Kisimre István

A „MAGYAR MÓDSZER" ALKALMAZÁSA

1. BEVEZETŐ

Ma már nem engedhetjük meg magunknak azt a fényüzést, hogy a gazdasági problémákat megoldó optimalizáló módszereket csak néhány szakember ismerje és csak itt-ott alkalmazzák. Nagyon fontos, hogy az operációkutatás matematikai módszereit ne csak matematikusok tudják használni, hanem egyszerűbb, típus esetekben mások is (közgazdászok, mérnökök, szervezők, stb.). Ezért kell közérthető módon ismertetni a módszerek alkalmazási módját és lehetőségeit. Ha ugyanis a felhasználó egyszerű, jól összeállított példákon keresztül megérti a probléma lényegét, és megtanulja a megoldás módját, akkor maga is alkalmazni próbálja, és előbb-utóbb eredményt is ér el vele.

Ebben a munkában a magyar módszer alkalmazását mutatjuk be szállítási, hozzárendelési és azokhoz hasonló problémák megoldására. Honnan ered a magyar módszer? H. W. Kuhn, amerikai matematikus 1955-ben az úgynevezett hozzárendelési probléma megoldásához a magyar König [6] és Egerváry [1] tételeit használta fel, és az így kapott módszert elnevezte magyar módszernek. Később Egerváry ezt a módszert tökéletesítette, és azóta igen sokféle probléma megoldására alkalmas.

A magyar módszer egyes kérdéseivel a [2], [3], [4] és [7] -es művek is foglalkoznak, de az eddigi egyetlen, a magyar módszer teljes problematikáját egy helyen összefoglaló mű az [5]. Abban megtalálható a módszer elméleti feldolgozása, a felhasznált tételek bizonyítása és a gyakorlati alkalmazás részletes ismertetése. Ismerteti a magyar módszer azon jó tulajdonságait, amelyek kiemelik azt a hasonló módszerek közül, mellékletként pedig egy FORTRAN programozási nyelven írt programot ad, amely segítségével elektronikus számítógépen is megoldhatók a szállítási és hozzárendelési problémák.

Ez a munka az [5]-nek egy részét tartalmazza kissé átdolgozva.

2. HOZZÁRENDELÉSI PROBLÉMA

Bizonyára sokak számára ismerős az a feladvány, amelyben 8 bástyát kell úgy elhelyezni a sakktáblán, hogy azok ne üssék egymást. Megoldást könnyű találni, hiszen 40320 különböző, a feltételt kielégítő elhelyezés között válogathatunk.

Bővítsük ki most ezt a feladványt új feltételekkel. Írjunk a sakktábla minden mezejére egy-egy számot, mondjuk, pozitív egész számokat Például így:

8	5	9	11	5	5	6	4	8
7	3	12	9	8	4	5	3	6
6	1	1	6	5	1	3	1	1
5	2	2	3	7	2	3	2	1
4	5	7	4	10	6	4	6	4
3	3	2	4	9	2	2	5	2
2	11	3	7	4	3	2	8	3
1	4	4	4	6	5	7	5	8
	a	b	c	d	e	f	g	h

A feladat ismét a 8 bástya elhelyezése a táblán úgy, hogy azok ne üssék egymást, ám követelmény még az is, hogy a bástyák által letakart számok összege a lehető legkisebb legyen.

Így már lényegesen nehezebb megoldást találni, hisz a 40320 különböző elhelyezés ugyanannyi összeget jelent, és azok kiszámítása, összehasonlítása igen sok munkát igényel. Ha az egy-egy elhelyezés által letakart számokat fél perc alatt összegezzük és hasonlítjuk össze a már kiszámolt összegek legkisebbikével, akkor is 336 óra kell a feladat ily módon való megoldásához.

Ilyen típusú problémákkal nemcsak szórakoztató feladványokban találkozunk, hanem a gyakorlati életben is. A konkrét feladatok pedig nem 8 „bástya" elhelyezését igénylik, hanem jóval többét. Viszont egy 20×20-as tábla esetében, ami még aránylag kisméretű, az összes változat kiszámítása pár milliárd millió évszázadig tartana!

Hogyan lehet akkor az ilyen feladatokat megoldani viszonylag egyszerűen és gyorsan? Erre mutatunk be ebben a fejezetben egy eljárást, a magyar módszert.

2.1 *A hozzárendelési probléma általános alakja*

Ahhoz, hogy egy általános eljárást tudjunk adni, a feladatot is általános alakban kell felírni. Mivel ehhez matematikai fogalmakat használunk, az általános alakot gyakran nevezzük matematikai modellnek is.

Tekintsük az előbb említett sakktáblát egy 8×8-as mátrixnak, melynek elemei a tábla mezőiben levő számok. Jelöljük C-vel a mátrixot, elemeit pedig c_{ij}-vel (ahol i azt mutatja, hogy a mátrix hányadik sorában, j pedig, hogy hányadik oszlopában van az az elem, az i és j az 1, 2, 3,... 8 számok közül vesz fel értékeket).

Még egy mátrixot használunk, a megoldásmátrixot, amely ugyanolyan méretű, mint a C. Ezt *X*-szel jelöljük, elemeit pedig x_{ij}-vel. Ha az i-edik sor és j-edik oszlop metszéspontjában bástya van, akkor x_{ij} eggyel egyenlő, különben nulla. Mivel minden sorban és minden oszlopban pontosan egy bástya lehet, ezért az x_{ij}-k soronkénti és oszloponkénti összege is egy, amit így írunk fel:

$$\sum_{i=1}^{8} x_{ij} = 1 \qquad j = 1,2,\ldots,8$$

$$\sum_{j=1}^{8} x_{ij} = 1 \qquad i = 1,2,\ldots,8.$$

A feladat azt is megköveteli, hogy a bástyák által lefedett mezők összege a lehető legkisebb, más szóval, minimális legyen. Ez a matematika eszközeivel felírva:

$$\sum_{i=1}^{8} \sum_{j=1}^{8} c_{ij} x_{ij} \longrightarrow \min.$$

A teljes általánosításnál a mátrix mérete sem lehet rögzített szám, hanem az is a feladattól függően változik. Jelöljük hát n-nel a sorainak számát, ami egyben az oszlopok száma is.

Az így általánosított feladatot hozzárendelési problémának nevezzük és matematikai modellje a következő:

$$\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \longrightarrow \min$$

$$\sum_{i=1}^{n} x_{ij} = 1 \qquad j = 1,2,\ldots,n$$

$$\sum_{j=1}^{n} x_{ij} = 1 \qquad i = 1,2,\ldots,n$$

$$x_{ij} \in \{0,1\} \qquad i,j = 1,2,\ldots,n,$$

($x_{ij} \in \{0,1\}$ azt jelenti, hogy x_{ij} értéke csak 0 vagy 1 lehet)

2.2 *A hozzárendelési probléma megoldása*

A hozzárendelési problémákat legeredményesebben a magyar módszerrel oldhatjuk meg. A továbbiakban ezt a módszert ismertetjük, méghozzá algoritmikus formában, azaz lépésről lépésre megadjuk, hogy mit kell tenni a megoldás érdekében.

A hozzárendelési problémánál adott a $C = ||c_{ij}||_n$ mátrix (a $||c_{ij}||_n$ azt jelenti, hogy egy $n \times n$-es mátrixszal van dolgunk, melynek elemeit c_{ij}-vel jelöljük), a megoldás pedig a mátrixnak n megjelölt eleme, melyeknek az összege a lehető legkisebb és amelyek közül semelyik kettő sincs azonos sorban vagy azonos oszlopban.

A hozzárendelési problémát megoldó eljárás (algoritmus) hét különböző műveletcsoportból áll, ezeket röviden lépéseknek fogjuk nevezni. Közülük az első kettő előkészítő jellegű, a többit pedig bizonyos szabályok szerint addig ismételjük, amíg a megoldáshoz el nem jutunk. Bebizonyítható, hogy a hozzárendelési problémának mindig van megoldása, és azt véges számú lépés után meg is kapjuk.

