Járatszerkesztési feladatok

DR. BENKŐ JÁNOS
Agrártudományi Egyetem, Gödöllő Mezőgazdasági Géptani Intézet

A járat alatt a logisztikában általában a járművek meghatározott, több állomást érintő útvonalát értjük, ami lehet menetrendszerű vagy eseti. A járatok tervezésére irányuló tevékenységet pedig járatszerkesztésnek nevezzük. A járatszerkesztés célját és megoldási módját tekintve, a konkrét feladattól függően, nagyon sokféle lehet. Egyes esetekben az útvonalak adottak, és a menetrend összeállítása a feladat, más esetekben csak alkalmi optimális útvonalakat kell meghatározni, de gyakran fordul elő a két alapeset kombinációja is. A járatszerkesztésre ezért egységes megoldási módszer nem adható. Mindig a konkrét feladat ismeretében kell megkeresni vagy kifejleszteni azt az eljárást, amelytől megoldást remélhetünk. E tanulmányban két, gyakran előforduló problémával foglalkozunk. Az egyik a szállítójárművek, targoncák stb. üres futásának minimalizálása, a másik a központi telephelyről indított, korlátozott számú és kapacitású járművek útvonalának optimalizálása.

Járatszerkesztés az üres menetek költségének minimalizálásával

Az üres menetek minimalizálásán alapuló járatszerkesztési probléma modelljét és algoritmusát az ATUKI megbízásából *Krekó Béla* és *Szántó Emil* dolgozta ki [4]. E helyen az algoritmus továbbfejlesztett változatát mutatjuk be [1].

Ez a sajátos probléma általában több telephellyel rendelkező üzemben merülhet fel, amikor olyan járatokat kell tervezni, amelyek a telephelyeket a szükséges gyakorisággal érintik és egyúttal a legrövidebbek is. A gyakorlatban sokféle módon megfogalmazható feladat alapmodellje a következő:

Adott n számú állomás $A_1, A_2, \ldots, A_n$. Az állomások egyidejűleg egyaránt lehetnek feladók és megrendelők. A j-edik állomás, mint megrendelő y_{ij} számú rakományt igényel az i-edik feladótól. Ezenkívül ismeretes az állomások közötti szállítási távolság vagy a fajlagos szállítási költség (c_{ij}).

A cél olyan útvonal megtervezése, amely a lehető legkevesebb üres futás mellett biztosítja, hogy a feladó állomásokról az áru a megrendelő állomásokra kerüljön. Könnyen belátható, hogy a kitűzött célt akkor érjük el, ha az üres meneteket úgy osztjuk el az egyes szállítási viszonylatokra, hogy az x_{ij} üres menetek száma és a c_{ij} fajlagos szállítási költségek szorzatainak összege minimális, azaz

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

Az elmondottakból az is kiderül, hogy ha valamennyi állomáson azonos a megrendelt és feladott rakományok száma, akkor nincs üres menet, és a feladatnak nincs értelme. A legkedvezőtlenebb esetben viszont minden rakott menetet üres menet követ.

A feladat ma már klasszikusnak tekinthető megoldása két lépésből áll. A feladatot szállítási problémaként kezelve, az első lépésben az üres meneteket programozzuk. A második lépés a járatkapcsolás, vagyis az üres és a rakott menetek valamilyen előírás szerinti összekötése.

A klasszikus módszer szerint az első lépésben egy olyan szállítási feladatot kell megoldani, amelyben a készletek a feladóhelyekről indított, az igények pedig a megrendelőhelyekre érkező járatoknak felelnek meg. A költségmátrix főátlójában csak 0 elemek szerepelnek, így a megoldás során a rakott menetek a főátlóra kerülnek. A főátlón kívüli relációkra programozott menetek az üres menetek számát adják, amelyek a mátrix transzponálásával kerülnek a tényleges helyükre. Ezután a járatkapcsoláshoz szükséges ún. munkamátrixot a rakott és az üres menetek mátrixának egymásra illesztésével kapjuk meg.

