Bináris és relaxált hátizsák

Hujter Mihály
(http://www.math.bme.hu/~hujter/knapsack.pdf)

Adottak pozitív egész számok. Elsőként n, melynek értéke a zárthelyin vagy vizsgán 5 és 10 közötti. Aztán adott n darab számpár pozitív számokból. Például $n = 6$ esetén: $\binom{8}{6}, \binom{4}{5}, \binom{6}{8}, \binom{5}{4}, \binom{9}{7}, \binom{2}{3}$. A számpárok első elemét euróban, a másodikat kilóban értjük. Tehát például az $\binom{5}{4}$ pár jelentése: egy olyan tárgy, melynek a haszna 5 euró, a súlya pedig 4 kiló. Adott továbbá egy b pozitív egész szám is. Ez általában kisebb, mint a felsorolt n darab súly összege, de nagyobb mint a legnagyobb egyedüli súly. A vizsgán és a zárthelyin általában a legnagyobb súly kétszere-háromszorosa körül lesz. Az itteni példánkban legyen $b = 25$. Úgy fogjuk fel, hogy van egy hátizsákunk, melybe legfeljebb b kiló pakolható. Ki kell válogatnunk a tárgyak közül néhányat, melyek összsúlya nem több, mint b, de az összhaszna a lehető legtöbb. Ez az úgynevezett *bináris hátizsák* feladat (angolul: *binay knapsack*). Azért bináris, mert mind az n darab tárgy esetében két-két lehetőségünk van: vagy kiválasztjuk, vagy nem.

Egészértékű lineáris programozási feladatként megfogalmazva: Ha a tárgyak sorszáma: $j = 1, 2, ..., n$, akkor a j-edik tárgy haszna c_j, a súlya a_j, továbbá legyen $x_j = 1$, ha a tárgyat a hátizsákba tesszük, és legyen $x_j = 0$, ha a tárgyat nem rakjuk a hátizsákba. Annak eldöntése, hogy mely tárgyakat rakjuk a hátizsákba lényegében az x vektor megkeresését jelenti.

Tehát ismeretlenek: $x_j \in \{0, 1\}$, $j = 1, 2, \ldots, n$. Feltétel: $\sum a_j x_j \leq b$. Célfüggvény, amit maximalizálunk: $\sum c_j x_j$.

Az a gond a bináris hátizsák feladattal, hogy már például $n = 100$-ra is reménytelen az összes lehetőség megvizsgálása számítógéppel, hiszen 2^n darab megoldásjelöltet kellene kipróbálni. Hogy mégis kezelni tudjunk ilyen feladatokat, először egy kicsit módosított változatát vizsgálgatjuk.

Az úgynevezetett *relaxált hátizsák* (angolul: *relaxed knapsack*) feladat esetében az egyik (de csak az egyik) x_j-nek megengedünk a 0 és 1 értéken kívül is valamely 0 és 1 közötti értéket is. Ez a 0 és 1 között érték egy hányados lehet, melyben a nevező valamely a_m, a számláló pedig egy olyan pozitív szám, mely a_m-nél kisebb.

Minden lényegesen leegyszerűsödik, ha feltesszük, hogy a tárgyaink olyan sorrendben adottak, hogy a c_j/a_j sorozat monoton fogyó. A monoton fogyó sorozat nem feltétlenül lesz szigorúan monoton fogyó! Tehát ha kapunk néhány haszon-súly számpárt, akkor az egy kilóra eső haszon hányadosok mentén monoton fogyó sorozatba rendezzük a számpárokat. Biztonsági okokból érdemes ellenőrizni, hogy jó-e a végső sorrend. Ezt legkönnyebben úgy tehetjük meg, hogy $j = 1, 2, ..., n-1$ esetén megnézzük, hogy fennáll-e $c_j a_{j+1} \geq c_{j+1} a_j$; ha valahol gubanc lenne, ott sorrendet kellene cserélni, és újra ellenőrizni.

Feltesszük tehát, hogy már jó sorrendben vannak a párok. Ekkor keresünk egy olyan k számot, melyre $\sum_{j=1}^{k} a_j < b$, de $\sum_{j=1}^{k+1} a_j > b$. Azaz megkeressük azt a k sorszámot, hogy vagy $k = n$, azaz minden elfár a hátizsákban, vagy $k < n$, de ilyenkor a $(k+1)$-edik tárgy már nem fér a hátizsákba, de az összes korábbi még befért. Itt elméletileg $k = 0$ is lehetséges, és ilyenkor a $\sum_{j=1}^{0}$ összeg értéke nulla lesz.