1. lépés: A C mátrix minden oszlopában megkeressük a legkisebb értékű, elemet, és az oszlop minden eleméből levonjuk azt az értéket. A kapott mátrix minden sorában szintén megkeressük a legkisebb értéket, és azt levonjuk a sor összes eleméből. Így jutunk el a C_0 mátrixhoz. Ennek a mátrixnak minden eleme nem negatív, és minden oszlopában és minden sorában van bár egy elem melynek értéke nulla.

2. lépés: A C_0 mátrix első oszlopában egy elemet, amelynek értéke nulla (röviden: nullát), megjelölünk csillaggal (az ilyen elemet ezután csillagos nullának hívjuk). Ezután a többi oszlopban is megkísérelünk egy-egy nullát csillaggal jelölni, de csak olyanokat, amelyeknek a sorában még nincs csillagos nulla. Vigyázni kell arra, hogy semelyik sorban és semelyik oszlopban se legyen egynél több csillagos nulla.

3. lépés: Megnézzük, hogy hány nulla van csillaggal jelölve. Ha n, akkor megkaptuk a megoldást, azt a megjelölt nullák képezik. Ha a C_0 mátrixban lévő csillagos nulláknak megfelelő elemeket a C mátrixban összeadjuk, akkor megkapjuk a legkisebb összeget is.

Ha a csillagos nullák száma kisebb mint n, akkor a C_0 mátrix minden oszlopát, amelyben csillagos nulla van keresztel jelöljük (fölé írunk egy keresztet). Ezentúl minden olyan elemét a mátrixnak megjelöltnek nevezzük, amely keresztel jelölt sorban vagy oszlopban van. Ha mind a sor, mind az oszlop megjelölt, akkor az elemet kétszerjelöltnek mondjuk. Az összes többi elem közös neve pedig megjelöletlen.

4. lépés: Megnézzük, hogy van-e a mátrixban megjelöletlen nulla (azaz, olyan nulla értékű elem, amelynek se sora, se oszlopa nincs keresztel jelölve). Ha nincs, akkor a 7. lépéssel folytatjuk a megoldást, ha van akkor

pedig kiválasztunk egyet, azt vesszővel annak sorát kereszttel jelöljük és áttérünk az 5. lépésre.

5. lépés: Tegyük fel, hogy a 4. lépésben egy olyan nullát jelöltünk meg vesszővel, amely az i_1-edik sorban és j_1-edik oszlopban van (ezt röviden igy írjuk: $0'_{i_1 j_1}$). Ha ennek a sorában van csillagos nulla, akkor annak oszlopa fölött töröljük a keresztet és visszatérünk a 4. lépésre. Ha pedig abban a sorban nincs csillagos nulla, a 6. lépés végrehajtása következik.

6. lépés: Az a nulla, amely idáig juttatott bennünket a $0'_{i_1 j_1}$ volt. Ha annak oszlopában nincs csillagos nulla, akkor töröljük róla a vesszőt és csillaggal jelöljük. Ezután a mátrixban töröljük a kereszteket és vesszőket és visszatérünk a 3. lépésre.

Ha viszont a $0'_{i_1 j_1}$ oszlopában van csillagos nulla, mondjuk az i_2-ik sorban (tehát $0^*_{i_2 j_1}$), akkor ugyanabban a sorban van vesszővel jelölt nulla is (mégpedig egy, legyen az $0'_{i_2 j_2}$). Ez utóbbi oszlopában újabb csillagos nulla lehet, és ha van, annak sorában megint van vesszős. Így haladva végig a csillagos és vesszős nullákon, valamelyik 0' oszlopában már nem lesz csillagos nulla és megszakad a lánc. Az addig elért nullák a következők:

$$0'_{i_1 j_1},\ 0^*_{i_2 j_1},\ 0'_{i_2 j_2}, \ldots,\ 0^*_{i_s j_{s-1}},\ 0'_{i_s j_s}.$$

Ezt a sorozatot láncnak nevezzük. Bebizonyítható, hogy ha megadjuk a lánc kezdő elemét (itt $0'_{i_1 j_1}$), akkor a többi eleme egyértelműen meghatározott; valamint az is, hogy a lánc mindig véges sok elemből áll, és vesszővel jelzett nullával végződik.

Ezen lánc mentén most a csillagos nullákról töröljük a csillagot, a vesszővel jelzetteket pedig csillaggal jelöljük. Mivel vesszővel jelölt nulla eggyel több van a láncban mint csillagos, a csillagos nullák száma most eggyel növekszik. A mátrixban töröljük a kereszteket és a vesszőket és a 3. lépésre térünk vissza.

7. lépés: Erre a lépésre akkor jutunk ha a C_0 mátrixban nincs többé megjelöletlen nulla. A megjelöletlen elemek közül megkeressük a legkisebbet, ami mindig pozitív szám. Ezt a számot levonjuk az összes megjelöletlen elemből és hozzáadjuk a kétszerjelöltekhez. A többi elemet, valamint a jelöléseket változatlanul átírjuk az új mátrixba, amit szintén C_0-val jelölünk. Az is csupa nem negatív elemből áll és tartalmaz megjelöletlen nullát. A megoldást a 4. lépéssel folytatjuk.

2.3 *Egy gyakorlati példa megoldása*

Az előbb ismertetett eljárással minden hozzárendelési problémát megoldhatunk, a kisebb méretűeket papíron, a nagyobbakat pedig elektronikus számítógéppel. Mi most egy egészen kisméretű feladaton mutatjuk be gyakorlatban is a megoldás menetét.

A probléma a következő:

Egy tervezőiroda főnökének öt elkészítendő tervet kell szétosztania öt mérnöke között. Tudja, hogy melyik mérnök, melyik tervet, hány nap alatt készítené el, és erről egy táblázatot készít:

	tervek				
	I	II	III	IV	V
1. mérnök	4	3	6	5	8
2. mérnök	3	4	4	3	9
3. mérnök	3	5	3	2	6
4. mérnök	3	2	4	2	7
5. mérnök	3	4	5	4	9

Mivel azt szeretné, hogy minél kevesebb munkanapot fordítsanak a tervek elkészítésére, keresi azt az elosztást, amely ezt biztosítja. (Minden mérnök egyedül készíti el a kapott tervet.)

Ez egy hozzárendelési probléma, amelynek C mátrixa a fenti táblázat. A feladat az, hogy abban 5 elemet jelöljünk ki, melyek mind más-más sorban és oszlopban helyezkednek el, és amelyeknek az összege a lehető legkisebb.

Mivel a feladat kisméretű, megoldást találgatással is lehet kapni, de itt a leírt algoritmussal való megoldást mutatjuk be.

1. lépés: A táblázat egyes oszlopaiban a legkisebb elemek: 3, 2, 3, 2, 6. Ennyivel kisebbítve a megfelelő oszlopok minden elemét a következő mátrixot kapjuk (3. ábra):

1	1	3	3	2
0	2	1	1	3
0	3	0	0	0
0	0	1	0	1
0	2	2	2	3

3. ábra

0	0	2	2	1
0	2	1	1	3
0	3	0	0	0
0	0	1	0	1
0	2	2	2	3

4. ábra

Ennek első sorában 1 a legkisebb elem, a többiben nulla. Ezért csak az első sor elemeit kell eggyel kisebbíteni, és megkapjuk a C_0 mátrixot (4. ábra).

2. lépés: Az első sor első elemét csillaggal jelöljük. A második oszlopban már csak a 4., a harmadikban pedig a 3. elem jelölhető csillaggal. A többi sorban viszont ezek után már nem tudunk nullát megjelölni (5. ábra).