A továbbiakban az ismertetett eljáráshoz hasonló, de kevesebb számítást igénylő algoritmust mutatunk be [1]. Tekintsük először az algoritmust:

(1) $$y_{ij} \geq 0,\ x_{ij} \geq 0,\ i = j = 1,2,\ldots,n,$$

(2) $$d_k = \sum_{j=1}^{n} y_{kj} - \sum_{i=1}^{n} y_{ik},\ k = 1,2,\ldots,n,$$

(3) $$r_k = \begin{cases} d_k,\ ha\, d_k > 0 \\ 0,\ ha\, d_k \leq 0 \end{cases},$$

(4) $$t_k = \begin{cases} |d_k|,\ ha\, d_k < 0 \\ 0,\ ha\, d_k \geq 0 \end{cases},$$

(5) $$\sum_{j=1}^{n} x_{kj} = t_k,$$

(6) $$\sum_{i=1}^{n} x_{ik} = r_k,$$

(7) $$c_{ij} = M,\ ha\ i = j,$$

(8) $$\sum_{i=1}^{n} \sum_{j=1}^{n} c_{ij} x_{ij} \to \min,$$

ahol: n az állomások száma,
y_{ij} a rakott menetek száma az ij relációban,
x_{ij} az üres menetek száma az ij relációban,
d_k a k-adik állomásról indított és a k-adik állomásra érkező rakott menetek számának a különbsége,
t_k a k-adik állomásról indított üres menetek száma,
r_k a k-adik állomásra érkező üres menetek száma,
c_{ij} a fajlagos szállítási költség.

Az algoritmust összehasonlítva az eredeti eljárással, az alapvető különbség az üres menetek számának meghatározásában mutatkozik, és mint látni fogjuk, ez lényegesen egyszerűsíti azok elosztását is.

Az algoritmus szerint először a (2), (3), (4) összefüggésekkel kiszámítjuk az üres menetek számát. A rakott menetek **Y** mátrixát soronként és oszloponként összegezzük, majd a k-adik sorösszegből levonjuk a k-adik oszlop összegét. A d_k differenciák előjele a (3) és (4) feltételeknek megfelelően megmutatja, hogy az üres menetet a k-adik állomásról kell-e indítani, vagy a k-adik állomásra kell teljesíteni. A d_k>0 azt jelenti, hogy a k-adik állomásról több rakott menetet indítanak, mint amennyi oda érkezik. Értelemszerűen ezért a k-adik állomásra r_k=d_k alkalommal kell üresen menni. Ha a d_k<0, akkor fordított a helyzet, és a k-adik állomásról t_k=d_k alkalommal üresen kell indítania járatot. A d_k=0 esetben magától értetődően nincs üres járat, vagyis r_k=t_k=0.

Az üres menetek számának meghatározása után valamelyik ismert algoritmussal megoldjuk az (5), (6), (7) feltételeket, és a (8) célfüggvényt kielégítő szállítási feladatot. A továbbfejlesztett eljárás előnye itt domborodik ki, mivel a **C** költségmátrix azon sorait, illetve oszlopait, ahol t_k=0, illetve r_k=0, elhagyhatjuk. Így az eredetinél jóval kisebb méretű szállítási feladatot kell megoldani. A legkedvezőtlenebb esetben is n-ről n/2-re csökken a költségmátrix rendje.

Az ismertetett előnyök szemléltetésére tekintsünk egy, az irodalomból ismert példát [3], így az érdeklődő olvasónak alkalma nyílik a régi és az új eljárás idő- és számításigényének összehasonlítására.

Egy üzemben öt munkahely, P_1, P_2,...,P_5 között a szállítást targoncával kívánjuk megoldani. Az üzemek között teljesítendő rakott menetek számát tartalmazó **Y** mátrix:

H o v a

H		P_1	P_2	P_3	P_4	P_5	$\Sigma_j y_{kj}$
o	P_1	–	2	3	1	–	6
n	P_2	–	–	2	3	4	9
n	P_3	–	4	–	1	2	7
a	P_4	2	2	1	–	1	6
n	P_5	–	1	2	4	–	7
	$\Sigma_i y_{ik}$	2	9	8	9	7	35

A fajlagos szállítási költségek mátrixa legyen a távolságmátrix, mivel a szállítási költség a távolság lineáris függvénye:

H o v a

H		P_1	P_2	P_3	P_4	P_5
o	P_1	0	8	10	5	15
n	P_2	8	0	4	3	10
n	P_3	10	4	0	9	8
a	P_4	5	3	9	0	6
n	P_5	15	10	8	6	0

Képezzük a (2) összefüggéseknek megfelelő különbségeket, és határozzuk meg a (6.3), (6.4) feltételek alapján az elosztandó üres meneteket:

d_1=4	r_1=4	t_1=0
d_2=0	r_2=0	t_2=0
d_3=-1	r_3=0	t_3=1
d_4=-3	r_4=0	t_4=3
d_5=0	r_5=0	t_5=0

Ezután azokat a sorokat és oszlopokat elhagyva, ahol t_k=0, illetve r_k=0, írjuk fel a szállítási feladat induló tábláját:

