Kombinatorikus optimalizálás (VISZMA09)

2. ZH (2024.05.09.)

Az útmutató mintamegoldásokat tartalmaz. A pontszámok tájékoztató jelleggel lettek megállapítva az értékelés egységesítése céljából. Egy pontszám előtt szereplő állítás kimondása, tétel felidézése nem jelenti automatikusan az adott pontszám megszerzését. Az adott részpontszám megítélésének az a feltétele, hogy a megoldáshoz vezető gondolatmenet megfelelő részének végiggondolása világosan kiderüljön a dolgozatból. Ha ez utóbbi kiderül, ám a kérdéses állítás, tétel, definíció nincs rendesen kimondva, akkor a részpontszám legalább részben jár.
Természetesen az ismertetettektől eltérő, ám helyes megoldásokért teljes pontszámok, részmegoldásokért pedig az útmutatóbeli pontozás intelligens közelítésével meghatározott arányos részpontszámok járnak. Számolási hibáért általában 1 pontot vonunk le.

1. A $H = \{a,b,c,d,e,f,g\}$ alaphalmazt szeretnénk minél olcsóbban lefedni. A fedéshez az alábbi részhalmazokat használhatjuk, minden részhalmaz után a költsége áll:
$\{a,b\}$, 5; $\{a,b,e,f,g\}$, 12; $\{a,b,f\}$, 7; $\{a,c,e\}$, 6; $\{b,c,d,e\}$, 9; $\{c,e,f,g\}$, 10.
a) Állapítsuk meg, hogy az órán tanult mohó algoritmus mely részhalmazokkal és milyen költséggel tudja lefedni a H halmazt. (Dokumentáljuk az algoritmus lépéseit is.) **(8 pont)**
b) Optimális-e a kapott megoldás? **(2 pont)**

Megoldás: **a)** A mohó algoritmus futtatásához készítünk egy táblázatot, melyben az oszlopok a választható részhalmazoknak felelnek meg, és minden sorba felírjuk az adott részhalmaz által újonnan fedett elemeinek fajlagos költségét (azaz a halmaz költségét elosztjuk a részhalmazban levő, jelenleg még fedetlen elemek számával). A minimális fajlagos költségű részhalmazt bevesszük a fedésbe (ezt keretezés fogja jelezni); ennek az elemei tehát már le lesznek fedve, így a fajlagos költségeket újra kell számolni, ezt fogja tartalmazni a következő sor. (Ha egy részhalmaznak már minden eleme le van fedve, azt nincs értelme bevenni; ennek tényét kihúzással jelöljük.) Ezt addig ismételjük, míg a kiválasztott részhalmazok együttesen az alaphalmaz összes elemét le nem fedik. **(3 pont)**

részhalmaz és költsége	$\{a,b\}$, 5	$\{a,b,e,f,g\}$, 12	$\{a,b,f\}$, 7	$\{a,c,e\}$, 6	$\{b,c,d,e\}$, 9	$\{c,e,f,g\}$, 10
1. lépés, fajlagos ktsg-ek	5/2	12/5	7/3	$\boxed{2}$	9/4	10/4
2. lépés, fajlagos ktsg-ek	5	4	$\boxed{7/2}$	–	9/2	5
3. lépés, fajlagos ktsg-ek	–	12	–	–	$\boxed{9}$	10
4. lépés, fajlagos ktsg-ek	–	12	–	–	–	$\boxed{10}$

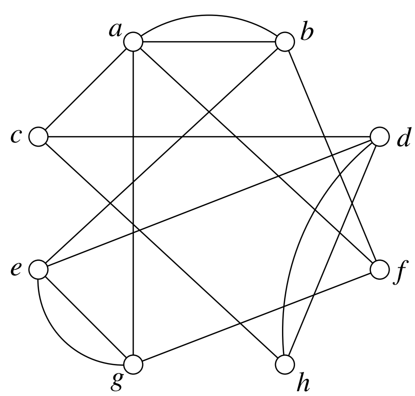
(3 pont)