Mivel így a mátrixban csak 3 csillagos nulla van, kereszttel jelöljük az 1., 2., és 3. oszlopokat (3. lépés – 6. ábra).

$$
\begin{array}{|ccccc|}
\hline
0^* & 0 & 2 & 2 & 1 \\
0 & 2 & 1 & 1 & 3 \\
0 & 3 & 0^* & 0 & 0 \\
0 & 0^* & 1 & 0 & 1 \\
0 & 2 & 2 & 2 & 3 \\
\hline
\end{array}
\qquad\qquad
\begin{array}{|ccccc|}
\multicolumn{1}{c}{x} & \multicolumn{1}{c}{x} & \multicolumn{1}{c}{x} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} \\
\hline
0^* & 0 & 2 & 2 & 1 \\
0 & 2 & 1 & 1 & 3 \\
0 & 3 & 0^* & 0 & 0 \\
0 & 0^* & 1 & 0 & 1 \\
0 & 2 & 2 & 2 & 3 \\
\hline
\end{array}
$$

5. ábra 6. ábra

Következik a 4. lépés, amelyben megjelöljük vesszővel a 0_{34}-et (a 3. sorban és 4. oszlopban lévő nullát) (7. ábra), majd az 5. lépés (8. ábra).

$$
\begin{array}{c|ccccc|}
\multicolumn{1}{c}{} & \multicolumn{1}{c}{x} & \multicolumn{1}{c}{x} & \multicolumn{1}{c}{x} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} \\
\cline{2-6}
 & 0^* & 0 & 2 & 2 & 1 \\
 & 0 & 2 & 1 & 1 & 3 \\
x & 0 & 3 & 0^* & 0' & 0 \\
 & 0 & 0^* & 1 & 0 & 1 \\
 & 0 & 2 & 2 & 2 & 3 \\
\cline{2-6}
\end{array}
\qquad\qquad
\begin{array}{c|ccccc|}
\multicolumn{1}{c}{} & \multicolumn{1}{c}{x} & \multicolumn{1}{c}{x} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} & \multicolumn{1}{c}{} \\
\cline{2-6}
 & 0^* & 0 & 2 & 2 & 1 \\
 & 0 & 2 & 1 & 1 & 3 \\
x & 0 & 3 & 0^* & 0' & 0 \\
 & 0 & 0^* & 1 & 0 & 1 \\
 & 0 & 2 & 2 & 2 & 3 \\
\cline{2-6}
\end{array}
$$

7. ábra 8. ábra

Újra a 4. lépés következik (a 0_{44} kerül megjelölésre), majd az 5. Ezután ismét 4. (0_{12}), majd 5. a sorrend (9. ábra).

$$
\begin{array}{c|ccccc|}
\cline{2-6}
x & 0^* & 0' & 2 & 2 & 1 \\
 & 0 & 2 & 1 & 1 & 3 \\
x & 0 & 3 & 0^* & 0' & 0 \\
x & 0 & 0^* & 1 & 0' & 1 \\
 & 0 & 2 & 2 & 2 & 3 \\
\cline{2-6}
\end{array}
\qquad\qquad
\begin{array}{c|ccccc|}
\cline{2-6}
x & 0^* & 0' & 2 & 2 & 1 \\
x & 0' & 2 & 1 & 1 & 3 \\
x & 0 & 3 & 0^* & 0' & 0 \\
x & 0 & 0^* & 1 & 0' & 1 \\
 & 0 & 2 & 2 & 2 & 3 \\
\cline{2-6}
\end{array}
$$

9. ábra 10. ábra

Most a 4. lépésben megjelöljük a 0_{21}-et vesszővel és az 5. lépésben megállapítjuk, hogy annak sorában nincs csillagos nulla. (10. ábra)

6. lépés: A $0'_{21}$-ből kiindulva a következő láncot állítjuk össze:

$$0'_{21},\ 0^*_{11},\ 0'_{12},\ 0^*_{42},\ 0'_{44}.$$

Elvégezzük az átjelöléseket és törlést (11. ábra), és mivel még mindig nincs öt csillagos nulla, folytatjuk a megoldást. A 3., majd a 4, lépés jön (0_{35} kerül megjelölésre), azután az 5. (12. ábra).

$$\begin{array}{|ccccc|} \hline 0 & 0^* & 2 & 2 & 1 \\ 0^* & 2 & 1 & 1 & 3 \\ 0 & 3 & 0^* & 0 & 0 \\ 0 & 0 & 1 & 0^* & 1 \\ 0 & 2 & 2 & 2 & 3 \\ \hline \end{array}$$

11. ábra

$$\begin{array}{c|ccccc|} & x & x & & x & \\ \hline & 0 & 0^* & 2 & 2 & 1 \\ & 0^* & 2 & 1 & 1 & 3 \\ x & 0 & 3 & 0^* & 0 & 0' \\ & 0 & 0 & 1 & 0^* & 1 \\ & 0 & 2 & 2 & 2 & 3 \\ \hline \end{array}$$

12. ábra

A következő 4. lépésben megállapítjuk, hogy ebben a mátrixban nincs már megjelöletlen nulla és ezért a 7. lépéssel folytatjuk a megoldást.

7. lépés: A mátrix legkisebb megjelöletlen eleme az 1. Ezt levonjuk a megjelöletlen elemekből és hozzáadjuk a kétszerjelöltekhez és a következő mátrixot kapjuk (13. ábra):

$$\begin{array}{c|ccccc|} & x & x & & x & \\ \hline & 0 & 0^* & 1 & 2 & 0 \\ & 0^* & 2 & 0 & 1 & 2 \\ x & 1 & 4 & 0^* & 1 & 0' \\ & 0 & 0 & 0 & 0^* & 0 \\ & 0 & 2 & 1 & 2 & 2 \\ \hline \end{array}$$

13. ábra

$$\begin{array}{c|ccccc|} & x & x & & x & \\ \hline & 0 & 0^* & 2 & 2 & 1 \\ & 0^* & 2 & 1 & 1 & 3 \\ x & 0 & 3 & 0^* & 0 & 0' \\ & 0 & 0 & 1 & 0^* & 1 \\ & 0 & 2 & 2 & 2 & 3 \\ \hline \end{array}$$

14. ábra

(Egy gyakorlati tanács: mikor a 7. lépésre jutunk célszerű a mátrix minden kereszttel jelölt oszlopának és sorának az elemeit egy-egy vonallal áthúzni és akkor könnyű megállapítani, hogy melyik elem a megjelöletlen, megjelölt vagy kétszerjelölt. Amelyiken ugyanis nem megy át vonal az megjelöletlen, amelyiken egy vonal megy át az megjelölt, ahol pedig két vonal kereszteződik, az kétszerjelölt (14. ábra). Ez még ellenőrzésként is szolgálhat, mert minden vesszővel és csillaggal jelölt nullának az egyszer keresztülhúzott elemek közt kell lennie, különben hibáztunk az eljárás során.)

A 13. ábrán lévő mátrixra most sorban a következő lépéseket alkalmazzuk: 4. (0_{15}), 5., 4.(0_{45}), 5., 4.(0_{23}), 5., 4.(0_{51}). A mátrix ezután így néz ki (15. ábra):

$$\begin{array}{c} x \\ x \\ x \\ x \\ \end{array}\boxed{\begin{array}{ccccc} 0 & 0^* & 1 & 2 & 0' \\ 0^* & 2 & 0' & 1 & 2 \\ 1 & 4 & 0^* & 1 & 0' \\ 0 & 0 & 0 & 0^* & 0' \\ 0' & 2 & 1 & 2 & 2 \end{array}} \qquad \boxed{\begin{array}{ccccc} 0 & 0^* & 1 & 2 & 0 \\ 0 & 2 & 0^* & 1 & 2 \\ 1 & 4 & 0 & 1 & 0^* \\ 0 & 0 & 0 & 0^* & 0 \\ 0^* & 2 & 1 & 2 & 2 \end{array}}$$

15. ábra 16. ábra

Mivel a 0_{51} sorában nincs csillagos nulla, az 5. lépés után a 6.-ra térünk át. A $0'_{51}$ a következő láncot adja:

$$0'_{51}, 0^*_{21}, 0'_{23}, 0^*_{33}, 0'_{35}.$$

Átjelölés és törlés után kapjuk a 16. ábrán látható mátrixot. Abban már 5 csillagos nulla van, ami azt jelenti, hogy a feladatot megoldottuk.

