JÁNDY GÉZA

A LINEÁRIS PROGRAMOZÁS, MINT A KÖZLEKEDÉS TUDOMÁNYOS GAZDASÁGI ÜZEMVEZETÉSÉNEK ÚJ MATEMATIKAI MÓDSZERE*

Az állandóan fokozódó igények és a korlátozott anyagi erőforrások a gazdasági élet jelenségeinek alaposabb megismerésére, tudományos feltárására kényszerítenek. Ez a folyamat pedig szükségszerűen a matematika újabb előretörését hozza magával. Amíg korábban a matematikának csaknem kizárólagos alkalmazási területe a fizika volt, ma már mind mélyebbre hatol a statisztikában és a gazdaságtudományban is. A matematika terjedésének forradalmi lendületet ad egyrészt az új matematikai módszerek, másrészt az elektronikus számológépek megjelenése. Ezek segítségével olyan problémák lesznek hozzáférhetők, illetve a problémák olyan mélységeibe hatolhatunk be, amelyek korábban elérhetetlenek voltak, vagy azért, mert a matematika meglevő kifejezési eszközei nem voltak elégségesek és megfelelők a probléma megfogalmazására, vagy mert a megoldás olyan óriási volumenű számítást kívánt, amelynek elvégzésére gondolni sem lehetett. A műszaki tervezésnél is megfigyelhető, hogy a tudatosságra és a gazdaságosságra törekvés miként támaszt mind nagyobb és nagyobb matematikai igényeket. A tervezésekhez tartozó számítások mind terjedelmesebbé válnak, hogy a lejátszódó fizikai folyamat minél pontosabban megfogható és követhető legyen, mert az az előfeltétele a gazdaságosabb és műszakilag is helyesebb megoldásoknak.

A mai élet hasonló igényekkel kezd fellépni a közgazdaság és az üzemvezetés területén is. Pontosabban és egzakt módon kívánjuk látni a gazdasági összefüggéseket és a különböző hatások vagy intézkedések gazdasági következményeit. És sok esetben, bár csak kvantitatíve, a matematika és az elektronikus számológépek alkalmazása már meg is adják erre a lehetőséget. Nyilvánvaló, hogy a szocialista tervgazdálkodásban a tudatosság érvényesülése döntő jelentőségű. A szocialista tervgazdálkodás feladatát akkor tudja helyesen betölteni, ha alkalmazza azokat a felkutatott és továbbfejlesztett tudományos módszereket, amelyek a gazdasági élet összefüggéseihez mind közelebb visznek, vagy amelyek a gazdasági és üzemvezetési intézkedéseket mind hatékonyabbá teszik.

Ebben a tanulmányban a közlekedési vállalatok vagy szállító részlegek egyes — a lineáris programozással megoldható — problémáiról lesz szó. A szimplex módszer mellé állítva bemutatjuk a disztribuciós (egyszerűsített szimplex) módszert és általánosítjuk a „magyar módszer" néven ismertté vált graph-elméleti módszert.

* Beérkezett: 1958. december 1-én.

Külföldi lapok egyes cikkeiből és már néhány hazai kísérletből is azt láthatjuk, hogy a lineáris programozás a tudományos gazdasági üzemvezetés számára nagy segítséget jelent és ezért alkalmazása egyre jobban terjed. Ez év januárjában a szovjet tudósok Moszkvai Házának statisztikai szakosztálya megvitatta *Ja. P. Gercsuk*nak „A lineáris programozás mint a gazdasági feladatok optimális megoldásának kiválasztására szolgáló új módszer" címmel tartott előadását. A szovjet szakirodalomban ez a módszer eddig sem volt ismeretlen, mert *L. V. Kantorovics* leningrádi professzor már 1939-ben „A termelés megszervezésének és tervezésének matematikai módszerei" címen kiadott könyvében „a megoldást hozó szorzók módszere" elnevezéssel ismertette azt. A januárban lezajlott vita résztvevői rámutattak arra, hogy ez a fiatal módszer a szocialista gazdaságban igen széles körben alkalmazható és véleményük szerint ennek érdekében meg kell tenni a szükséges intézkedéseket.