A mohó algoritmus tehát az $\{a,c,e\}$ és a $\{a,b,f\}$, $\{b,c,d,e\}$ és $\{c,e,f,g\}$ részhalmazokat veszi be a fedésbe, összesen $6+7+9+10 = 32$ költséggel. Ezek persze valóban lefedik az alaphalmazt. **(2 pont)**

b) A talált fedés közel sem optimális, hiszen az $\{a,b,e,f,g\}$ és a $\{b,c,d,e\}$ részhalmazok 21 költséggel is lefedik az alaphalmazt. **(2 pont)**
(De az is szembetűnő lehet, hogy a mohó eljárás által kiválasztott részhalmazok közül pl. az $\{a,c,e\}$ elhagyásával továbbra is fedést kapunk, és persze olcsóbbat, így az eredeti nyilván nem volt optimális.)

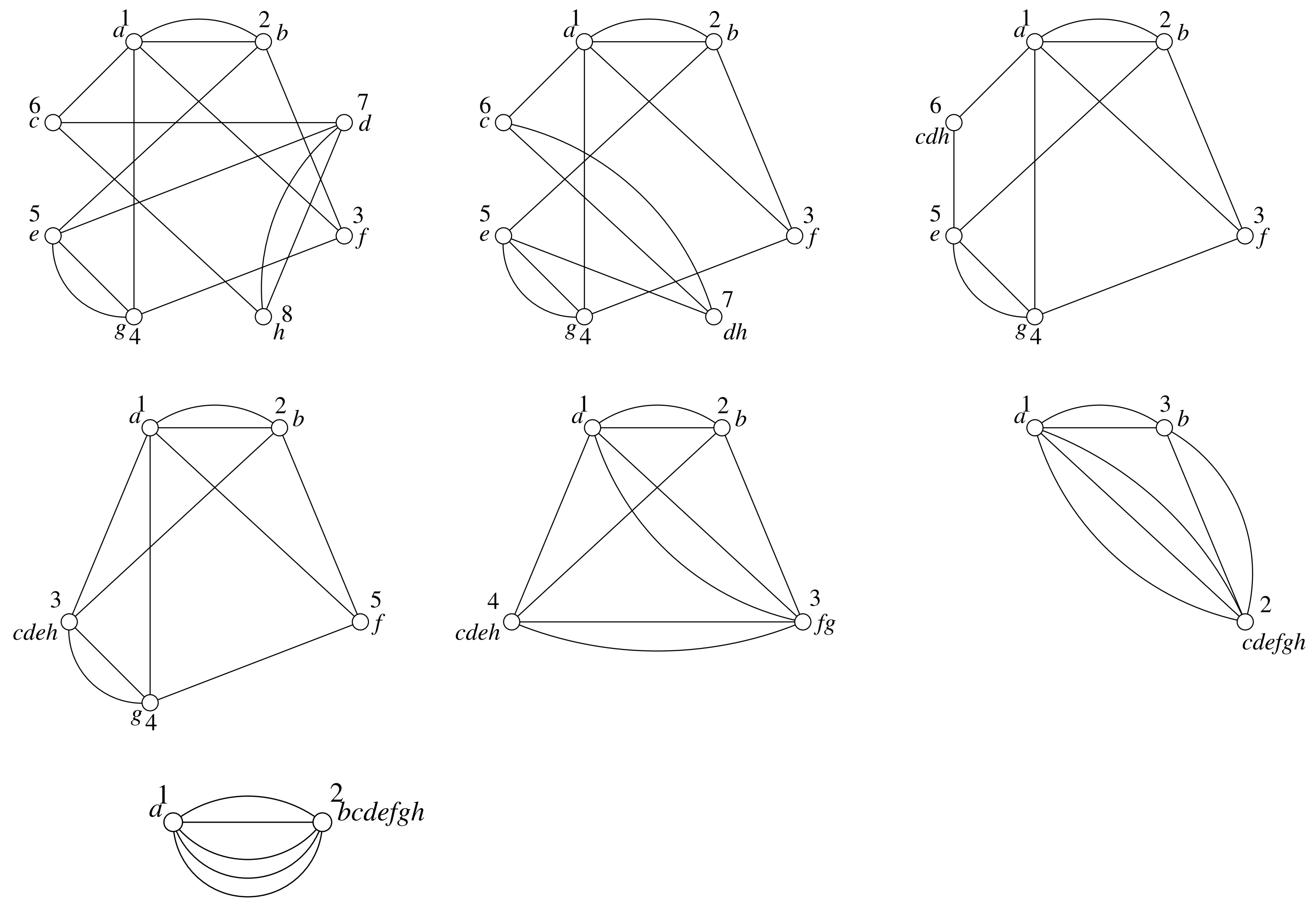
---

2. a) Határozzuk meg az ábrán látható gráf egy minimális vágását a Nagamochi–Ibaraki-algoritmus segítségével úgy, hogy amikor egy lépés során több csúcs közül is lehet választani, akkor mindig azt választjuk, amelyik betűrendbe sorolva a legelső közülük. **(9 pont)**
b) Van-e olyan maxvissza sorrendje a csúcsoknak, melyben a két utolsó csúcs c és e (valamilyen sorrendben)? **(5 pont)**



Nagyon figyeljünk oda a gráf átrajzolásának és az algoritmus lépéseinek pontos kivitelezésére, mert könnyű elrontani!

Megoldás: **a)** Minden lépésben maxvissza sorrendet keresünk, ügyelve a betűrendiségre is olyankor, amikor több lehetőségünk is van. A maxvissza sorrendben utolsó két csúcsot összeolvasztjuk, az esetlegesen keletkező hurokéleket töröljük, majd ismételjük az eljárást, míg már csak két csúcsunk marad. **(3 pont)**
Az algoritmus lépései az alábbi ábrákon láthatók.



