Lineáris programozás

munkák :

fordítók :

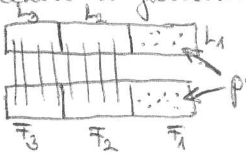
← ezek a párosítások lehetségesek

Cél: a lehető legtöbb munkát végezzük el

Mo.: maximális párosítás keresése

A szaggatott éleket vesszük be a párosításba a másik 2 már behúzott él helyett, így nő a párosítások száma. – javítóutas algoritmus

Ha leállt a javítóutas algoritmus, akkor:



L_2: eljuthatunk ide F_1-ből alternáló úton keresztül

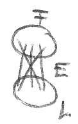
F_1 és F_2 párjai a max. párosításban $= L_2$

$N(F_1 \cup F_2) = L_2$

Tehát tényleg max. párosítást találtunk.

Adott: egy páros gráf : $G(F, L, E)$

$w : E \to \mathbb{R}$ (minden élhez rendel egy számot)



Keressük: M egy párosítás

$\max_M \sum_{e \in M} w(e)$ (maximalizáljuk az M-beli élek összsúlyát)

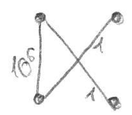
} optimum assignment

Adott: $G(F, L, E)$ (feltesszük, hogy G-ben ∃ teljes párosítás)

$w : E \to \mathbb{R}$

Keressük: M egy teljes párosítás

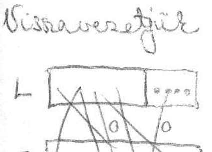
$\max_M \sum_{e \in M} w(e)$ (maximalizáljuk az M-beli élek összsúlyát)



max. párosítás: 10^6

max. teljes párosítás: 2

Visszavezetjük az 1. feladatot a 2. feladatra:



A kisebb ponthalmazt kiegészítjük annyi ponttal, hogy a 2 halmazban ua.-an legyenek.

Behúzunk 0 súlyú éleket, ahol csak lehet, amik még nincsenek összekötve.

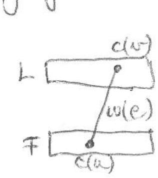
A negatív súlyú éleket már az elején töröljük.

Hogyan bizonyítjuk be, hogy amit találtunk, az tényleg max. teljes párosítás?

Def: Címkézés:

$c: F \cup L \to \mathbb{R}$

$\forall uv \in E$-re $c(u)+c(v) \geq w(e)$



$$\sum_{e=uv\in M} w(e) \leq \sum_{e=uv\in M} c(u)+c(v) = \sum_{v\in F\cup L} c(v)$$

↑ teljes párosítás

Lemma: M teljes párosítás; c címkézés. Ha $\forall e\in M$-re $c(u)+c(v)=w(e)$, akkor M max. összsúlyú teljes párosítás.

Egerváry-algoritmus:

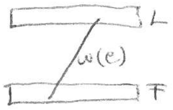
Nyilvántartjuk: M párosítás
c címkézés

Az $e=uv$ élet piros élnek nevezzük, ha $c(u)+c(v)=w(e)$.

M minden éle piros legyen.

Cél: M teljes párosítás

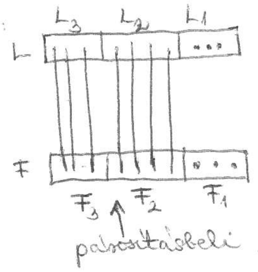
0. lépés: $M=\emptyset$



$$c(v)=\begin{cases} 0, \text{ ha } v\in L \\ \max_{t\in L} w(vt), \text{ ha } v\in F \end{cases}$$

1. lépés: M-ből indulva a piros részgráfban max. párosítás legyen M'.
Ha M' teljes párosítás, akkor kész vagyunk és kiadjuk M'-t.