Mivel a csillagos nullák a 0_{12}, 0_{23}, 0_{35}, 0_{44} és a 0_{51} ez azt jelenti, hogy az első mérnöknek a 2. tervet, a másodiknak a 3.-at a harmadiknak az 5.-et, a 4.-nek a 4.-et és az ötödiknek az első tervet kell adni. Ha az induló-táblázatban összegezzük az így szükséges napokat, 18 napot kapunk, ami a legkevesebb amennyit az öt tervre fordítani kell.

A fejezet elején említett sakkprobléma szintén megoldható ezzel az eljárással. A megoldás szerint a bástyákat a következő mezőkre kell helyezni: a1, b3, c4, d8, e6, f2, g7 és h5. Az általuk letakart számok összege 22. Ezt az eredményt kevesen találnák meg két-három próbálkozással „gyalog"-módszerrel!

3. SZÁLLÍTÁSI PROBLÉMA

Ha utána néznénk, hogy az operációkutatás matematikai módszerei közül melyeket használják ma a gyakorlatban, láthatnánk, hogy a szállítási problémákat megoldó módszerek az elsők között vannak. Ennek oka, hogy a szállítás a temelésnek egyik alapvető része és annak ésszerű megszervezése igen jelentős megtakarításokat eredményezhet. Azonkívül a szállítási problémákat könnyű közérthető módon megfogalmazni és az optimalizálás eredményei is mindenki számára „kézzelfoghatóak".

Ime példaként egy jellegzetes szállítási probléma:
Egy nagyüzemnek 4 telephelye van a község területén, jelöljük azokat A, B, C, és D-vel. Ezeken a telepeken egyfajta terméket gyártanak, mégpedig az A telepen naponta 50 mázsát, a B-n 80-at, a C-n 30-at és a D-n napi 100 mázsát. Az üzemnek 5 elárusítóhelye (üzlete) van a községben (a, b, c, d és e), azoknak napi szükséglete sorrendben 50, 40, 60, 50 és 60 mázsa.

Az üzletekbe naponta szállítják az árut a termelőhelyekről és nyilvánvaló, hogy a rosszul megszervezett szállítás naponta jelentős többletkiadást eredményezne. Ezért szeretné az üzem vezetősége megállapítani azt, hogy melyik telepekről melyik üzleteket kell ellátni áruval, hogy a szállítási költségek a lehető legkisebbek legyenek. Ehhez rendelkeznek egy táblázattal, melyben feltüntették, hogy mennyibe kerül egy mázsa áru elszállítása adott teleptől adott üzletig (17. ábra).

	a	b	c	d	e
A	5	4	3	4	2
B	2	1	3	3	4
C	6	1	4	4	5
D	7	2	5	2	∞

17. ábra

Ennyi adat éppen elegendő, hogy az alábbiakban ismertetett eljárással megoldhassuk a felvetett problémát. (Ami igen leegyszerűsített képe a hasonló gazdasági problémáknak, de arra megfelelő, hogy rajta keresztül megismerkedjünk a szállítási problémákat megoldó módszerek egyik legjobbikával.)

3.1 *A szállítási probléma matematikai modellje*

Ahhoz, hogy a szállítási problémát megoldó eljárásokat megérteni és alkalmazni tudjuk, feltétlenül meg kell ismerkednünk a probléma matematikai modelljével. A megoldási eljárást ugyanis a matematikai modell segítségével adjuk meg, mert az csak úgy lehet általános.

A szállítási problémához kell egy úgynevezett költségmátrix, amely m sorból és n oszlopból áll és C-vel jelöljük (tehát $C = \|c_{ij}\|_{mn}$). Az előbb említett példa táblázata egy ilyen költségmátrix, ahol $m = 4$ és $n = 5$.

A megoldás eredményét egy szintén m sorból és n oszlopból álló X mátrixban ($X = \|x_{ij}\|_{mn}$) kapjuk. Ennek x_{ij} eleme azt mutatja az adott példánál, hogy az i-edik telepről a j-edik üzletbe hány mázsányi árut kell szállítani, hogy a szállítás összköltsége a lehető legkisebb legyen. A két mátrix ugyanazon helyen lévő (ugyanazon indexü) elemeit megfelelő elemeknek nevezzük.

A szállítási problémához tartozik két feltételvektor is, melyeket a-val és b-vel jelöljük, elemeiket pedig $a_1, a_2, \ldots, a_m$-mel, illetve $b_1, b_2, \ldots, b_n$ nel. Az előbbi példában $a = (50, 80, 30, 100)$, azaz a telepek napi termelése, és $b = (50, 40, 60, 50, 60)$, az üzletek napi szükséglete.

Láthatjuk, hogy a $c_{ij}x_{ij}$ szorzat azt mutatja, hogy mennyibe kerül x_{ij} mennyiség elszállítása i helyről j helyre. Minket az érdekel, hogy a szállítások összköltsége a lehető legkisebb legyen és ezt így írjuk fel:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}x_{ij} \longrightarrow \min.$$

Ügyelni kell arra is, hogy egy-egy telepről csak annyit szállítsunk, amenynyit ott termelnek (azaz: $\sum_{j=1}^{n} x_{ij}=a_i$, $i=1,2, \ldots,m$), egy-egy üzletbe annyit, amennyit igényel($\sum_{i=1}^{m} x_{ij}=b_j$, $j=1, 2,\ldots, n$) és, hogy nem szállíthatunk negatív mennyiségű árut ($x_{ij} \geq 0$). Mindezt összegezve, a szállítási probléma matematikai modellje a következő:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}x_{ij} \longrightarrow \min \qquad (1)$$

$$\sum_{j=1}^{n} x_{ij}=a_i \qquad i=1,2,\ldots,m \qquad (2)$$

$$\sum_{i=1}^{m} x_{ij}=b_j \qquad j=1,2,\ldots,n \qquad (3)$$

$$x_{ij} \geq 0 \quad i=1,2,\ldots,m; \quad j=1,2,\ldots,n \qquad (4)$$

Bebizonyítható, hogy ha $\sum_{i=1}^{m} a_i=\sum_{j=1}^{n} b_j$, akkor a szállítási problémának mindig van megoldása és azt véges számú lépésen belül meg is kapjuk.

A gyakorlati életben előforduló esetek miatt célszerű ezt a szállítási modellt kissé kibővített alakban használni. Megtörténik ugyanis, hogy vannak bizonyos előre megadott kikötések, hogy erről és erről a telepről ide és ide nem kívánunk, vagy nincs módunkban szállítani. (Így például az általunk adott feladatnál a D telep nem szállíthat az e üzletbe, amit ∞ jellel jelöltünk.)

A matematikai modellben ezt úgy fogalmazzuk meg, hogy a megfelelő i, j indexpárokra megköveteljük, hogy x_{ij} nulla legyen. Ezen i, j párokat egy T halmazban (a tiltott szállítások halmaza) tároljuk. Az így kibővített szállítási probléma matematikai modellje az (1).-(4) képletek mellett tartalmazza még a következőt: $x_{ij}=0$, ha $(i, j) \in T$.

Mivel a gyakorlati problémákban igen sokszor szükség van erre a kibővítésre, ezért a következőkben az ilyen típusú feladatokra adjuk meg a megoldási eljárást, azzal, hogy az a „szabályos" szállítási problémákra éppúgy érvényes.

A szállítási problémákat igen eredményesen oldhatjuk meg a magyar módszerrel. A megoldási eljárás hasonlít a hozzárendelési problémák megoldási módjához, csak valamivel bonyolultabb. Ha a második fejezetben ismertetett eljárást jól áttanulmányozzuk, akkor könnyebben boldogulunk ezzel is.

A szállítási problémát megoldó eljárás is 7 lépésből áll, melyek közül az első kettő előkészítő jellegű, a többit pedig többször is alkalmazzuk a megoldás folyamán.

A megoldandó feladat alapadatai a költségmátrix (C), amely a tiltott szállítások helyén ∞ jelet tartalmaz, és a két feltételvektor (a, b), amelyekre teljesül a

$$\sum_{i=1}^{m} a_i = \sum_{j=1}^{n} b_j \qquad (5)$$

feltétel. A feladat megoldását a megoldásmátrixban (X) kapjuk, vagy pedig arra a megállapításra jutunk, hogy a nem korrekt tiltások miatt a feladat nem oldható meg. Mint már mondottuk, az (5) feltételt kielégítő, tiltásokat nem tartalmazó szállítási problémának mindig van megoldása.

