A számítástudomány alapjai 2020. I. félév

4. gyakorlat Összeállította: Fleiner Tamás (`fleiner@cs.bme.hu`)

Tudnivalók

Def: Adott $G = (V, E)$ (ir) gráf és egy $\ell : E \to \mathbb{R}$ élhosszfv. Egy G-beli (ir) út hossza az út éleinek összhossza, $dist(u, v)$ pedig az (ir) uv-utak közül a legrövidebb hosszát jelöli. Az ℓ hosszfv *konzervatív*, ha nincs G-ben negatív összhosszúságú (ir) kör.

Def: Adott $G = (V, E)$ (ir) gráf, $r \in V$ és egy $\ell : E \to \mathbb{R}$ élhosszfv. A $d : V \to \mathbb{R}$ függvényt *(r, ℓ)-felső becslés*nek nevezzük, ha $d(r) = 0$ és $d(v) \geq dist(r, v)$ teljesül G minden v csúcsára. Az *$e = vw$ él menti javítás* esetén a $d(w)$ értéket a $\min\{d(w), d(v) + \ell(vw)\}$ értékkel helyettesítjük.

Megfigyelés: (1) Ha ℓ konzervatív, akkor tetsz. (r, ℓ)-f.b. élmenti javítása (r, ℓ)-f.b.-t ad.
(2) Ha élmenti javítás nem tud változtatni egy d (r, ℓ)-felső becslésen, akkor $d(v) = dist(r, v)$ $\forall v \in V$.

Dijkstra-algoritmus Input: $G = (V, E)$ (ir) gráf, $\ell : E \to \mathbb{R}_+$ nemneg hosszfv, $r \in V$ gyökér. Output: $dist(r, v)$ minden $v \in V$-re. Működés: Kezdetben $U_0 := \emptyset$, $d(r) = 0$ és $d(v) = \infty$ ha $v \neq r$. Az algoritmus i-dik fázisában ($i = 1, 2, \ldots, |V|$) a következő történik.
1. Legyen u_i az a v csúcs a $V \setminus U_{i-1}$ halmazból, amelyre $d(v)$ minimális és legyen $U_i := U_{i-1} \cup \{u_i\}$.
2. Végezzünk él menti javításokat minden U_i-ből kivezető $u_i x$ élen.
Az output a $|V|$-dik fázis utáni d függvény. Szokás megjelölni a végső $d(v)$ értékeket beállító éleket.

Megfigyelés: Ha az output a d (r, ℓ)-felső becslés, akkor (1) $d(u_i) \leq d(u_{i+1})$ $\forall 1 \leq i < n$-re (2) $d(u_1) \leq d(u_2) \leq \ldots \leq d(u_n)$, valamint (3) Élmenti javítás nem változtat d-n.

Köv.: (1) A Dijkstra-algoritmus helyesen működik, azaz $dist(r, v) = d(v)$ $\forall v \in V$ teljesül.
(2) Az algoritmus során megjelölt élek egy legrövidebb utak fáját alkotják G-ben: az r gyökérből minden r-ből elérhető csúcshoz vezet olyan legrövidebb út is, ami csak megjelölt éleket tartalmaz.
(3) A Dijkstra-algoritmus lépésszáma legfeljebb $konst \cdot n^2$, ahol $n = |V|$.

Ford-algoritmus Input: $G = (V, E)$ (ir) gráf, $\ell : E \to \mathbb{R}$ konzervatív hosszfv, $r \in V$ gyökérpont. Output: $dist(r, v)$ minden $v \in V$-re. Működés: Legyen $E = \{e_1, e_2, \ldots, e_m\}$. Kezdetben legyen $d(r) = 0$ és $v \neq r$ esetén $d(v) = \infty$. Az i-dik fázis $i = 1, 2, \ldots, n-1$ esetén abból áll, hogy elvégezzük az $e_1, e_2, \ldots, e_m$ élek menti javításokat. A végén az output $dist(r, v) = d(v)$ minden v-re.

Állítás: (1) A Ford-algoritmus i-dik fázisa után $dist(r, v) = d(v)$ minden olyan v-re, ahova van legfeljebb i élű legrövidebb út v-ből. (2) A Ford-algoritmus lépésszáma legfeljebb $konst \cdot n^3$.
(3) Ahogy Dijkstra esetén, itt is legrövidebb utak fáját alkotják a végső $d(v)$ értékeket beállító élek.