Azt tapasztaljuk, hogy nincs ma olyan matematikai kongresszus, melynek anyagában ne szerepelne a lineáris programozás. Ez évben például a belga Mons-ban rendezett nemzetközi, de a balatoni matematikai kongresszuson is több előadásban foglalkoztak a lineáris programozás matematikai módszereivel, alkalmazásával és elektronikus számológépre való programozásával is.

Az egyre nagyobb tért hódító operáció-kutatásnak — ami alatt a gazdasági üzemvezetésben az üzemelés közben fellépő problémáknak tudományos módszerekkel és fejlett eszközökkel történő feltárását és megoldását értik —, a lineáris programozás igen jól használható módszere.

Több országban működnek már olyan elektronikus központok, melyek különböző üzemekkel, ipari és közlekedési vállalatokkal kábeles összeköttetésben vannak és bizonyos, a gép számára előre programozott, gyakran ismétlődő feladattípusok megoldásának eredményét ilyen módon ezek a vállalatok igen gyorsan megkaphatják. Ezeknek a feladatoknak jelentős része ugyancsak lineáris programozás.

Nemrégen számolt be egy olasz lap arról, hogy a francia vasútnál a Vezérigazgatóságon belül egy „operáció-kutató" csoportot szerveztek, mely rövidesen elektronikus számológéppel is rendelkezni fog. Tudjuk, hogy a Brit Vasutak Operáció Kutatási Osztálya már pár éve működik.

A szállítás területén adódó feladatokat vizsgálva azt látjuk, hogy vannak olyanok, amelyeket napjában (esetleg napjában többször) rendszeresen meg kell oldani. Ilyen lehet a vasútnál a különböző típusú üres kocsik rendszeres szétosztása, vagy más fuvarozó vállalat járműveinek irányítása. De vannak olyan feladatok is, amelyek ritkább időközönként jelentkeznek, mint pl. a tűzifaelosztás, kavicselosztás, cukorrépaelosztás stb. Ezeknek a feladatoknak nagy része (kis országokban különösen) manuálisan is gyorsan elvégezhető.

Az eddigi tapasztalatok szerint az alkalmazás szempontjából a lineáris programozás leggazdagabb területe a közlekedés. Itt ugyanis igen sokféle — és gyakran jelentkező — olyan probléma van, amelynek optimális megoldásához célszerűen a lineáris programozásnak, mint matematikai modellnek segítségével juthatunk. Ezeknek a problémáknak a zöme három nagyobb feladattípusba sorolható.

1. Az egyidejűleg vagy időben eltolva, gyakran naponként, esetleg naponként többször jelentkező fuvarozási szükségleteknek megfelelően a *szállítóeszközök* (tehergépkocsik, különböző típusú vagonok stb.) oly módon való *szétosztása*, hogy ezek a leghatékonyabb mozgatással (pl. minimális

üzemanyag-ráfordítással vagy minimális idő alatt) kerüljenek rendeltetési helyükre.

2. Különböző szállítási feladatokhoz a rendelkezésre álló — esetleg különböző fajtájú — *szállítóeszközök* oly módon való *beosztása,* hogy azok együttes kihasználása optimális legyen (pl. minimális ráfordítással a maximális teljesítményt nyújtsák). Ide tartozik általában a munkaeszközök vagy a dolgozók különböző munkafeladatok elvégzésére történő beosztása.