És most lássuk a megoldási eljárást.

1. lépés: A C mátrix minden oszlopában megkeressük a legkisebb értékű elemet, és azt az értéket levonjuk azon oszlop minden eleméből. (Kivételt képeznek a ∞ jellel jelölt elemek, amelyek állandóan ∞-k maradnak, és a legkisebb elem kiválasztása során nem kell őket figyelembe venni az 1. lépés folyamán.) Az így kapott új mátrix minden oszlopában lesz bár egy nulla. Most az új mátrix minden sorában keressük meg a legkisebb elemet és azt vonjuk ki a sor minden eleméből. A kapott mátrixot C_0-val jelöljük. Ennek minden eleme nemnegatív, és minden sorában és minden oszlopában van nulla értékű elem.

Ezután létrehozzuk a $d=(d_1, d_2, \ldots, d_n)$ és $e=(e_1, e_2, \ldots, e_m)$ segédvektorokat. Az e olyan hosszú mint az a, és kezdetben az értékeik is megegyeznek (azaz: $e_i=a_i$, $i=1, 2, \ldots, m$), ugyanaz érvényes a b és d vektorokra is ($d_j=b_j$, $j= 1, 2, \ldots, n$).

2. lépés: Kiszámítjuk az X megoldásmátrix kezdőértékét a következőképpen:

Legyen a C_0 mátrix első oszlopában egy nulla értékű elem az $(i_i, 1)$ helyen (i_1-edik sor, 1. oszlop). Megnézzük, hogy e_{i_1} vagy d_1 a kisebb, és azt az értéket az $(i_1, 1)$ helyre írjuk az X mátrixban. Ezenkívül ha e_{i_1} a kisebb, akkor az e vektorban e_{i_1} új értéke nulla, míg a d vektorban d_1 új értéke $d_1 - e_{i_1}$; ha pedig d_1 a kisebb, akkor d_1 (új)$=0$, e_{i_1} (új)$=e_{i_1} - d_1$.

Ha azt, hogy e_{i_1} és d_1 közül a kisebbet vesszük, úgy jelöljük, hogy min (e_{i_1}, d_1), akkor az előbbieket így írhatjuk röviden:

$$x_{i_1 1}=\min(e_{i_1}, d_1),\ e_{i_1}(\text{új})=e_{i_1}-\min(e_{i_1}, d_1),\ d_1(\text{új})=d_1-\min(e_{i_1}, d_1).$$

Természetesen most már e_{i_1} és d_1 régi értékét el is felejthetjük.

Ezután megnézzük, hogy van-e még nulla a C_0 mátrix első oszlopában. Ha van, akkor megismételjük az előbbi eljárást, tehát, ha a nulla az $(i_2, 1)$ helyen van, akkor $x_{i_2 1}=\min(e_{i_2}, d_1)$ (ahol d_1 egyszer más kisebbítődött és lehet, hogy most már nulla, és akkor $x_{i_2 1}=0$), $e_{i_2}(\text{új})=e_{i_2}-\min(e_{i_2}, d_1)$, $d_1(\text{új})=d_1-\min(e_{i_2}, d_1)$.

Ha az első oszlopban már minden nullát sorra kerítettünk, akkor rátérünk a második oszlop nulláira, majd a többire. A C_0 mátrix minden nullájára alkalmazzuk az említett eljárást, nem felejtve el, hogy az e és d vektorok elemei ezáltal állandóan csökkennek (ám negatívvá sohasem válnak!). A lépés végén az X mátrix még kitöltetlen helyeire nullákat írunk. Ha jól számoltunk, akkor a 2. lépés végén

$$e_i=a_i-\sum_{j=1}^{n}x_{ij} \qquad i=1,2,\dots,m$$

$$d_j=b_j-\sum_{i=1}^{m}x_{ij} \qquad j=1,2,\dots,n.$$

Matematikai eszközökkel a 2. lépést (az X mátrix kezdőértékének kiszámítását) a következőképpen írhatjuk fel:

ha a C_0 mátrix j-edik oszlopában r_j nulla van és a k-adikat, amely az i-edik sorban helyezkedik el, i_{kj}-vel jelöljük, akkor

$$x_{ij}=\begin{cases}0 & \text{ha } i\neq i_{k,j} \quad k=1,2,\dots,r_j\\ \min\left(a_i-\sum_{p=1}^{j-1}x_{ip},\ b_j-\sum_{q=1}^{i-1}x_{qj}\right) & i=i_{k,j} \text{ valamelyik k-ra } (k=1,2,\dots,r_j)\end{cases}$$

ahol $i=1, 2, \dots, m$; $j=1, 2, \dots, n$ és $\sum_{i=1}^{0}=0$.

3. lépés: Kiszámítjuk a

$$D=\sum_{i=1}^{m}a_i+\sum_{j=1}^{n}b_j-\sum_{i=1}^{m}\sum_{j=1}^{n}x_{ij}$$

értéket. Ha $D=0$, akkor a feladatot megoldottuk, és a megoldást az X mátrix tartalmazza. Ha pedig $D\geq 0$, akkor keresztet teszünk a C_0 mátrix minden olyan oszlopa fölé, ahol a d segédvektorban nulla van (a d vektornak ugyanis annyi eleme van, ahány oszlopa a C_0 mátrixnak, és a j-edik eleme a j-edik oszlopnak „társa"). Ezentúl a C_0 mátrix azon elemeit, amelyeknek sora előtt, vagy oszlopa felett kereszt van, közös néven megjelölt elemeknek nevezzük, a többit pedig megjelöletlennek. A megoldást a 4. lépéssel folytatjuk.

4. lépés: Megnézzük, hogy a C_0 mátrixban van-e megjelöletlen nulla. Ha nincs, akkor a 7. lépésre térünk, ha van, akkor kiválasztunk egyet. Legyen az a mátrix i_1-edik sorában és j_1-edik oszlopában ($0_{i_1 j_1}$). Azt plusszal, a sorát pedig kereszttel jelöljük. Ezután megnézzük hogy, az e vektor i_1-edik eleme nulla-e. Ha igen, akkor az 5. lépéssel, ha nem, akkor a 6.-kal folytatjuk a megoldást.

5. lépés: Végignézzük a C_0 mátrix i_1-edik sorának nulla értékű elemeit. Ha azok közül valamelyik megjelölt oszlopban van, és az X mátrixban a neki megfelelő elem pozitív, akkor a nullát mínusszal jelöljük, az oszlopa fölül pedig töröljük a keresztet. Miután a sor összes nulláját sorra kerítettük, visszatérünk a 4. lépésre.

6. lépés: Erre a lépésre akkor jutunk, ha a 4. lépésben olyan jelöletlen nullát találunk (legyen az $0_{i_1 j_1}$), melyre $e_{i1} \geq 0$. Ezt a nullát már ott megjelöltük plusszal. Ha az oszlopban nincs mínusszal jelölt nulla, akkor $x_{i_1 j_1}$ új értéke $x_{i_1 j_1} + \min(e_{i_1}, d_{j_1})$, $e_{i_1}(\text{új}) = e_{i_1} - \min(e_{i_1}, d_{j_1})$ és $d_{j_1}(\text{új} = d_{j_1} - \min(e_{i_1}, d_{j_1})$. A C_0 mátrixban töröljük a plusszokat, mínuszokat, kereszteket és visszatérünk a 3. lépésre.