		H	*o*	*v*	*a*		
H		P_1	P_2	P_3	P_4	P_5	t_k
o	P_1						
n	P_2						
n	P_3	10^{1}					1
a	P_4	5^{3}					3
n	P_5						
	r_k	4					

Amint látható a szállítási feladat egy 2×1-es mátrixra redukálódott, amelynek megoldását az induló táblában azonnal feltüntettük:

$$x_{31}=1,\ x_{41}=3.$$

Az üres menetek költsége:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}\, x_{ij} = 25$$

A járatkapcsoláshoz szükséges munkamátrixot a rakott és üres menetek mátrixainak összegzésével nyerjük: **Y'=Y+X**

		H	*o*	*v*	*a*	
H		P_1	P_2	P_3	P_4	P_5
o	P_1	-	2	3	1	-
n	P_2	-	-	2	3	4
n	P_3	1	4	-	1	2
a	P_4	5	2	1	-	1
n	P_5	-	1	2	4	-

A szállítás összes költsége:

$$\sum_{i=1}^{n}\sum_{j=1}^{n} c_{ij}\,(y_{ij} + x_{ij}) = 255\,,$$

amelyből 25 egység az üres menetek költsége.

A járatkapcsolás intuitív feladat, amelynek során különböző korlátozó feltételeket is figyelembe lehet venni. Például, ha korlátozott a járművek által naponta megtehető távolságegységek száma, akkor a szállítást több járművel kell megoldani. Esetünkben egy targonca naponta 54 távolságegységet képes teljesíteni, így a feladatot 5 targoncával tudjuk elvégezni.

		H	*o*	*v*	*a*	
H		P_1	P_2	P_3	P_4	P_5
o	P_1	-	2	3	1	-
n	P_2	-	-	2	3	4
n	P_3	1	4	-	1	2
a	P_4	5	2	1	-	1
n	P_5	-	1	2	4	-

Indítsuk az I. jelű targoncát a P_1 üzemből a P_3 -ba, ezt egy vízszintes és egy függőleges szakasszal jelöljük, miközben az y'_{13} értékét 3-ról 2-re csökkentjük. Ezzel 1 rakott menetet teljesítettünk a P_1 -ből a P_3 -ba. A targonca által megtett út:

$$s_{\mathrm{I}}=c_{13}=10.$$

Ezt követően irányítsuk a targoncát a P_3 -ból P_2 -be, amit ismét egy vízszintes és egy függőleges szakasszal jelölünk. Elvégezve az előző műveleteket:

$$y'_{32}=4-1=3,$$

$$s_{\mathrm{I}}=s_{\mathrm{I}}+c_{32}=10+4=14.$$

1. táblázat

Targonca	Útvonal	Távolság
I	$P_1 \to P_3 \to P_2 \to P_5 \to P_4 \to P_1 \to P_2 \to P_4 \Rightarrow P_1$	51
II	$P_1 \to P_3 \to P_2 \to P_5 \to P_4 \to P_1 \to P_2 \to P_4 \Rightarrow P_1$	51
III	$P_1 \to P_3 \to P_2 \to P_3 \to P_5 \to P_3 \to P_2 \to P_5 \to P_4$	54
IV	$P_4 \to P_2 \to P_3 \to P_4 \to P_3 \to P_5 \to P_4 \Rightarrow P_1 \to P_4$	49
V	$P_4 \to P_2 \to P_5 \to P_2 \to P_4 \to P_5 \to P_3 \Rightarrow P_1$	50
Összesen		255

A továbbiakban a műveleteket nem részletezve, a bemutatott módon addig kötjük össze az üzemeket, amíg a targonca által megtehető 54 távolságegységet el nem érjük. Ezután kijelöljük a II., III. stb. targoncák útvonalait.

A végső megoldást a *1. táblázatban* foglaltuk össze, ahol a $\to$ jellel a rakott, a $\Rightarrow$ jellel pedig az üres meneteket jelöltük.

Járműkapacitással korlátozott egycentrumos járatszerkesztés

Az elosztási vagy felvásárlási tevékenységet folytató vállalatoknál, raktári bázisokon stb. gyakori feladat a járművek menetrendjének, útvonalának kijelölése. A probléma nagyon sokféle formában jelentkezhet, de az egyik leggyakoribb eset az, amikor a vállalatnak egy központi helyről, a centrumból kell ellátnia a fogyasztókat vagy megrendelőket, és az igényeinek kielégítésére a vállalat véges számú valamint kapacitású járműparkkal rendelkezik.