3. Újonnan létesítendő elosztó- vagy begyűjtőhelyeknek, iparvállalatoknak, mezőgazdasági üzemeknek, gépállomásoknak, *szállítóeszközök telephelyeinek,* garázsoknak stb. oly módon való elhelyezése, *telepítése,* hogy a létesítmény üzemével kapcsolatos össz-szállítás, illetve a szállítóeszközök kihasználása optimális lehessen (pl. minimális üresjárat). Ide tartozik a közlekedési vonalak útirányváltozatai közötti döntés feladata is, amikor azoknak — a közlekedési hálózatban — a fuvarozási önköltség alakulására gyakorolt hatásait vesszük figyelembe. A telepítési feladatokkal itt nem foglalkozunk, mivel ezek másfajta matematikai módszerek alkalmazását is szükségessé teszik. Például az optimális rayonok és centrumok meghatározása feltételes szélső érték fokozatos közelítéssel való megoldására vezet.

Bár nem közvetlenül a közlekedési vállalatok, üzemek problémája, mégis meg kell említeni, hogy a szállítási feladatok egyik nagy típusa : építőanyagok, üzemanyagok, nyersanyagok, mezőgazdasági termékek stb. oly módon való *elosztása,* hogy azt optimális szállítással (minimális szállítási teljesítménnyel, üzemanyag-ráfordítással, reálárakon számított minimális szállítási összköltséggel vagy minimális üresjárattal stb.) lehessen lebonyolítani. Ez a feladattípus a matematikai megfogalmazás és megoldás szempontjából az 1. feladattípussal rokon.

A lineáris programozás az erősen fejlődő tudományos gazdasági üzemvezetés sajátos módszere, melynek segítségével bizonyos gazdasági adottságok (így a szállítóeszközök) legcélszerűbb felhasználása meghatározható. Jelen esetben feladata, hogy az igen nagy számú lehetséges elosztási, szétosztási, beosztási és telepítési program (terv) közül kiválassza a legkedvezőbb, optimális változatot (vagy változatokat). Alkalmazásának két alapvető feltétele van. Először az elosztásra kerülő, vagyis a feladatban szereplő mennyiségek és a szállítás vizsgált és összegében minimummá teendő jellemzőjének mennyiségei között lineáris összefüggéseknek kell fennállni. Például egy bizonyos szállítási útvonalon a szállítási teljesítmény arányos a szállított tömeggel. Másodszor a programozásra kerülő mennyiségeknek bizonyos egységekre, pl. 1 m^3-re, 1 vagonra, 1 tonnakilométerre (tkm) stb. oszthatóknak kell lenniük.

Amennyiben ezen feltételeknek eleget téve meg tudjuk szerkeszteni a felvetett probléma modelljét, úgy arra a lineáris programozás alkalmazható és azzal valóban a kívánt optimális megoldást kapjuk.

Bizonyos elosztási vagy szétosztási probléma modellje tartalmazza a kiindulási és rendeltetési helyeket, a kiindulási helyeken rendelkezésre álló és a rendeltetési helyeken igényelt mennyiségeket és minden kiindulási helyről minden rendeltetési helyre történő fuvarozáshoz tartozó szállítási koefficienseket, vagyis az egyes szállítási útvonalakhoz tartozó fajlagos ráfordításokat. Ez lehet távolság, költség, üzemanyagfogyasztás stb., aszerint, hogy a szállítás mely jellemzőjét : a szállítási teljesítményt, a szállítás összköltségét vagy üzemanyagfelhasználását stb. akarjuk minimummá tenni. (A szállítási koefficiensek, másképpen nevezve paraméterek viszonylagos állandósága a linearitási követelménnyel kapcsolatos.) Amennyiben a szállítási koefficiensek több részből tevődnek össze, ezek közül azok, melyek a feladat határain belül konstans értékkel szerepelnek, vagyis melyek minden lehetséges szállítási útvonalat egyformán terhelnek, elhagyhatók. Például operatív viszonyok között, amikor a szállítóeszközök adottak, a minimális szállítási összköltséggel megvalósítható megoldást keresve a szállítás állandó költségét figyelmen kívül hagyhatjuk.