Ha viszont a j_1 oszlopban van mínusszal jelölt nulla, legyen a $0^{-}_{i_2 j_1}$, akkor az i_2-ik sorban biztos van egy (és csak egy) plusszal jelölt nulla ($0^{+}_{i_2 j_2}$). Ennek oszlopában megint lehet egy mínusszal jelölt nulla, és ha van, akkor annak sorában van egy plusszal jelölt is. Bebizonyítható, hogy minden plusszal jelölt nulla egyértelműen meghatároz egy ilyen sorozatot:

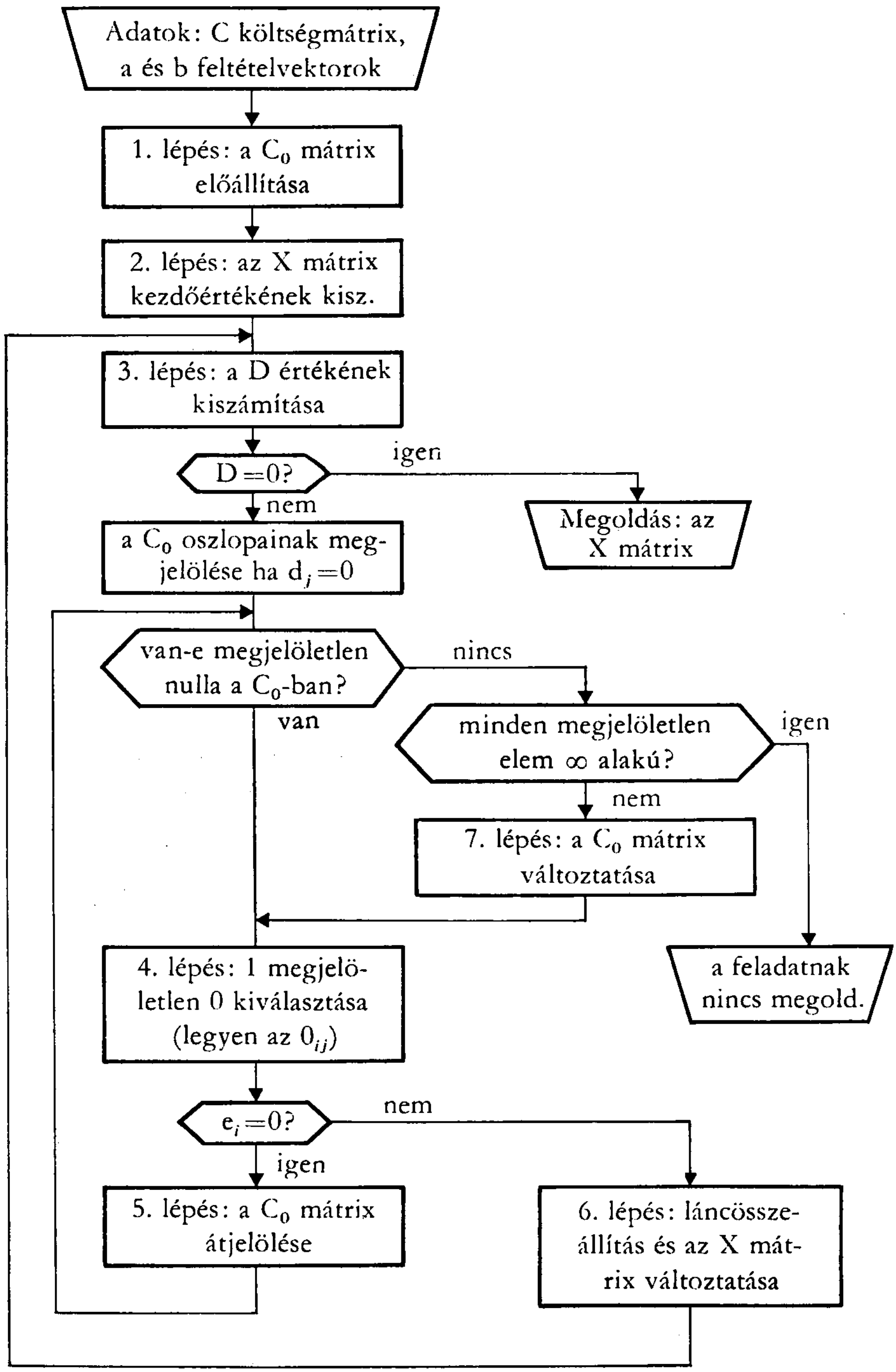
$$0^{+}_{i_1 j_1},\ 0^{-}_{i_2 j_1},\ 0^{+}_{i_2 j_2},\ \ldots\ 0^{-}_{i_s j_{s=1}},\ 0^{+}_{i_s j_s}.$$

Ezt a sorozatot láncnak nevezzük, és adott kezdőelemre ($0^{+}_{i_1 j_1}$) csak egy lánc állítható össze, amely mindig plusszal jelölt nullával végződik, és véges sok elemből áll.

Most megkeressük a legkisebbet a következő értékek közül: e_{i_1}, d_{j_s} és az X mátrixnak azon elemei, amelyek a láncban szereplő mínusszal jelölt nulláknak felelnek meg (azokkal azonos indexüek). Jelöljük ezt az elemet t-vel, tehát

$$t = \min_{1 \leq k \leq s-1}(e_{i_1}, d_{j_s}, x_{i_{k+1} j_k})$$

Ezzel az értékkel kisebbítjük e_{i_1}-et, d_{j_s}-t és az X mátrix azon elemeit, amelyek a lánc mínusszal jelölt elemeinek felelnek meg. Ugyanennyivel növeljük X azon elemeit, amelyek a lánc plusszal jelölt nulláinak felelnek meg. Az X mátrix többi eleme változatlan marad, és az összes elem továbbra sem negatív, éppúgy, mint az e és d vektorok. A C_0 mátrixban törölünk mindenféle jelölést és visszatérünk a 3. lépésre.



Adatok: C költségmátrix, a és b feltételvektorok
1. lépés: a C_0 mátrix előállítása
2. lépés: az X mátrix kezdőértékének kisz.
3. lépés: a D értékének kiszámítása
$D=0$?
igen
nem
Megoldás: az X mátrix
a C_0 oszlopainak megjelölése ha $d_j=0$
van-e megjelöletlen nulla a C_0-ban?
nincs
van
minden megjelöletlen elem ∞ alakú?
igen
nem
7. lépés: a C_0 mátrix változtatása
a feladatnak nincs megold.
4. lépés: 1 megjelöletlen 0 kiválasztása (legyen az 0_{ij})
$e_i=0$?
nem
igen
5. lépés: a C_0 mátrix átjelölése
6. lépés: láncösszeállítás és az X mátrix változtatása

18. ábra

7. lépés: Ha a 4. lépés folyamán nem akadunk megjelöletlen nullára, akkor jutunk erre a lépésre. Amennyiben a C_0 mátrix összes megjelöletlen eleme ∞ alakú, akkor az eljárást beszüntetjük, mivel a nem helyesen meghatározott, tiltott szállítások miatt a szállítási probléma megoldhatatlan.

Ha viszont vannak ∞-től különböző megjelöletlen elemek is, akkor azok mind pozitívek. Azok közül megkeressük a legkisebb értékűt. Azzal az értékkel csökkentjük a megjelöletlen elemek értékét, és növeljük az olyan elemekét, amelyek megjelölt oszlop és megjelölt sor metszetében vannak. A C_0 mátrix többi eleme változatlan marad, sőt még a jelölését is megtartja (az összes plusszal és mínusszal jelölt nulla ezek között van). A megoldást a 4. lépéssel folytatjuk.

Ezzel a szállítási probléma megoldási eljárását bemutattuk. Hogy az egyes lépések egymás közötti kapcsolata még érthetőbb legyen, lerajzoltuk az eljárás folyamatábráját is (18. ábra), amelyben az egyes lépéseket csak röviden jellemezzük, mivel a részletes leírásukat már megadtuk.

3.3 *Egy példa megoldása*

Ha az ismertetett eljárással oldjuk meg a fejezet elején említett példát, akkor a következőképpen jutunk el az eredményhez (itt a megoldás menetének csak egyes mozzanatait mutatjuk be:

A feladat adatai a költségmátrix (19. ábra) és az a =(50, 80, 30, 100) és b =(50, 40, 60, 50, 60) feltételvektorok.