A feladat megoldásának legegyszerűbb módja az, hogy minden fogyasztóhoz egyedi járművel szállítjuk ki a megrendelt mennyiséget. A megrendelt mennyiségek azonban általában nem kötik le a járművek kapacitását, ami lehetővé teszi az utak összevonását, ún. járatok szervezését. Természetesen az összevonások csak bizonyos megszorítások mellett eszközölhetők, így:

- a megrendelők igényét a célállomásokon maradéktalanul ki kell elégíteni,
- a szállított mennyiség nem lépheti túl a járathoz rendelt jármű kapacitását,
- a jármű által megtett út vagy a szállítási idő nem léphet túl egy előre meghatározott megengedett értéket.

Tekintve, hogy az utak összevonására általában nagyon sokféle lehetőség van, felvetődik a kérdés, található-e az intuitív döntéseknél jobb, az optimálishoz közel álló megoldás. A továbbiakban egy ilyen kézi számolásra és programozásra egyaránt alkalmas algoritmusra teszünk javaslatot, amelynek lépéseit egy mintapéldán mutatjuk be, majd az eredményeket általánosítva összefoglaljuk az eljárást [2].

Legyen adott egy debreceni székhelyű vállalat, P_0, amelynek Debrecenből kell az ország különböző pontjaira (Budapest, Győr, stb.) települt P_1, P_2,...,P_n fogyasztókhoz meghatározott q_1, q_2,...,q_n, mennyiségű terméket eljuttatnia. A fogyasztókat, telephelyeiket és az igényeiket az *.2. táblázat* tartalmazza.

2. táblázat

Megrendelések

Fogyasztó	P_1	P_2	P_3	P_4	P_5	P_6
Telephely	Budapest	Győr	Miskolc	Nyíregyháza	Szombathely	Zalaegerszeg
q_i[t]	4	3	4	2	3	2

A vállalat 7 tehergépkocsival rendelkezik, a járműpark összetételét a *3. táblázatban* foglaltuk össze.

3. táblázat

Járműpark

Teherbírás [t]	Darabszám
10	2
6	5

Ismeretesek továbbá a P_0 centrum és a P_1, P_2,...,P_n célállomások közötti legrövidebb utak c_{0j}, c_{i0}, és c_{ij} (*4. táblázat*). A feladat a megrendelt mennyiségek kiszállítása a centrumból a fogyasztókhoz a rendelkezésre álló járműpark felhasználásával úgy, hogy a járművek által megtett összes út a lehető legrövidebb legyen. A járművek által megtehető utat és a szállítási időt illetően nem élünk megszorításokkal, ami azonban az eljárás lényegét nem érinti. Az összevonások szempontjából csak a járművek kapacitása jelentsen korlátot.

4. táblázat

Távolsági mátrix [km]

	Debrecen P_0	Budapest P_1	Győr P_2	Miskolc P_3	Nyíregyháza P_4	Szombathely P_5	Zalaegerszeg P_6
Debrecen P_0	0	225	348	99	50	453	447
Budapest P_1	225	0	123	167	237	228	222
Győr P_2	348	123	0	290	360	105	169
Miskolc P_3	99	167	290	0	87	395	389
Nyíregyháza P_4	50	237	360	87	0	465	459
Szombathely P5	453	228	105	395	465	0	65
Zalaegerszeg P_6	447	222	169	389	459	65	0

A javasolt megoldás alapja az a triviális tény, hogy az utak összevonása a gyakorlatban általában út megtakarítást eredményez, ami nagyon egyszerűen belátható. Ha a P_0-P_i-P_0 és a P_0-P_j-P_0 utakat egyesítjük P_0-P_i-P_j-P_0 járattá, akkor az elérhető megtakarítás:

$$s_{ij} = 2c_{0i} + 2c_{0j} - (c_{0i} + c_{ij} + c_{0j}) = c_{0i} + c_{0j} - c_{ij}.$$

Az is világos, hogy minél nagyobb az adott járathoz rendelt jármű kapacitása, annál több út összevonására van lehetőség. Ezért a járatok szervezésekor a nagyobb járműveknek prioritást kell biztosítani.

Természetesen indokoltan vetődhet fel a kérdés, hogy az útmegtakarítás minden esetben költségmegtakarítást jelent-e. A kérdés indoka egyrészt az hogy, az utak összevonása szállításimunka (tkm) növekedéssel járhat, másrészt a nagyobb kapacitású járművek fajlagos üzemeltetési költsége általában nagyobb, mint a kisebbeké. Mégis a kérdésre egyértelmű igennel válaszolhatunk. A szállítási munkaigény növekedése ugyanis nem jelent számottevő költségnövekedést, mivel a terhelt és a terheletlen jármű költsége között nincs lényeges különbség. A nagyobb és a kisebb teherbírású járművek fajlagos üzemeltetési költségében mutatkozó eltérés pedig nem tartozik e feladat lényegéhez, tekintve, hogy e feladatban meglévő jármű állományt kell optimálisan kihasználni. Más kérdés az, ha a járműállomány összetételéről kell dönteni, akkor magától értetődően mérlegelni kell a fajlagos költségeket is.