2. lépés:



Mivel M' maximális, ezért (készítsük el az előző felosztást) $F_1\cup F_2$ és $L_1\cup L_3$ között nincs piros él.

$$\delta = \min_{\substack{u\in F_1\cup F_2 \\ v\in L_1\cup L_3}} \{c(u)+c(v)-w(e) \mid uv=e\in E\} > 0$$

Elkészítünk egy új címkézést a régi címkézés segítségével.

$$c'(v)=\begin{cases} c(v)-\delta, \text{ ha } v\in F_1\cup F_2 \\ c(v)+\delta, \text{ ha } v\in L_2 \\ c(v), \text{ egyébként} \end{cases}$$

Visszatérünk az 1. lépésre M'-vel és c'-vel.

Áll: c' is címkézés

biz: $c(u)+c(v) \geq w(e)$

$\underset{uv\in E}{c(u)+c(v)}$ csökken $\Leftrightarrow$ $u \in F_1 \cup F_2$; $v \in L_1 \cup L_3$

De pont csak δ-val csökken és igaz lesz: $c'(u)+c'(v) \geq w(e)$

P = piros részgráf

P-ből kikerül $e = uv$, ha: $u \in F_3$; $v \in L_2$

$\Rightarrow$ M' élei pirosak maradnak

$\Rightarrow$ az L_2-beli lányok továbbra is elérhetőek F_1-ből

P-be bekerül $e = uv$ $\Leftrightarrow$ e olyan $F_1 \cup F_2$ és $L_1 \cup L_3$ között menő él, ahol a δ minimum felvétetik (legalább 1 ilyen él van)

$\Rightarrow$ Legfeljebb $|F| = |L| = n$ ciklus után M nő, mert kiürül L_3 és az új él csak L_1-be mehet.

$\Rightarrow$ Legfeljebb n^2 ciklus után megvan a teljes párosítás.

$\Rightarrow$ $O(n^2 e)$ lépésszámban végetér az algoritmus.

2006. 09. 14.

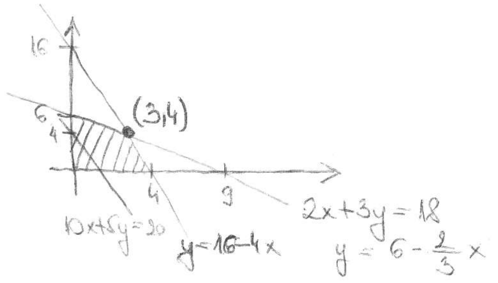
Drakula-művek:

1. szörny	2. szörny	raktár
4 fej	1 fej	16 fej
2 láb	3 láb	18 láb
↓	↓	
x db	y db	

profit: 1. szörny: 10 dollár
2. szörny: 5 dollár

$$\left.\begin{array}{r} 4x+y \leq 16 \\ 2x+3y \leq 18 \\ x, y \geq 0 \end{array}\right\}$$

$\max 10x+5y = S$



pl. 20 dollárnyi profit megvalósítható

$y = \frac{S}{5} - 2x$ Mi lesz a legnagyobb lehetséges S?

$S = 10 \cdot 3 + 5 \cdot 4 = \underline{\underline{50}}$ a max. profit

Ált. lin. programozási feladat:

$$\left.\begin{array}{l} a_{11}x_1 + a_{12}x_2 + \dots + a_{1n} \leq b_1 \\ \vdots \\ a_{m1}x_1 + a_{m2}x_2 + \dots + a_{mn} \leq b_m \\ \max c_1x_1 + \dots + c_nx_n \end{array}\right\} \text{LP alapfeladata}$$

$A = \begin{pmatrix} a_{11} & \dots & a_{1n} \\ \vdots & & \vdots \\ a_{m1} & & a_{mn} \end{pmatrix}$ $x = \begin{pmatrix} x_1 \\ \vdots \\ x_n \end{pmatrix}$ $b = \begin{pmatrix} b_1 \\ \vdots \\ b_m \end{pmatrix}$ $\Rightarrow$ $Ax \leq b$

$c = (c_1, c_2, \dots, c_n)$

LP alapfeladata:
$\max\{cx : Ax \leq b\}$