden csúcsára igaz, hogy $dist(r,v) = f_{|V|}(v)$.

Biz: A Dijsktra-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 $f_{|V|}(v) = dist_l(r,v) \forall v \in V(G)$. $\square$

,,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 $konst \cdot n^2$ lépést igényel. Ezen kívül legfeljebb m élmenti javítást véges, ami $konst' \cdot m$ lépés. Összességében tehát legfeljebb $konst'' \cdot (n^2 + m)$ lépésre van szükség, az algoritmus hatékony.

4 Legrövidebb utak, DFS, PERT

5 Euler-séták és Hamilton-körök

6 Síkgráfok

7 Lineáris egyenletrendszerek

8 Az $\mathbb{R}^n$ tér alaptulajdonságai

9 Altér bázisa és dimenziója

10 Négyzetes mátrix determinánsa

11 Mátrixműveletek és lineáris leképezések

12 Mátrix rangja és inverze

13 Mátrixegyenletek