A gyakorlatban előfordulhat, hogy egy-egy fogyasztó megrendelése meghaladja a legnagyobb jármű kapacitását. Ebben az esetben a fuvart megosztjuk két vagy több jármű között. Tekintettel arra, hogy az így telítődő járműveknél nincs további lehetőség út összevonásra, a telített járműveket nem vizsgáljuk, csak a megosztás utáni maradékokat kezeljük megrendelésként.

A megoldás első lépése az ún. **megtakarítási mátrix** felállítása, amelynek elemei az előzőek alapján azt mutatják, hogy mennyi idő vagy út megtakarítás érhető el azzal, ha a P_0-P_i-P_0 és a P_0-P_j-P_0 utakat egyesítjük P_0-P_i-P_j-P_0. A teljes megtakarítási mátrixot az *5. táblázat* tartalmazza, ami esetünkben egy szimmetrikus mátrix, de ennek a megoldás során nincs jelentősége. Például, a P_0-P_1-P_0 és a P_0-P_2-P_0 utak összevonásakor a P_2 sor és a P_1 oszlop metszéspontjában, a megtakarítás:

$$s_{21} = c_{01} + c_{02} - c_{21} = 225 + 348 - 123 = 450\,.$$

5. táblázat

Megtakarítási mátrix [km]

	P_1	P_2	P_3	P_4	P_5	P_6
P_1	0	450	157	38	450	450
P_2	450	0	157	38	696	626
P_3	157	157	0	62	157	157
P_4	38	38	62	0	38	38
P_5	450	696	157	38	0	835
P_6	450	626	157	38	835	0

A továbbiakban a számításokat a szemléletesség és az érthetőség megkönnyítése érdekében két táblázatban végezzük. Az első tábla a megtakarítási mátrixot és a megrendelt mennyiségeket, a második t_1, t_2,...t_l terhelhetőség szerint csökkenő sorba rendezett a J_1, J_2,...,J_l járműveket és m_k terhelésüket tartalmazza. Az induló táblák az alábbiak:

	P_1	P_2	P_3	P_4	P_5	P_6	q_i
P_1	0	450	157	38	450	450	4
P_2	450	0	157	38	696	626	3
P_3	157	157	0	62	157	157	4
P_4	38	38	62	0	38	38	2
P_5	450	696	157	38	0	835	3
P_6	450	626	157	38	835	0	2

Jármű	J_1	J_2	J_3	J_4	J_5	J_6	J_7
t_k [t]	10	10	6	6	6	6	6
m_k [t]	0	0	0	0	0	0	0

A következő lépésként megkeressük az s_{ij} mátrix legnagyobb elemét, ezt jelöljük s_{xy}-nal, azaz

$$s_{xy} = \max\{s_{ij} \mid i = 1,2,\ldots,n;\, j = 1,2,\ldots,n\}.$$

A második táblából pedig kiválasztjuk az első járművet, ennek teherbírása t_k, majd megvizsgáljuk, hogy a jármű kapacitása lehetővé teszi-e a két út összevonását, azaz ha

$$q_x + q_y \le t_k\,,$$

akkor az utak összevonhatók.

A példában

$$s_{xy} = s_{56} = \max\{s_{ij} \mid i = 1,2,\ldots,n; j = 1,2,\ldots,n\} = 835,$$

és a

$$q_5 + q_6 = 3 + 2 \le t_1 = 10,$$

vagyis a P_0-P_5-P_0 és a P_0-P_6-P_0 utak P_0-P_5-P_6-P_0 járattá egyesíthetők. Az elérhető útmegtakarítás 835 km. A P_5 és P_6 fogyasztók igényeit ezzel kielégítettnek tekinthetjük. Az s_{56} elem ismételt kiválasztását és a járat záródását úgy akadályozzuk meg, hogy az első táblában lefedjük az 5-dik sort és a 6-dik oszlopot, valamint az s_{65} elem helyére 0-t írunk. Ezzel egyidejűleg a q_5 és q_6 értékét csökkentsük 0-ra, a második táblában pedig az J_1 járművet megterheljük q_5+q_6=5-tel. Az adminisztrációk után a táblázatok:

	P_1	P_2	P_3	P_4	P_5	P_6	q_i
P_1	0	450	157	38	450	450	4
P_2	450	0	157	38	696	626	3
P_3	157	157	0	62	157	157	4
P_4	38	38	62	0	38	38	2
P_5	450	696	157	38	0	835	0
P_6	450	626	157	38	0	0	0