(3 pont)

Az órán tanultak alapján az eredeti gráf egy minvágása előáll az algoritmus valamelyik lépésében úgy, mint a maxvissza sorrendben utolsó csúcsból induló élek halmaza. Az utolsó csúcsok fokszámai sorrendben 3, 3, 2, 3, 4, 4, 5, tehát a legkisebb szóba jövő vágás a harmadik lépésben előálló, a cdh csúcsból induló élekből álló kételemű vágás, ami az eredeti gráfban a $\{c,d,h\}$ halmazból kilépő ca és de éleknek felel meg. Tehát a ca, de élek minvágást alkotnak a gráfban. **(3 pont)**

b) Ha volna ilyen maxvissza sorrend, akkor a tanultak szerint a c és e csúcsokat elvágó élek minimális száma, $\lambda(c,e)$ megegyezne az utolsó csúcs, azaz c vagy e fokával, így $\lambda(c,e) \geq 3$ volna. **(2 pont)**
Node az előző részfeladatban talált kételemű (minimális) vágás elvágja c-t és e-t (elhagyva a ca és de éleket c és e különböző komponensekbe kerülnek), azaz $\lambda(c,e) \leq 2$. **(2 pont)**
Mivel ez lehetetlen, nincs ilyen maxvissza sorrend. **(1 pont)**

---

3. **a)** Írjuk fel az alábbi lineáris programozási feladat duálisát (hasonló formában)! **(8 pont)**

$$
\begin{aligned}
&\max\{x_1+x_2+2x_3+x_4\}\\
&\text{ha}\\
&x_1-3x_2+2x_4 \leq 3\\
&x_1+x_2-3x_3+x_4 \geq -2\\
&-2x_2+x_3+2x_4 \geq 3\\
&x_1-x_2+2x_3+6x_4 = 2\\
&x_3, x_4 \geq 0
\end{aligned}
$$

b) Mutassuk meg, hogy az $x_1 = x_2 = -7/6$, $x_3 = 0$, $x_4 = 1/3$ a primál feladat optimális megoldása, és az $y_1 = y_2 = y_4 = 1$, $y_3 = 3$ a duál feladat optimális megoldása. **(4 pont)**