Floyd-algoritmus Input: $G = (V, E)$ (ir) gráf, $\ell : E \to \mathbb{R}$ konz. Output: $dist(u, v)$ $\forall u, v \in V$. Működés: Legyen $V = \{v_1, v_2, \ldots, v_n\}$ és $d^{(k)}(i, j)$ a legrövidebb olyan $v_i v_j$ út hossza, aminek belső pontjai csak $v_1, v_2, \ldots v_k$ lehetnek. Kezdetben $d^{(0)}(i, j) = \ell(v_i, v_j)$, ha $v_i v_j \in E$, különben $d^{(0)}(v_i, v_j) = \infty$. A k-dik fázisban

$$d^{(k)}(i, j) = \min\{d^{(k-1)}(i, j), d^{(k-1)}(i, k) + d^{(k-1)}(k, j)\} \tag{1}$$

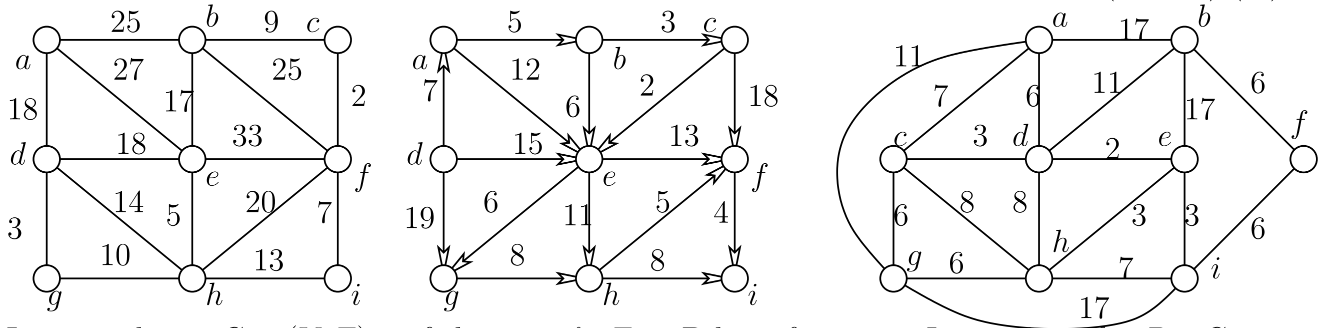
alapján a $d^{(k)}$ függvényt határozzuk meg. Az n-dik fázis után $dist(v_i, v_j) = d^{(n)}(i, j)$ az output.

Állítás: Az (1) fennáll, tehát a Floyd-algoritmus helyes. Lépésszáma pedig legfeljebb $konst \cdot n^3$.

Gyakorlatok

1. Rajzoljunk gráfot, és keressük meg egy csúcsból kiindulva a BFS fáját. Megfelelő élhosszok megadásával gyakoroljuk a Dijkstra-, Ford- és Floyd-algoritmusokat. Bátran használjuk ehhez a túloldali gráfokat. (✓)
2. Legyen $G = (V, E)$ (irányított) gráf, $\ell : E \to \mathbb{R}_+$ nemnegatív élhosszfüggvény és legyenek u, v, w a G csúcsai. Igazak-e az alábbi állítások? (1) Ha P a G egy legrövidebb uv útja és w csúcsa P-nek, akkor a P út u-tól w-ig tartó ill. w-től v-ig tartó részei a G egy legrövidebb uw- ill. wv-útját alkotják. (2) Ha P_1 és P_2 a G egy legrövidebb uw- ill. wv-útja, akkor a P_1 és P_2 egymás után fűzése a G egy legrövidebb uv-útja lesz. (✓) És ha ℓ konzervatív?
3. Tervezzünk csavaranyákból és cukorspárgából gravitációs elven működő mechanikus számítógépet, ami alkalmas az inputként megadott, nemnegatív élhosszokkal rendelkező irányítatlan gráf tetszőleges gyökérpontjából a többi csúcs távolságának a meghatározására. (!)
4. Legyen $V(G) = \{v_3, v_4, \ldots, v_{10}\}$, és $v_i v_j \in E(G)$, ha i és j nem relatív prímek, azaz van 1-nél nagyobb közös osztójuk. Legyen a $v_i v_j$ él hossza $\min(i, j) - 1$. Határozzunk meg a v_5 csúcsból minden más csúcsba egy-egy legrövidebb utat, ha van. (pZH '14) (✓)

5. A bal oldali ábrán látható gráf éleire írt számok az adott él hosszát jelentik. Órán tanult módszer felhasználásával határozzunk meg minden e-től különböző v csúcsra egy legrövidebb ev utat. (ZH '16) (✓)