Jármű	J_1	J_2	J_3	J_4	J_5	J_6	J_7
t_k [t]	10	10	6	6	6	6	6
m_k [t]	5	0	0	0	0	0	0

Az összevonást követően, mivel a J_1 jármű még nem telített, a P_5-P_6 járatot próbáljuk P_5-be menő vagy P_6-ból induló úttal bővíteni. Ezért maximális s_{ij} elemet keresünk a 6-dik sorban és az 5-dik oszlopban, miközben a lefedett elemeket figyelmen kívül hagyjuk:

$$\max\{s_{6j} \mid j = 1,2,\ldots,n\} = s_{62} = 626,$$
$$\max\{s_{i5} \mid i = 1,2,\ldots,n\} = s_{25} = 696.$$

Mivel s_{25}>s_{62}, továbbá a

$$q_2 + q_5 + q_6 = 3 + 3 + 2 \leq t_1 = 10,$$

ezért a P_5-P_6 járatot elölről bővítjük. Az összevonás után az első járat állomásai P_0-P_2-P_5-P_6-P_0 lesznek. Az első táblában fedjük le a 2-dik sort és az 5-dik oszlopot, valamint az s_{62} elem helyére írjunk 0-t. Ezzel egyidejűleg a q_2 értékét csökkentsük 0-ra, a második táblában pedig az J_1 jármű terhelését növeljük q_2=3-mal.

	P_1	P_2	P_3	P_4	P_5	P_6	q_i
P_1	0	450	157	38	450	450	4
P_2	450	0	157	38	696	626	0
P_3	157	157	0	62	157	157	4
P_4	38	38	62	0	38	38	2
P_5	450	696	157	38	0	835	0
P_6	450	0	157	38	0	0	0

Jármű	J_1	J_2	J_3	J_4	J_5	J_6	J_7
t_k [t]	10	10	6	6	6	6	6
m_k [t]	8	0	0	0	0	0	0

A következő lépésben kíséreljük meg a P_2-P_5-P_6 járatot P_2-be menő vagy P_6-ból induló úttal bővíteni. Most maximális s_{ij} elemet a 6-dik sorban és a 2-dik oszlopban keresünk:

$$\max\{s_{6j} \mid j = 1,2,\ldots,n\} = s_{61} = 450,$$
$$\max\{s_{i2} \mid i = 1,2,\ldots,n\} = s_{12} = 450.$$

Az P_1 állomás egyaránt kapcsolható lenne P_2 elé vagy P_6 mögé, ha ezt a J_1 jármű teherbírása lehetővé tenné. Mivel azonban a

$$q_1 + q_2 + q_5 + q_6 = 4 + 3 + 3 + 2 > t_1 = 10,$$

az összevonás nem lehetséges. A P_2-P_5-P_6 járatot ezért lezárjuk, amit az első táblában a 6-dik sor és a 2-dik oszlop, a második táblában pedig az első oszlop lefedésével jelzünk.

	P_1	P_2	P_3	P_4	P_5	P_6	q_i
P_1	0	450	157	38	450	450	4
P_2	450	0	157	38	696	626	0
P_3	157	157	0	62	157	157	4
P_4	38	38	62	0	38	38	2
P_5	450	696	157	38	0	835	0
P_6	450	0	157	38	0	0	0

Jármű	J_1	J_2	J_3	J_4	J_5	J_6	J_7
t_k [t]	10	10	6	6	6	6	6
m_k [t]	8	0	0	0	0	0	0

Az első járat így véglegesen kialakult: P_0-P_2-P_5-P_6-P_0, az elért útmegtakarítás 1531 km.

A második járat indításához ismét megkeressük az s_{ij} mátrix fedetlen elemei között a legnagyobbat, a második táblából pedig kiválasztjuk a második járművet:

$$s_{xy} = s_{13} = \max\{s_{ij} \mid i = 1,2,\ldots,n\,;\, j = 1,2,\ldots,n\} = 157,$$

és a

$$q_1 + q_3 = 4 + 4 \le t_2 = 10\,.$$

Elvégezzük a szokásos adminisztrációt: az első táblában lefedjük az első sort és a 3-dik oszlopot, az s_{31} elem helyére 0-t írunk, a q_1 és q_3 értékét 0-ra csökkentjük, a második táblában a J_2 jármű terhelését q_1+q_3=8-cal növeljük.