Megoldás: **a)** Először hozzuk sztenderd alakra a rendszert. Mivel a célfüggvényben maximalizálunk, a lineáris feltételek $=$ és $\leq$ alakúak lehetnek (a nemnegativitási feltételek megmaradnak). Így a sztenderd alak az alábbi:

$$\begin{aligned}
&\max\{x_1 + x_2 + 2x_3 + x_4\}\\
&\text{ha}\\
&x_1 - 3x_2 + 2x_4 \leq 3\\
&-x_1 - x_2 + 3x_3 - x_4 \leq 2\\
&2x_2 - x_3 - 2x_4 \leq -3\\
&x_1 - x_2 + 2x_3 + 6x_4 = 2\\
&x_3, x_4 \geq 0
\end{aligned}$$

(1 pont)

A dualizálási ökölszabályok szerint a duális (DLP) feladat változói a primál (LP) feltételeinek (az együtthatómátrix sorainak) felelnek meg, **(1 pont)**

pontosan azon duál változók nemnegatívak, ahol a megfelelő primál feltételben egyenlőtlenség van, **(1 pont)**

mivel az LP-ben maximalizálunk, a DLP célfüggvényében minimalizálunk, az optimalizálandó mennyiség pedig a duál változóknak a primál feltételek jobb oldalaival vett kombinációja, **(1 pont)**

a DLP lineáris feltételeinek jobb oldalai a primál célfüggvény együtthatói, és mivel a DLP is sztenderd alakú, $=$ és $\geq$ relációk szerepelhetnek, **(1 pont)**

mégpedig pontosan a nemnegatív primál változóknak megfelelő duális feltételeknél lesz egyenlőtlenség. **(1 pont)**

Felrajzolhatjuk a szamárvezetőt is: **(1 pont)**

(itt az együtthatómátrix helyes megállapítására jár a pont)

$$\begin{array}{cc|rrrr|l}
 & & & & 0\leq & 0\leq & \\
 & & x_1 & x_2 & x_3 & x_4 & \\
\hline
0\leq & y_1 & 1 & -3 & 0 & 2 & \leq 3\\
0\leq & y_2 & -1 & -1 & 3 & -1 & \leq 2\\
0\leq & y_3 & 0 & 2 & -1 & -2 & \leq -3\\
 & y_4 & 1 & -1 & 2 & 6 & = 2\\
\hline
 & & = & = & \geq & \geq & \\
 & & 1 & 1 & 2 & 1 &
\end{array}$$

Ezek alapján a DLP:

$$\begin{aligned}
&\min\{3y_1 + 2y_2 - 3y_3 + 2y_4\}\\
&\text{ha}\\
&y_1 - y_2 + y_4 = 1\\
&-3y_1 - y_2 + 2y_3 - y_4 = 1\\
&3y_2 - y_3 + 2y_4 \geq 2\\
&2y_1 - y_2 - 2y_3 + 6y_4 \geq 1\\
&y_1, y_2, y_3, \geq 0
\end{aligned}$$

(1 pont)

(A DLP felírására járó részpontok annak járnak, aki a megfelelő dualizálás szabályt sikerrel alkalmazta (tehát a megfogalmazás önmagában nem ér pontot, de a megfogalmazás hiánya nem jelent okvetlen pontveszteséget.)

b) Az LP feltételeibe behelyettesítve az adott $x_1, \ldots, x_4$ értékeket könnyen kiszámolhatjuk, hogy a megoldásjelöltünk mindegyiknek megfelel: a nemnegativitási feltételek triviálisak, a lineáris feltételek pedig épp egyenlőséggel teljesülnek. A DLP-nél hasonló a helyzet. **(1 pont)**

Kiszámolva a primál és a duál célfüggvények értékét az adott megoldásokra, mindkettőnél -2 adódik. **(1 pont)**

Mivel tetszőleges primál megoldásra a maximalizálandó célfüggvény értéke legföljebb annyi, mint tetszőleges duál megoldásra a minimalizálandó duál célfügvény értéke, **(1 pont)**

nincs olyan LP megoldás, mely -2-nél nagyobb, sem olyan DLP megoldás, mely -2-nél kisebb célfüggvényértéket eredményezne. **(1 pont)**

