A számítástudomány alapjai

Katona Gyula Y.

Számítástudományi és Információelméleti Tanszék
Budapesti Műszaki és Gazdaságtudományi Egyetem

Legszélesebb utak

Legszélesebb utak

Definíció

Jelöljük gráf e élének szélességét $w(e)$-vel. Legyen a gráf egy P útjának szélessége az úton található legkisebb szélességű él szélessége, azaz $w(P) = \min_{e \in P}\{w(e)\}$.

Feladat

- *Keressük meg egy gráf s pontjából a többi pontjába a legszélesebb utakat.*
- *Keressük meg egy gráf bármely két pontja között a legszélesebb utakat.*

Például egy számítógép hálózatban keressük a legnagyobb sávszélességű összeköttetést.

Legszélesebb utak keresése irányítatlan gráfban

Módosítsuk a Kruskal algoritmust: Minden lépésben a legszélesebb olyan élet választjuk, ami még nem alkot kört a már korábban kiválasztottakkal.

Tétel

Egy összefüggő gráfban az így kapott feszítőfa a gráf bármely két pontja között egy legszélesebb utat határoz meg.

Bizonyítás.

Jelöljük F-el az algoritmus által adott fát.Indirekt tegyük fel, hogy valamely s, t pontokra van olyan P-vel jelölt $s-t$ út, amelyik szélesebb az F-beli $s-t$ útnál.Legyen e az F-beli $s-t$ út egy minimális szélességű éle.Mivel P szélesebb $w(e)$-nél, ezért P minden éle is szélesebb $w(e)$-nél.

Legszélesebb utak keresése irányítatlan gráfban

Bizonyítás.

Ha F-ből elhagyjuk e-t, két komponensre esik, s az egyik, t a másik komponensbe esik. A P útnak van olyan e' éle, ami $F-e$ két komponense között megy. Ekkor $w(e') > w(e)$.
Amikor az algoritmus az e élet bevette a fa élei közé, akkor e'-t kellett volna választania, hiszen az sem alkothatott kört az F korábban kiválasztott éleivel, de nagyobb a szélessége. □

Megjegyzés

Az így konstruált fa egyébként egy maximális össz-szélességű (=max. súlyú) feszítőfa is egyben.

Legszélesebb utak keresése irányított gráfban

Módosítsuk a Dijkstra algoritmust:

- Minden pontra nyilvántartjuk az addig megtalált legszélesebb út szélességét: $w(v)$
- Kezdetben: $w(s) = \infty$ és minden $v \neq s$-re $w(v) = 0$.
- Javítás az $e = (a, b)$ élen: Ha $\min(w(a), w(e)) > w(b)$, akkor legyen $w(b) = \min(w(a), w(e))$.
- u_0 kiválasztása: Válasszuk a T-ből azt az u_0 pontot, amire $w(u_0)$ maximális.

Legszélesebb utak keresése irányított gráfban

Állítás

Az így kiválasztott u_0-ra $w(u_0)$ a legszélesebb út szélessége lesz.

Bizonyítás.

A bizonyítás ugyanúgy működik, mint a Dijkstra algoritmus bizonyítása. Ehhez elég, hogy az út szélesség definíciója rendelkezik a következő 2 tulajdonsággal:

- Egy út egyik részútja sem lehet kevésbé széles az egész út szélességénél.
- Ha az út egy részét (pl. az elejét) keskenyebbre cseréljük, akkor az egész út szélessége nem növekedhet.

□

Legszélesebb legrövidebb utak

Feladat

Keressük a legrövidebb utat, de ha több legrövidebb is van, akkor azok közül a legszélesebbet.

Módosítsuk a Dijkstra algoritmust:

- Minden pontra nyilvántartunk egy rendezett párt: $(d(v), w(v))$, ahol $d(v)$ az eddig megtalált legrövidebb út hossza, $w(v)$ pedig az ilyen rövidek közül az eddig megtalált legszélesebb út szélessége.
- Kezdetben: $(d(s), w(s)) = (0, \infty)$ és minden $v \neq s$-re $(d(v), w(v)) = (\infty, 0)$.

Legszélesebb legrövidebb utak

- Javítás az $e = (a, b)$ élen: Ha $d(a) + d(e) < d(b)$, akkor $d(b) = d(a) + d(e)$, $w(b) = \min(w(a), w(e))$.
  Ha $d(a) + d(e) = d(b)$, akkor $d(b)$ nem változik, de ha $w(b) < \min(w(a), w(e))$, akkor $w(b) = \min(w(a), w(e))$.
  Más esetben nincs változás.
- u_0 kiválasztása: Válasszuk a T-ből azt az u_0 pontot, amire $(d(u_0), w(u_0))$ lexikografikus értelemben minimális, azaz elsősorban a d értékek nagysága dönt (a kisebbet választjuk), ha azok egyenlőek, akkor a w érték dönt (a nagyobbat választjuk).

Legrövidebb legszélesebb utak

Feladat

Keressük a legszélesebb utat két adott pont között, de ha több legszélesebb is van, akkor azok közül a legrövidebbet.

Sajnos itt nem működik a Dijkstra algoritmus módosítása.
Ugyanis egy részút lehet lehet szélesebb, mint az egész út, viszont ezen részúton az egész út szélességének megfelelőek közül kellene a legrövidebbet nyilvántartani, nem a részút szélességének megfelelőek közül.

Legrövidebb legszélesebb utak

Algoritmus

- Valamely adott w élszélesség esetén a w-nél keskenyebb éleket elhagyva, BFS-el eldönthetjük, hogy a maradék gráfban van-e út a két pont között.
- Bináris kereséssel meghatározzuk, hogy mi az a legnagyobb szélesség, amikor még van ilyen út, ez megadja a legszélesebb út szélességét.
- Ebben a gráfban (a keskenyebb élek elhagyása után), Dijkstra algoritmusával megkeressük a legrövidebb ilyen szélességű utat.

Ez az algoritmus lassabb, mint a Dijkstra algoritmusa és csak két adott pont között keresi meg az utat.

Legrövidebb legszélesebb utak

Egy gyakran előforduló speciális eset, amikor az élek hossza egységesen 1, azaz a legszélesebb utak közül a legkevesebb élből állót keressük.

Ilyenkor használhatjuk a Dikstra-ához hasonlóan módosított Ford algoritmust az élszélességekkel. Ez a k-adik körben meghatározza a legfeljebb k élből álló utak közül a legszélesebbet, amiből már könnyen megkapható az eredmény.