	P_1	P_2	P_3	P_4	P_5	P_6	q_i
P_1	0	450	157	38	450	450	0
P_2	450	0	157	38	696	626	0
P_3	0	157	0	62	157	157	0
P_4	38	38	62	0	38	38	2
P_5	450	696	157	38	0	835	0
P_6	450	0	157	38	0	0	0

Jármű	J_1	J_2	J_3	J_4	J_5	J_6	J_7
t_k [t]	10	10	6	6	6	6	6
m_k [t]	8	8	0	0	0	0	0

A P_1-P_3 járatot próbáljuk meg P_1-be menő vagy P_3-ból induló úttal bővíteni. Ezért maximális s_{ij} elemet keresünk a 3-dik sorban és az első oszlopban:

$$\max\{s_{3j} \mid j = 1,2,\ldots,n\} = s_{34} = 62\,,$$

$$\max\{s_{i1} \mid i = 1,2,\ldots,n\} = s_{41} = 38\,.$$

Mivel s_{34}>s_{41}, továbbá a

$$q_1 + q_3 + q_4 = 4 + 4 + 2 = t_2 = 10\,,$$

ezért a P_1-P_3 járatot hátulról bővítjük. Az összevonás után a második járat által érintett pontok P_0-P_1-P_3-P_4-P_0 lesznek. Az első táblában lefedjük az 3-dik sort és az 4-dik oszlopot, valamint az s_{41} elem helyére 0-t írunk. A q_4 értékét 0-ra csökkentjük, a második táblában pedig az J_2 jármű terhelését növeljük q_4=2-vel. Az elért útmegtakarítás 219 km.

	P_1	P_2	P_3	P_4	P_5	P_6	q_i
P_1	0	450	157	38	450	450	0
P_2	450	0	157	38	696	626	0
P_3	0	157	0	62	157	157	0
P_4	0	38	62	0	38	38	0
P_5	450	696	157	38	0	835	0
P_6	450	0	157	38	0	0	0

Jármű	J_1	J_2	J_3	J_4	J_5	J_6	J_7
t_k [t]	10	10	6	6	6	6	6
m_k [t]	8	10	0	0	0	0	0

További összevonásra már nincs lehetőség. A szállítást a leggazdaságosabban a két járattal, a 10 t teherbírású járművekkel lehet lebonyolítani. A járatok által érintett pontok:

$$P_0\text{-}P_2\text{-}P_5\text{-}P_6\text{-}P_0,$$

$$P_0\text{-}P_1\text{-}P_3\text{-}P_4\text{-}P_0.$$

A példa után fogalmazzuk meg kicsit általánosabban a problémát és foglaljuk össze az algoritmust:

A szállítási pontokat (a centrumot és a fogyasztókat) szimbolizálja a $G=(P,E)$ irányítatlan gráf, amely a P szállítási pontok és az E élek halmazából áll. A P halmaz elemeit jelölje p_i (i=0,1,2,...,n), az E halmaz elemeit pedig e_{ij} (i=j=0,1,2,...,n). Ha a p_i össze van kötve p_j-vel, akkor e_{ij}=1, különben e_{ij}=0. Az e_{ij}-hez rendelt távolsági mátrix c_{ij} elemei jelentsék a szállítási pontok közötti legrövidebb utakat. Ha e_{ij}=0, akkor c_{ij}=M, ahol M végtelen nagy szám. A P halmaz p_i elemeihez rendeljük a megrendelésvektor q_i elemeit. Megállapodás szerint p_0 jelentse a centrumot.

Legyen J a rendelkezésre álló járművek halmaza, amelynek minden j_k (k=1,2,... ,l) eleméhez hozzárendeljük a járműveket jellemző t_k teherbírás- és az m_k terhelésvektorokat.

1. Rendezzük a J halmazt a t_k teherbírás szerint csökkenő sorrendbe.

2. A kapacitáskorlát miatt összevonásra alkalmatlan utakat a vizsgálatból kivonjuk. Ehhez képezzük a $q_i^{(u)} / t_k$ hányadosokat minden $i>0$-ra és k-ra. Ha a

$$q_i^{(u)} / t_k > 1 \text{ és } m_k = 0,$$

akkor a

$$q_i^{(u+1)} := q_i^{(u)} - t_k \;\; \textit{és} \;\; m_k := t_k,$$

és vesszük a következő járművet, vagyis a k index értékét növeljük eggyel. Ha

$$q_i^{(u)} / t_k < 1,$$

akkor vesszük a következő megrendelőt, azaz az i értékét növeljük eggyel. A 2. lépést addig ismételjük, amíg a $q_i^{(u)} / t_k$>1. (A felső indexben az u a ciklusváltozó.) Végül azokat a szállítási pontokat, ahol q_i=0 elhagyjuk, illetve az elhagyott pontoknak megfelelő sorokat és oszlopokat a c_{ij} mátrixból töröljük.

