Maximális folyamok

Előadás összefoglaló, 2012. február 20.

Kovács Péter
kpeter@inf.elte.hu

1. Feladat

Adott egy $G = (V, E)$ irányított gráf, az élein egy $c : E \to \mathbb{R}_0^+$ *kapacitásfüggvény*, valamint egy $s \in V$ *termelő* és egy $t \in V$ *fogyasztó* csúcs ($s \neq t$).

Definíció. Egy $f : E \to \mathbb{R}_0^+$ függvényre vezessük be az alábbi jelöléseket, amelyek egy adott csúcsból *kifolyó* és a csúcsba *befolyó folyam* mennyiségét adják meg.

$$f_{ki}(v) := \sum_{(v,w)\in E} f(v,w)$$

$$f_{be}(v) := \sum_{(u,v)\in E} f(u,v)$$

Definíció. Hasonlóan definiáljuk egy $f : E \to \mathbb{R}_0^+$ függvény esetén egy $W \subseteq V$, $s \in W$, $t \notin W$ halmazból *kifolyó* és a halmazba *befolyó folyam* mennyiségét az alábbi módon.

$$f_{ki}(W) := \sum_{(u,v)\in E\,:\,u\in W,\ v\notin W} f(v,w)$$

$$f_{be}(W) := \sum_{(u,v)\in E\,:\,u\notin W,\ v\in W} f(u,v)$$

Definíció. Egy $f : E \to \mathbb{R}_0^+$ függvényt *folyamnak* (vagy s–t folyamnak) nevezünk, ha teljesül az alábbi két feltétel:

(1) *kapacitásfeltétel*: $\forall (u,v) \in E : 0 \le f(u,v) \le c(u,v)$;

(2) *folyammegmaradás*: $\forall v \in V \setminus \{s,t\} : f_{ki}(v) = f_{be}(v)$.

Definíció. Egy f *folyam nagysága* az $\|f\| := f_{ki}(s) - f_{be}(s)$ mennyiség.

Feladat. A *maximális folyam* feladat az, hogy a G gráfban adott c kapacitásfüggvény mellett keressünk egy f folyamot, amelyre $\|f\|$ maximális.

2. Algoritmus

A feladat megoldására számos algoritmus létezik. A legalapvetőbb az alábbiakban bemutatott *Ford–Fulkerson-algoritmus*, valamint annak egy változata, az *Edmonds–Karp-algoritmus* [4].

INPUT: $G = (V, E)$ irányított gráf; $c : E \to \mathbb{R}_0^+$ kapacitásfüggvény; $s, t \in V$, $s \neq t$

procedure FORD–FULKERSON-ALGORITMUS
    $G' := (V, E \cup E^*)$ irányított gráf építése, ahol E^* a megfordított élek halmaza,
        ahol $e = (u, v) \in E$ megfordítottja $e^* = (v, u) \in E^*$ és $(e^*)^* = e$
    for all $e \in E$ **do** ▷ *reziduális kapacitások inicializálása*
        $r(e) := c(e)$
        $r(e^*) := 0$
    end for
    while G'-ben létezik P irányított út s-ből t-be, amelyre $\forall e \in P : r(e) > 0$ **do**
        P irányított *javító út* keresése G'-ben s-ből t-be ($\forall e \in P : r(e) > 0$)
        $\delta := \min\{r(e) : e \in P\}$ ▷ *δ a min. reziduális kapacitás a P úton*
        for all $e \in P$ **do** ▷ *δ egységnyi folyam küldése a P úton*
            $r(e) := r(e) - \delta$
            $r(e^*) := r(e^*) + \delta$
        end for
    end while
end procedure

OUTPUT: $\forall e \in E : f(e) := r(e^*)$ módon definiált $f : E \to \mathbb{R}_0^+$ függvény egy maximális s–t folyam

Az *Edmonds–Karp-algoritmus* csak annyiban különbözik Ford és Fulkerson általános módszertől, hogy megszorítást tesz a javító utak választására: minden iterációban válasszuk az egyik legrövidebb (legkevesebb élből álló) javító utat.

Megjegyezzük, hogy a Ford–Fulkerson-algoritmus legtermészetesebb megvalósítása, amelyben szélességi kereséssel keresünk javító utakat, éppen az Edmonds–Karp-algoritmust adja, így az semmi további nehézséget nem jelent, ugyanakkor jobb futásidőt garantál.