Két triviális eset van: Ha $k = 0$, akkor az első tárgy nem fér a hátizsákba. Mint hasznavehetetlent, dobjuk el mindenestül, és n értékét csökkentsük 1-gyel, és kezdjük elölről az egész eljárást. Természetesen a kialakított sorrendet megtarthatjuk az eggyel kevesebb számpárra.

Ha $k = n$, akkor minden tárgy befért a hátizsákba; ez nyilván a remélhető legjobb megoldás. Nincs vele több gondunk.

Az általános esetben azonban a $k \in \{1, 2, ..., n-1\}$. A relaxált hátizsák feladat optimális megoldását így készíthetjük el: Ha $k \in \{1, 2, ..., n-1\}$, akkor legyen

$$\begin{aligned} x_j &= 1, \qquad \text{ha } j = 1, 2, \ldots, k \\ x_{k+1} &= \frac{b - \sum_{j=1}^{k} a_j}{a_{k+1}} \\ x_j &= 0, \qquad \text{ha } j = k+2, k+3, \ldots, n \end{aligned}$$

Ha pedig $k \in \{0, n\}$ szám a fenti értelemben, akkor legyen

$$\begin{aligned} x_j &= 1, \qquad \text{ha } \sum_{p=1}^{j} a_p \leq b \\ x_j &= 0, \qquad \text{máskülönben} \end{aligned}$$

Viszonylag könnyen belátható: A relaxált hátizsák feladat optimális megoldását kaptuk. Sőt, ha az így definiált x_j számokra kiszámítjuk a $\sum_{j=1}^{n} c_j x_j$ összeget, és azt egész számra lefelé kerekítjük, akkor legfeljebb ennyi lehet a bináris hátizsák feladat optimális célfüggvényértéke is. Példánk esetében a tárgyakat monoton fogyó c_j/a_j sorozatba rendezve és a

	c_1	c_2	$\cdots$	c_n
b	a_1	a_2	$\cdots$	a_n

séma szerint elrendezve ezt kapjuk:

	8	9	5	4	6	2
25	6	7	4	5	8	3

Ha a relaxált feladat megoldását is beírjuk a

	c_1	c_2	$\cdots$	c_n
b	a_1	a_2	$\cdots$	a_n
	x_1	x_2	$\cdots$	x_n

sémába, akkor a példánk esetében ezt kapjuk:

	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	1	1	1	3/8	0

A relaxált optimáis megoldás célfüggfényértékét lefele kerekítve ezt kapjuk:

$$8 + 9 + 5 + 4 + \left\lfloor \frac{3}{8} \cdot 6 \right\rfloor = 28$$

Most rátérve a bináris feladatra, ebből az utolsó táblázatból azt látjuk, hogy a bináris feladat optimális megoldása legalább $8 + 9 + 5 + 4 = 26$ és legfeljebb 28. A bináris feladatra vonatkozó információt a következő táblázatos formában rögzítjük:

26<28	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	1	1	1	0	0

Az eddig megtalált lehető legjobb bináris célfüggvényérték 26 volt, tehát a továbbiakban csak olyan megoldások keresésére szoritkozunk, ahol a bináris célfüggvényérték a 27,28 számok valamelyike lesz. Ehhez okvetlenül szükséges, hogy legalább egy olyan x_j értéke 0 legyen, mely x_j értéke az eddig megtalált legjobb bináris megoldásban pozitív volt. Jelen esetben tehát vagy $x_1 = 0$, vagy $x_1 = 1$, $x_2 = 0$, vagy $x_1 = 1$, $x_2 = 1$, $x_3 = 0$, vagy $x_1 = 1$, $x_2 = 1$, $x_3 = 1$, $x_4 = 0$, vagy $x_1 = 1$, $x_2 = 1$, $x_3 = 1$, $x_4 = 1$, $x_4 = 0$ lesz. Ezt az öt esetet a következő öt táblázatban írhatjuk fel:

	8	9	5	4	6	2
25	6	7	4	5	8	3
	0					

	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	0				

	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	1	0			

	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	1	1	0		

	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	1	1	1	0	

Mind az öt esetben megoldhatjuk a megfelelő relaxált feladatot, és a bal felső sarokba beírhatjuk a célfüggvényértéket illetve a kétféle célfüggvényértéket azokban az esetekben, amikor $\left(b - \sum_{j=1}^{k} a_j\right) c_{k+1} \geq a_{k+1}$. Ezeket a táblazatokat kapjuk, melyekben az áttekinthetőség érdekében az újonnan kiszámított x_j-értékeket kisebb számjegyekkel írtuk be:

24	8	9	5	4	6	2
25	6	7	4	5	8	3
	0	1	1	1	1	1/3

23<24	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	0	1	1	1	2/3

21<26	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	1	0	1	7/8	0

28	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	1	1	0	1	0

28	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	1	1	1	0	1

Látjuk tehát, hogy a negyedik esetnél már sikerült megtalálni a 28-at, aminél jobb nem is lehetséges. Az alsó sorból olvashatjuk ki az optimális megoldást. (Az ötödik változatot már nem is kellett volna kiszámítanunk.)