3. Az új c_{ij} távolsági mátrixból az

$$s_{ij} = c_{0i} + c_{0j} - c_{ij}, \text{ ha } e_{ij} = 1,$$

illetve az

$$s_{ij} = 0, \text{ ha } e_{ij} = 0,$$

képletekkel kiszámítjuk az s_{ij} megtakarítási mátrix elemeit.

4. Az s_{ij} mátrix fedetlen elemei között megkeressük a legnagyobbat:

$$s_{xy} = \max\{s_{ij} \mid i = 1,2,...3;\, j = 1,2,...,n\}.$$

Ha találtunk 0-nál nagyobb elemet, akkor az 5. lépéssel folytatjuk, különben az eljárás befejeződött.

5. A rendezett J halmazból vegyük a következő j_k járművet, amelynél m_k=0, és megvizsgáljuk a p_0-p_x-p_0 és a p_0-p_y-p_0 utak összevonásának lehetőségét:

Ha a $q_x+q_y\leq t_k$, akkor az utak összevonhatók. Lefedjük az x-edik sort és az y-adik oszlopot, majd végrehajtjuk a következő változtatásokat:

$$m_k := q_x + q_y,$$

$$q_x := 0,\ q_y := 0,\ s_{yx} := 0,$$

és a 6. lépéssel folytatjuk.

Ha a $q_x+q_y\geq t_k$, akkor nincs lehetőség az utak összevonására. Lefedjük az y-adik oszlopot, majd végrehajtjuk a következő változtatásokat:

$$m_k := q_y,$$

$$q_y := 0,$$

és visszatérünk a 4. lépéshez.

6. A p_x-p_y járatot megpróbáljuk p_x-be menő vagy p_y-ból induló úttal bővíteni. Ezért megkeressük az y-adik sor és az x-edik oszlop maximális elemeit, majd ezek közül először a nagyobbat választjuk:

$$\max\{s_{yj} \mid j = 1,2,\ldots,n\} = s_{yj^*},$$

$$\max\{s_{ix} \mid i = 1,2,\ldots,n\} = s_{i^*x}\ .$$

Ha az $s_{yj^*} \geq s_{i^*x}$, és a $t_k \geq m_k + q_{j^*}$, akkor a járatot hátulról, az y-ból induló és a j^*-ba menő úttal bővítjük. Lefedjük az y-adik sort és a j^*-edik oszlopot, majd

$$m_k := m_k + q_{j^*},$$

$$q_{j^*} := 0,\ s_{j^*x} := 0,\ y := j^*,$$

és megismételjük a 6. lépést.

Ha az $s_{yj^*} < s_{i^*x}$, és a $t_k \geq m_k + q_{i^*}$, akkor a járatot elölről, az i^*-ból induló és az x-be menő úttal bővítjük. Lefedjük az i^*-edik sort és a x-edik oszlopot, majd

$$m_k := m_k + q_{i^*},$$

$$q_{i^*} := 0,\ s_{yi^*} := 0,\ x := i^*,$$

és megismételjük a 6. lépést.

Különben nincs lehetőség az útösszevonásra, ezért lezárjuk a járatot: lefedjük az y-adik sort és az x-edik oszlopot, majd visszatérünk a 4. lépéshez.

A mintapéldában a járatok szerkesztésekor csak a járművek kapacitáskorlátait vettük figyelembe, de mint utaltunk rá annak sincs akadálya, hogy egyéb megszorításokat tegyünk, például, hogy a jármű által megtehető utat vagy időt a kapacitáskorláthoz hasonlóan az útösszevonás feltételeként vizsgáljuk. Az sem szorul különösebb magyarázatra, hogy az algoritmus nemcsak elosztási, hanem gyűjtőjáratok, továbbá személyszállítást végző autóbuszok járatainak szerkesztésére is alkalmas.

IRODALOM

1. **Benkő J.**: Algoritmus a járatszerkesztési probléma számításigényének csökkentésére. A+CS, 32. évf. 5. sz. 1987. 134-136 p.

2. **Benkő J.**: Járatszerkesztés korlátozott járműkapacitással. Járművek, Építőipari és Mező-gazdasági Gépek, 40. évf. 10. sz. 1993.

3. Felföldi L.: Anyagmozgatási kézikönyv. Műszaki Könyvkiadó, Budapest, 1976.

4. Szántó E.: A körutazási és járatszerkesztési modell. KÖZDOK, 1972.

Publikálva: Logisztikai Évkönyv ’94 MLE 1994. 35-48 p