2.1. Helyesség

Definíció. Egy $W \subseteq V$ halmazt *vágásnak* nevezünk, ha $s \in W$ és $t \notin W$. Egy W *vágás nagysága*, illetve *kapacitása* pedig a vágásból kilépő élek kapacitásának összege:

$$c_{ki}(W) := \sum_{(u,v) \in E \,:\, u \in W,\; v \notin W} c(u, v).$$

Lemma (folyammegmaradás vágásokra). Minden f folyam és $W \subseteq V$ vágás esetén

$$f_{ki}(W) - f_{be}(W) = \|f\|.$$

Bizonyítás (vázlat). Egyszerű meggondolással adódik, hogy

$$f_{ki}(W) - f_{be}(W) = \sum_{x \in W} (f_{ki}(x) - f_{be}(x)),$$

amiből pedig a (2) folyammegmaradási feltételt alkalmazva megkapjuk a lemma állítását. •

A *maximális folyam* és *minimális vágás* feladatok *primál-duál* kapcsolatát, valamint a Ford–Fulkerson-algoritmus helyességét az alábbi tétel fogalmazza meg.

Maximális folyam – minimális vágás tétel. *$G = (V, E)$ irányított gráf, $c : E \to \mathbb{R}_0^+$, $s, t \in V$, $s \neq t$ esetén a maximális s–t folyam nagysága megegyezik a minimális s–t vágás méretével. A bevezetett jelöléseinkkel:*

$$\max_{f \text{ folyam}} \|f\| = \min_{W \text{ vágás}} c_{ki}(W).$$

Ha a c kapacitásfüggvény egészértékű, akkor a maximális folyam is választható egészértékűnek.

Bizonyítás (vázlat). Egyrészt tetszőleges f folyam és W vágás esetén $\|f\| \leq c_{ki}(W)$, hiszen a folyammegmaradási lemma miatt $\|f\| = f_{ki}(W) - f_{be}(W)$, továbbá az (1) kapacitásfeltétel miatt $f_{ki}(W) \leq c_{ki}(W)$ és $f_{be}(W) \geq 0$. Másrészt belátható, hogy a Ford–Fulkerson-algoritmus megad egy f folyamot és egy W vágást, amelyekre $\|f\| = c_{ki}(W)$. Ekkor az előző megfigyelés miatt f szükségszerűen maximális folyam, W pedig minimális vágás, ezzel pedig a tétel helyességét is igazoltuk.

Tekintsük a Ford–Fulkerson-algoritmus végső állapotát, vagyis amikor már nem találtunk javító utat. Jelöle W a G' gráfban s pontból $r > 0$ éleken elérhető pontok halmazát:

$$W = \{v \in V : \exists P \text{ irányított út } G'\text{-ben } s\text{-ből } v\text{-be, amelyre } \forall e \in P : r(e) > 0\}.$$

Mivel az algoritmus leállt, $t \notin W$, tehát W egy vágás. Az f folyamot pedig úgy adja meg az algoritmus, hogy minden $e \in E$ élre: $r(e) = c(e) - f(e)$ és $r(e^*) = f(e)$.

W-ből definíció szerint nem vezet ki $r > 0$ él G'-ben, tehát minden $(u, v) \in E$ eredeti (G-beli) kimenő élére $f(u, v) = c(u, v)$, továbbá minden $(u, v) \in E$ eredeti (G-beli) bejövő élére $f(u, v) = 0$ ($r(e^*) = 0$ miatt). Tehát $f_{ki}(W) - f_{be}(W) = c_{ki}(W)$, és a lemma miatt $\|f\| = f_{ki}(W) - f_{be}(W)$, tehát $\|f\| = c_{ki}(W)$. •

Megjegyzés. A maximális folyam – minimális vágás tétel a lineáris programozásban ismert *erős dualitás tétel* speciális esete, a tétel egyszerűbben belátható része ($\|f\| \leq c_{ki}(W)$ minden f folyamra és W vágásra) pedig a *gyenge dualitás tételnek* felel meg.

2.2. Műveletigény

Az általános Ford-Fulkerson-algoritmus futási idejére nem lehet erősen polinomiális korlátot adni, vagyis olyan polinomiális korlátot, amely csak a gráf méretétől függ, a kapacitásfüggvénytől nem. Belátható azonban, hogy az Edmonds–Karp-algoritmus futási ideje $O(ne(n + e))$, ezért ezt a változatot érdemes alkalmazni.

Hivatkozások

[1] Thomas H. Cormen, Charles E. Leiserson, Ronald L. Rivest, Clifford Stein. *Új algoritmusok*. Scolar, 2003.

[2] Rónyai Lajos, Ivanyos Gábor, Szabó Réka. *Algoritmusok*. TypoTeX, 1998.

[3] Ravindra K. Ahuja, Thomas L. Magnanti, James B. Orlin. *Network Flows: Theory, Algorithms, and Applications*. Prentice-Hall, Inc., February 1993.

[4] Lester R. Ford and Delbert R. Fulkerson. *Flows in Networks*. Princeton University Press, 1962.