5	4	3	4	2
2	1	3	3	4
6	1	4	4	5
7	2	5	2	∞

19. ábra

3	3	0	2	0
0	0	0	1	2
4	0	1	2	3
5	1	2	0	∞

20. ábra

Az első lépésben a C_0 mátrixot kapjuk meg (20. ábra), valamint az e = =(50, 80, 30, 100) és d =(50, 40, 60, 50, 60) segédvektorokat. A második lépésben az X mátrix kezdőértékét számítjuk ki (21. ábra), miközben a segédvektorok is változnak. Értékük a lépés végén: e =(0, 0, 20, 50), d = (0, 0, 10, 0, 60).

0	0	50	0	0
50	30	0	0	0
0	10	0	0	0
0	0	0	50	0

21. ábra

				x	
x	3	3	0^+	1	0
x	0^-	0^-	0^+	0	2
	4	0^+	1	1	3
	4	0	1	1	∞

22. ábra

Ezután a 3., 4.(0_{13}), 5., 4.(0_{23}), 5., majd a 4.(0_{32}) lépés kerül sorra (22. ábra). Így eljutunk a 0_{32} elemhez, és megállapítjuk, hogy $e_3=20\neq 0$. Ezért a 6. lépésre megyünk és 0_{32}-ből kiindulva a következő láncot kapjuk: 0^+_{32}, 0^-_{22}, 0^+_{23}. Mivel $e_3=20$, $d_3=10$ és $x_{22}=30$ közül a legkisebb érték 10, csökkentjük tízzel e_3-at, d_3-at és x_{22}-t, növeljük tízzel x_{32}-t és x_{23}-at. Így megkapjuk az X mátrix új értékét (23. ábra) és $e=(0, 0, 10, 50)$, $d=(0, 0, 0, 0, 60)$.

Folytatva az eljárást még négyszer jutunk a 6. lépésre, háromszor pedig a 7-re, és a végén a következő megoldást kapjuk (24. ábra):

0	0	50	0	0
50	20	10	0	0
0	20	0	0	0
0	0	0	50	0

23. ábra

0	0	0	0	50
50	0	30	0	0
0	20	0	0	10
0	20	30	50	0

24. ábra

Innen leolvashatjuk, hogy például az A telepről az e üzletbe 50 mázsa árut kell szállítani, a B-ből az a-ba szintén 50-et, a c-be pedig 30-at, stb. Az összes szállítás költsége 650.

4. A MAGYAR MÓDSZER MÁS ALKALMAZÁSI LEHETŐSÉGEI

Az előző fejezetekben bemutattunk két feladattípust, amelyek megoldásánál alkalmazni lehet a magyar módszert, és meg is mutattuk, hogy hogyan.

Felmerülhet a kérdés, hogy miért választottuk külön a hozzárendelési problémát a szállításitól. Valójában a hozzárendelési probléma egy speciális esete a szállításinak, ahol a feltételvektorok minden elemének értéke egy, a költségmátrix pedig négyzetes. Ezek szerint megoldható lenne a 3. fejezetben közölt algoritmussal. Nos, a különválasztás azért történt, mert a hozzárendelési problémát megoldó eljárás lényegesen egyszerűbb, könnyebben megérthető és elsajátítható, és azon keresztül azután a másikat is könnyebben felfoghatjuk. Azonkívül igen sok hozzárendelési probléma jelentkezik a gyakorlatban, és célszerű rájuk az egyszerűbb algoritmust alkalmazni.

Érdekesség kedvéért megemlítjük, hogy a szállítási probléma is felírható (és megoldható), mint hozzárendelési, de mivel akkor a költségmátrix és feltételvektorok nagyon megnövekszenek (például a 3. fejezeti példánál a költségmátrix 260×260-as lesz), ezért nem oldjuk a szállítási problémákat azzal az eljárással.

Az említett feladattípusok mellett még számos problémát oldhatunk meg magyar módszerrel. Ezek közül mutatunk itt be néhány egyszerűbbet.

4.1 *Maximum feladatok*

Megtörténik, hogy egy hozzárendelési problémánál a speciálisan kiválasztott elemektől nem azt várjuk el, hogy összegük a lehető legkisebb, hanem azt, hogy a legnagyobb legyen. Ha egy szállítási problémánál a „költségmátrixban" nem az egyes szállítások költségét, hanem mondjuk, a szállítással megvalósított jövedelmet tüntetjük fel, akkor sem a minimum, hanem a lehető legnagyobb összeg, a maximum érdekel bennünket. Az ilyen feladatokat nevezzük maximum feladatoknak.

Ezeknek matematikai modellje a következő:

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}x_{ij} \longrightarrow \max \qquad (1)$$

$$\sum_{j=1}^{n} x_{ij} = a_i \qquad i = 1,2,\ldots,m$$

$$\sum_{i=1}^{m} x_{ij} = b_j \qquad j = 1,2,\ldots,n$$

$$x_{ij} \geq 0 \qquad i = 1,2,\ldots,m \quad j = 1,2,\ldots,n.$$

Tudjuk azonban azt, hogy ha egy számsorozat minden elemének megváltoztatjuk az előjelét, akkor az addigi legnagyobb a legkisebb lesz és fordítva. Éppen ezért, amikor (1) teljesül, ugyanakkor $\Sigma\,\Sigma(-c_{ij})x_{ij} \rightarrow \min$ is teljesül. Ebből látható, hogy egyszerűen a költségmátrix minden elemét be kell szorozni mínussz eggyel (megváltoztatni az előjelüket), és az új költségmátrixra alkalmazni a 3. fejezetben ismertetett eljárást. Ha megkapjuk az eredményt, az összköltséget természetesen az eredeti költségmátrix segítségével számítjuk ki.

Ha például a második fejezetben említett sakkfeladványban azt kérjük, hogy a bástyák által letakart mezők összege a lehető legnagyobb legyen, akkor ezen a módon könnyen eljutunk az eredményhez, ami: a2, b7, c8, d5, e4, f6, g3, h1 és az összeg 63.

4.2 *Szállítási probléma, ahol* $\Sigma a_i \neq \Sigma b_j$

Az előző fejezetben feltételeztük, hogy a szállítási problémánál mindig teljesül a $\Sigma a_i = \Sigma b_j$ feltétel, és úgy is írtuk le a megoldási eljárást. Ez a konkrét feladatban azt jelentette, hogy amennyit az üzem termel, annyi kell az elárusító üzleteknek is. Ilyen „egyensúly" azonban ritkán van. Gyakoribb az, hogy vagy az üzem termel többet a keresletnél, vagy az üzletek igényelnek többet, mint amennyi elkészül.

A magyar módszert ezeknek a problémáknak megoldásánál is alkalmazni lehet. Tegyük fel például, hogy említett feladatunknál az üzem raktárra is termel. Kérdés, hogy melyik telepeken maradjon ott az árufölösleg, és hogyan alakuljon a szállítás, hogy a szállítási költségek továbbra is a legkisebbek legyenek.

Ilyen esetben a költségmátrixot kibővítjük még egy oszloppal, amibe nullákat írunk (hisz amit nem szállítunk el, annak nincs szállítási költsége sem). Ezenkívül a b feltételvektort is meghosszabbítjuk eggyel, és annak az elemnek az értéke $\Sigma a_i - \Sigma b_j$. Az így kibővített mátrixszal és vektorokkal a már ismertetett eljárás szerint dolgozunk. A megoldásként kapott X mátrix utolsó oszlopában kapott értékek mutatják, hogy melyik telepen mennyi áru marad.

Ha $\Sigma b_j > \Sigma a_i$, ugyanígy járunk el, csak akkor a mátrixot egy új sorral bővítjük, és az a vektort egy új elemmel.

Ha például az említett 3. fejezeti feladatban a D telep 100 helyett csak 60 mázsát termel, akkor egyes üzletek nem kapnak elég árut. De melyek és mennyivel kevesebbet? A feladat kibővített költségmátrixa a 25. ábrán látható, feltételvektorai pedig a következők: a =(50, 80, 30, 60, 40), b =(50, 40, 60, 50, 60).