Tekintsünk egy másik példát, ahol rögtön felírjuk a relaxált megoldásból származó adatokat is:

70<85	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	1	1	1	1	7/10	0	0

Mivel öt darab x_j pozitív, ezért öt darab esetet kapunk:

	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	0						

	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	1	0					

	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	1	1	0				

	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	1	1	1	0			

	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	1	1	1	1	0		

Sorban mindegyik esetben kiszámítjuk a relaxált megoldásokat és az azokhoz tartozó becsléseket:

73<80	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	0	1	1	1	1	1	4/9

85	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	1	0	1	1	1	0	0

77<84	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	1	1	0	1	1	1	4/9

83	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	1	1	1	0	1	0	0

80<83	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	1	1	1	1	0	1	2/9

Látható, hogy a kiindulási 70-es optimumhoz képest az első esetben javulást találtunk 73-ra, a második esetnél ez tovább javult a 85-re, a harmadik esetnél nem volt további javulás, de a negyedik esetnél 83-ra jött ki a bináris optimális megoldás, az ötödik esetnél sem jött ki jobb. A második esetnél lett meg az optimális megoldás!

A fentiekben vázolt algoritmus sok táblázat leírását és újraírását igényli. Most felülvizsgáljuk az algoritmusunkat, és próbálunk rövidítéseket beiktatni. Tekintsük az első példa kiindulási relaxált megoldását:

26<28	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	1	1	1	3/8	0

Itt azon c_j számok összegére, melyre $x_j < 1$, ez kapjuk: $6 + 2 = 8$. Tehát ha a 26-ot, mint célfüggvényértéket, növelni szeretnénk, akkor a 26-hoz legfeljebb 8-at adhatunk hozzá, de el kell venni azon c_j számok valamelyikét, melyekre $x_j = 1$, azaz a $8, 9, 5, 4$ számok valamelyikét. Jelen esetben a 26-os szám csak akkor növelhető, ha 8-nál kisebb számot veszünk el. Biztosra vehetjük tehát, hogy az optimális bináris megoldásban $x_1 = 1$ és $x_2 = 1$. A relaxált megoldásunkban tehát x_1 és x_2 értékét kisméretű egyesről nagyméretűre változtathatjuk:

26<28	8	9	5	4	6	2
25	6	7	4	5	8	3
	1	1	1	1	3/8	0

De most már össze is vonhatjuk a c_1 és c_2 számokat, illetve az a_1 és a_2 számokat:

26<28	17	5	4	6	2
25	13	4	5	8	3
	1	1	1	3/8	0

Ez a feladatág 4 irányban ágazik szét:

	17	5	4	6	2
25	13	4	5	8	3
	0				

	17	5	4	6	2
25	13	4	5	8	3
	1	0			

	17	5	4	6	2
25	13	4	5	8	3
	1	1	0		

	17	5	4	6	2
25	13	4	5	8	3
	1	1	1	0	

Ezek az ágak egyszerűsíthetők úgy, hogy olyan oszlopokat, ahol $x_j = 0$, egyszerűen elhagyjuk, az egymás melleti olyan oszlopokat pedig, ahol $x_j = 1$, összevonjuk:

	5	4	6	2
25	4	5	8	3

	17	4	6	2
25	13	5	8	3
	1			

	22	6	2
25	17	8	3
	1		

	26	2
25	22	3
	1	

Mindegyik ágban kiszámítjuk a relaxált megoldásokat:

17	5	4	6	2
25	4	5	8	3
	1	1	1	1

21<26	17	4	6	2
25	13	5	8	3
	1	1	7/8	0

28	22	6	2
25	17	8	3
	1	1	0

28	26	2
25	22	3
	1	1

Az utolsó előtti vagy az utolsó ágon megkapjuk az optimális megoldást.
A másik példa esetében a kiindulási relaxált megoldás ez volt:

70<85	29	7	25	9	22	10	17
37	12	3	11	4	10	5	9
	1	1	1	1	7/10	0	0

Ebből 5 ág származik, melyekbe rögtön bele is írjuk a relaxált megoldásokat, és örvendetes módon már a második táblázatnál abbahagyhatjuk a keresgélést:

73<80	7	25	9	22	10	17
37	3	11	4	10	5	9
	1	1	1	1	1	4/9

85	29	25	9	22	10	17
37	12	11	4	10	5	9
	1	1	1	1	0	0