6. Legyen adott a $G = (V, E)$ gráf élein egy $\ell : E \to \mathbb{R}$ hosszfüggvény. Igaz-e, hogy ha P a G egy legrövidebb uv-útja az ℓ hosszfüggvényre, akkor P egyúttal legrövidebb út az ℓ' hosszfüggvényre is, ahol $\ell'(e) = \ell(e)^2$ teljesül G minden e élére? (ppZH '14)
7. Adott egy $G = (V, E)$ gráf, egy $\ell : E \to \mathbb{R}_+$ hosszfüggvény, valamint egy r gyökérpont. Egyetlen Dijkstra-algoritmus lefuttatása segítségével találjuk meg G mindazon e éleit, amelyekre igaz az, hogy önmagában attól, hogy e hosszát eggyel csökkentjük egyetlen csúcs r-től mért távolsága sem csökken.
8. Adott egy $G = (V, E)$ gráf, egy $\ell : E \to \mathbb{R}_+$ élhosszfüggvény valamint egy $e = uv \in E$ él. Javasoljunk gyors eljárást annak a maximális λ értéknek a meghatározására, amennyivel G két csúcsának a távolsága megnövekszik akkor, ha töröljük az e élt G-ből.
9. Bizonyítsuk be, hogy a Ford-algoritmus minden gráf és minden konzervatív élhosszfüggvény esetén futtatható úgy, hogy a második fázisban a d felső becslés már ne változzon. Bizonyítsuk be azt is, hogy ha minden legrövidebb rv-útnak legalább k éle van, akkor G éleinek van olyan sorrendje, hogy ezzel a sorrenddel a Ford-algoritmusnak legalább k fázisra van szüksége a végső d felső becslés megtalálásához. (*)
10. Forintot szeretnénk különféle valutákra átváltani. Külföldön élő ismerőseink révén nem csak forintot, hanem számos más valutát is közvetlenül át tudunk váltani bizonyos valutákra. A cél, hogy esetleg ilyen átváltások felhasználásával minél jobb árfolyamot érjünk el a forintunk konverziója során. E célból elkészítettünk egy irányított gráfot, aminek a csúcsai az egyes valutáknak, az élek pedig az egyes közvetlen tranzakcióknak felelnek meg. Minden uv élhez ismert az adott váltásnál alkalmazott árfolyam, azaz, hogy hány egységet kell fizetnünk az u pénznemben a v pénznem egy egységéért. Adjunk hatékony módszert arra, hogy meghatározzuk, legfeljebb mennyit kaphatunk az egyes valutákból 1 Ft-ért, ill. határozzuk meg azt is, milyen átváltásokat kell ehhez végeznünk. (!)
11. Adott $n \times k$ méretű táblázat minden mezőjében 0 vagy 1 áll. Találjunk a táblázat bal felső sarkától a jobb alsó sarokig egy mezőhatárok mentén jobbra és lefelé haladó olyan utat, amire igaz, hogy a vonal alatti 1-esek és a vonal feletti 0-k számának összege a lehető legkisebb. Hogyan érdemes eljárni? (*) (Hint: alighanem vmiféle gráfban kéne legrövidebb utat keresni.)
12. Legyen $G = (V, E)$ irányított gráf, $\ell : E \to \mathbb{R}_+$ nemnegatív élhosszfüggvény, $e = uv$ pedig G egy éle. Javasoljunk gyors eljárást, amelynek segítségével megtalálhatók G mindazon w csúcsai, amelyek számára *fontos* ill. *hasznos* az e él. (*) (Az e él akkor fontos w számára, ha van olyan t csúcs G-ben, amelynek a w-től mért távolsága növekszik az $\ell(e)$ bármilyen kis mértékű növelése nyomán. Az e él akkor hasznos w számára, ha van olyan t csúcs, amelynek a w-től mért távolsága csökken, az $\ell(e)$ bármilyen kis mértékű csökkentése nyomán.)
13. Hogyan lehet 3 buszjegy birtokában a lehető leggyorsabban eljutni a város egyik pontjából a másikba? Formálisan: Adott a $G = (V, E)$ irányított gráf, G éleinek egy F részhalmaza, G egy v csúcsa, valamint egy k pozitív egész szám. Tegyük fel, hogy $t(e)$ jelöli azt, hogy mennyi ideig tart az e él mentén történő áthaladás. Javasoljunk gyors eljárást, amely a G minden u csúcsára meghatározza G egy olyan vu-útját, ami legfeljebb k db F-beli élt tartalmazza, és ami az ilyen utak között a lehető leggyorsabban bejárható, azaz amire az út éleihez rendelt t értékek összege minimális. (*)