5	4	3	4	2
2	1	3	3	4
6	1	4	4	5
7	2	5	2	
0	0	0	0	0

25. ábra

0	0	0	0	50
50	0	30	0	0
0	30	0	0	0
0	10	0	50	0
0	0	30	0	10

26. ábra

A megoldásból (26. ábra) láthatjuk többek között, hogy a c üzlet 30 mázsával, az e pedig 10 mázsával kevesebb árut kap.

4.3 *Előre meghatározott szállítások*

Egyes esetekben a következőket kötjük ki egy-egy szállítási problémával kapcsolatban: úgy kell meghatározni, hogy honnan hová és mennyit szállítsunk, hogy a szállítási költségek minimálisak legyenek, de egy (vagy több) adott helyről egy (vagy több) adott helyre, adott mennyiséget kell szállítani.

Így például kiköthetjük, hogy a már sokat emlegetett 3. fejezeti példában az A telepről a c üzletbe feltétlenül 20 mázsa árut kell szállítani, a többit pedig úgy elosztani, hogy a szállítási összköltség a legkisebb legyen.

Ilyen esetben is magyar módszerrel oldjuk meg a feladatot, csak a megoldás kezdete előtt kissé átalakítjuk a mátrixok és vektorok tartalmát. Az X mátrixba a megfelelő helyre beírjuk az előlátott mennyiséget, a költségmátrixba ∞ jelet teszünk ugyanoda (hogy azon az útvonalon többet ne szállíthatsunk), a feltételvektorokban pedig csökkentjük a megfelelő elemeket.

Konkrétan, a példánkban x_{13} 20 lesz (27. ábra), c_{13} ∞ (28. ábra) és $a=(30, 80, 30, 100)$, $b=(50, 40, 40, 50, 60)$.

0	0	20	0	0
0	0	0	0	0
0	0	0	0	0
0	0	0	0	0

27. ábra

5	4	∞	4	2
2	1	3	3	4
6	1	4	4	5
7	2	5	2	∞

28. ábra

Ilyen kezdő adatok mellett a megoldás:

0	0	20	0	30
50	0	10	0	20
0	10	10	0	10
0	30	20	50	0

a költség pedig 690. (A 20 mázsa árunak A-ból c-be való szállítása ezek szerint 40 egységgel drágítja a szállítást, hisz anélkül, a 3. fejezetben láttuk, 650 volt a költség.)

4.4 *A szűk keresztmetszet problémája*

Ez a feladattípus lényegesen eltér az eddig ismertetettektől, ezért egy konkrét példán mutatjuk be.

Tegyük fel, hogy a 2. fejezetben említett tervezőiroda-probléma táblázata most a következő:

	tervek				
	I	II	III	IV	V
1. mérnök	2	4	3	4	5
2. „	3	4	2	1	4
3. „	6	4	3	2	6
4. „	3	1	5	2	8
5. „	2	6	4	7	6

Ennek megoldása 1-V, 2-IV, 3-III, 4-II, 5-I és a szükséges tervezői napok száma 12.

Legyen az iroda érdeke az is, hogy az összes terv elkészüljön 4 napon belül. Akkor a kapott megoldás nem jó, hisz az 1. mérnöknek 5 nap kell az ötödik terv elkészítéséhez, és feltételeztük, hogy nem segíthet neki senki. Olyan megoldást kell tehát keresni, amelynél egyik mérnök sem dolgozik 4 napnál tovább a kapott terven, és amely ilyen féltételek mellett a legjobb (legkevesebb napot igényel). (Tehát a feladat „szük keresztmetszete" a 4 napos határidő.)

Ez a probléma is megoldható a magyar módszer ismertetett eljárásaival, mégpedig úgy, hogy az adattáblázatban először minden 4-nél nagyobb számot ∞-nel helyettesítünk (tiltott hozzárendelés), és csak azután kezdünk hozá a megoldáshoz.

Az indulóhálózat tehát így néz ki:

2	4	3	4	∞
3	4	2	1	4
∞	4	3	2	∞
3	1	∞	2	∞
2	∞	4	∞	∞

A megoldás pedig: 1-IV, 2-V, 3-III, 4-II, 5-I, ami összesen 14 napot jelent (az előbbi 12-vel szemben), de így minden mérnök legtöbb 4 nap alatt kész a munkájával.

A megoldáshoz alkalmazhatjuk a 3. fejezetben ismertetett eljárást, azzal hogy a és b elemei eggyel egyenlőek, de ugyanúgy be lehet vezetni a hozzárendelési probléma megoldási eljárásába is a tiltott hozzárendelés fogalmát (a ∞ jelet), csak az 1. és 7. lépésben kell figyelembe venni, hogy ∞-hez sem hozzáadni, sem belőle kivonni nem szabad.

A szűk keresztmetszet problémája különösen olyan szállításoknál érdekes és hasznos, ahol nemcsak az a lényeges, hogy a szállítás mint egész a leggazdaságosabb legyen, hanem az egyes szállításokra is vannak bizonyos korlátozó feltételek (főleg időkorlátok).

Irodalomjegyzék

[1]Egerváry Jenő: Mátrixok kombinatorikus tulajdonságairól, Matematikai Fizikai Lapok, 38, (1931.).

[2] Judin, D. B., E. G. Goljštejn: Zadači i metodi linejnogo programmirovanija, Izdanie vtoroe, Izdateljstvo ,,Sovetskoe Radio", Moskva, 1964.

[3] Kaufmann, A.: Az optimális programozás, Műszaki könyvkiadó, Budapest, 1964.

[4] Kaufmann, A., R. Faure: Bevezetés az operációkutatásba, Műszaki Könyvkiadó, Budapest 1969.

[5] Kisimre István: Mađarska metoda u programiranju, magistarski rad, Beograd, 1974

[6] König Dénes: Gráfok és Mátrixok, Matematikai Fizikai Lapok, 38, (1931.).

[7] Kreko, Dr Bela: Linearno programiranje, Savremena administracija, Beograd, 1966.

Rezime

Primena mađarske metode

Američki matematičar Kuhn je 1955. godine za rešavanje problema pridruživanja koristio stavove mađarskih matematičara Königa [6] i Egerváry-ja [1] i zbog toga svoj postupak nazvao ,,mađarskom metodom". Kasnije je Egervány pokazao da je mađarska metoda pogodna i za rešavanje transportnih problema.

Cela problematika mađarske metode obrađena je sa teorijske i praktične strane u radu[5].

U ovom radu opisan je način primene mađarske metode za rešavanje problema pridruživanja i transportnog problema. Postupak rešavanja dat je u algoritamskom obliku, korak po korak, tako da je pogodan za ručno rešavanje problema manjeg obima, kao i za programiranje za elektronski računar. Priloženi izrađeni primeri čine opisane metode još preglednijem.

U četvrtom poglavlju rada date su još neke mogućnosti za primenu mađarske metode na probleme operacionog istraživanja (problem maksimuma, problem uskog grla, itd.).

Resümee

Die Verwendung der ungarischen Methode

Der amerikanische Matematiker Kuhn verwand im Jahre 1955 fürs Lösen des Anschlußproblems di Methoden der ungarischen Matematiker König[6] und Egervary[1] und daher nannte er sein Verfahren ,,Ungarische Methode". Später zeigte Egervary, dass die ungarische Methode auch für das Lösen der Transportproblemen geeignet ist.

Die ganze Problematik de ungarischen Methode ist sowohl von der theoretischen, als auch von der praktischen Seite bearbeitet.

In dieser Arbeit wird die Verwendungsweise der ungarischen Methode für das Lösen des Anschluss- und Transportproblems beschrieben. Das Lösungsverfahren wird in der algorythmischen Form gegeben, Schritt für Schritt, so daß es sowohl für die manuelle

Lösung eines Problems kleineren Umfangs, als auch für das Programieren der elektronischen Rechenmaschine geeignet ist. Die beifolgenden ausgearbeiteten Beispiele machen die beschriebene Methode noch deutlicher.

In dem vierten Teil der Arbeit sind noch einige Möglichkeiten für die Verwendung der ungarischen Methode gegeben, lösen des Problems der Operationsuntersuchungen (Problem des Maximums, Problem des Engpasses).