---

4. Egy logisztikai raktárban egy automata targonca tölti meg az áruszállító teherautókat különféle méretű ládákkal, melyek egy polc két szintjén helyezkednek el. Ha a soron következő láda nem férne be a várakozó teherautók egyikébe sem, akkor a láda mozgatása előtt a targonca hív egy új járművet, majd annak megérkezése után folytatja a rakodást

(így tesz például rögtön az elején, amikor még egy teherautó sincs ott). Minden teherautó kapacitása 100 egység. Az alsó polcon levő $a_1,\ldots,a_4$ ládák méretei 30, 60, 50, 40, míg a fölső polcon lévő $f_1,\ldots,f_4$ ládáké 40, 20, 20, 40 egység.
a) Hány teherautóra lesz szüksége a targoncának, ha az FFD algoritmus szerint pakolja a ládákat? **(6 pont)**

Előfordulhat, hogy az FFD algoritmus több láda közül is választhat. Ilyenkor a raktár rakodási protokollja szerint prioritást élveznek a fölső polcon levő ládák, illetve az azonos polcon levők közül a kisebb sorszámúak.
b) Az alsó polcon levő ládák átrakodási ideje 1 egység, a fölső polcon levőké 2 egység, új teherautó hívása esetén a várakozási idő 3 egység. Mennyi az előzőleg vizsgált, a raktári protokollnak megfelelő rakodás teljes átfutási ideje, illetve a ládák átpakolásának átlagos átfutási ideje? Lehet-e csökkenteni a teljes átfutási időt? **(8 pont)**

Megoldás: **a)** Az FFD eljárást követve a targonca mindig a legnagyobb elpakolandó ládát teszi az első teherautóba, melybe belefér (szükség esetén újat hív), azaz 60, 50, 40, 40, 40, 30, 20, 20 sorrendben helyezi el a ládákat. **(2 pont)**
Az egyes teherautókba rakott ládák méretei ezek alapján a következőképpen alakulnak: T_1: $60+40$; T_2: $50+40$; T_3: $40+30+20$; T_4: 20. **(3 pont)**
Tehát 4 teherautóra lesz szükség. **(1 pont)**
b) Részletezve az előző folyamatot, a műveletek sorrendje és a befejezési időpontjuk az alábbi:

művelet	T_1 hív	$a_2 \to T_1$	T_2 hív	$a_3 \to T_2$	$f_1 \to T_1$	$f_4 \to T_2$	T_3 hív	$a_4 \to T_3$	$a_1 \to T_3$	$f_2 \to T_3$
bef. idő	3	4	7	8	10	12	15	16	17	19

művelet	T_4 hív	$f_3 \to T_4$
bef. idő	22	24

(3 pont)
A rakodás teljes átfutási ideje tehát 24 egység. **(1 pont)**
(Ez persze rögtön látszik abból is, hogy a 8 ládát $4\cdot 1+4\cdot 2=12$ egységnyi idő alatt rakodjuk át, és a 4 teherautóra $4\cdot 3=12$ időegységet kell várni. Akinél hiányzik a pontos rakodási folyamat áttekintése, de helyesen meghatározza a teljes átfutási időt, az erre **2 pont**-ot kaphat.)
A ládák átlagos átrakodási ideje a 8 ládapakolási művelet befejezési időpontjainak átlaga (a teherautóhívások miatti várakozások miatt egyes ládák átrakodásának befejezési ideje kitolódik, de a teherautóhívás maga nem ládapakolási művelet), azaz $(4+8+10+12+16+17+19+24)/8$. **(2 pont)**
(Ez egyébként $110/8=13\frac{6}{8}$.) **(0 pont)**
A teljes átfutási idő az átrakodási műveletek, valamint a teherautóhívások összidőigényéből tevődnek össze. Az előbbi állandó ($4\cdot 1+4\cdot 2=12$), az utóbbi azonban csökkenthető lehet, ha a ládák beférnek 3 teherautóba is. Márpedig beférnek, így: T_1: $60+40$; T_2: $50+30+20$; T_3: $40+40+20$. Tehát a válasz igen, a teljes átfutási idő csökkenthető (mégpedig 35-re). **(2 pont)**