Egy beosztási probléma modellje tartalmazza a rendelkezésre álló egyes szállítóeszközöket, az elvégzendő egyes szállítási feladatokat, és minden lehetséges beosztáshoz ad egy — a vizsgálat szempontjának megfelelő — hatékonysági mutatót. A vizsgálat szempontja különböző (pl. a szállítóeszközök kihasználása, a megtett összes út, az üzemelési idő vagy költség, az elérhető gazdasági eredmény stb.) lehet. A lineáris programozás alkalmazásával olyan beosztási programot nyerünk, mely a kívánt optimális össz-hatékonyságot nyújtja.

A modellek megszerkesztése új feladatoknál nagy körültekintést, a probléma alapos feltárását kívánja és ilyenkor a szükséges adatok összegyűjtése, értékelése, statisztikai verifikációja rendszerint hosszabb időt vesz igénybe.

Természetesen egyidejűleg több szempontból is vizsgálhatjuk a felvetett problémát, akkor ennek megfelelően több optimális programot kapunk és ezek közül további mérlegeléssel választhatjuk ki a megvalósítandó programot.

Gyakran egy kívánt optimális megoldás sok egyéb kedvező hatást is biztosít számunkra. Pl. vasúti üzemnél a szállítási önköltség csökkentésével csökken a kocsiforduló, csökkennek a tüzelőanyag ráfordítások, kisebb lesz a járműszükséglet, csökkennek a mellékteljesítmények stb., ami a szállítási önköltség még nagyobb csökkenéséhez és beruházások megtakarításához vezet. Ezek az előnyök hatékonysági elemzéssel vizsgálhatók.

A lineáris programozás módszereinek nagy előnye, hogy a terjedelmes, sok ismeretlenes problémákat igen egyszerű matematikai apparátussal mind kézi számításra, mind elektronikus számológéppel való feldolgozásra különösen alkalmassá teszik, ezen problémák áttekinthető és könnyen mechanizálható megoldását biztosítják. A lineáris programozás segítségével igen gyorsan juthatunk régebben megoldhatatlannak látszó gazdasági feladatok pontos eredményéhez.

Matematikailag a szállítási feladat általában a következő összefüggésekkel írható le.

$$\sum_{j=1}^{n} x_{ij} = \alpha_i, \text{ ahol } \alpha_i > 0 \quad \text{és} \quad i = 1, 2, \ldots, m \tag{1}$$

$$\sum_{i=1}^{m} x_{ij} = \beta_j, \text{ ahol } \beta_j > 0 \quad \text{és} \quad j = 1, 2, \ldots, n \tag{2}$$

$$x_{ij} \geqslant 0 \tag{3}$$

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}\, x_{ij} = \min \tag{4}$$

ahol „α_i" jelenti a rendelkezésre álló mennyiséget „i" helyen, „β_j" jelenti a szükségelt mennyiséget „j" helyen és „c_{ij}" az „i" helyről „j" helyre történő legkedvezőbb szállítás fajlagos jellemzője, vagyis a szállítási koefficiens.

Tehát az (1) és (2)-vel megadott — „$m + n$" egyenletet és „$m \cdot n$" ismeretlent tartalmazó — egyenletrendszernek a nem-negatív változókra vonatkozó oly megoldását keressük, mely mellett a (4) lineáris függvény (az ún. gazdaságossági függvény) értéke minimumot vesz fel. Közvetlenül belátható, hogy ilyen megoldás — legalább egy — mindig létezik.

Az egyenletrendszer tömör kifejezését az

$$\mathbf{A}\,\mathbf{x} = \mathbf{b} \tag{5}$$

matrixegyenlet adja. A lineáris programozás különlegesen egyszerű módszerei, melyekkel a szállítási feladatok megoldhatók, az **A** matrixnak állandó jellegű és egyszerű felépítésű struktúrájából folyóan jöhetnek létre.

Az **A** matrix szerkezetét az [1] és [2] jelű irodalomban ismertettük.

A szállítási feladatok modellje megkívánja, hogy a

$$\sum_{i=1}^{m} \alpha_i = \sum_{j=1}^{n} \beta_j , \tag{6}$$

egyenlőség fennálljon. Amennyiben az adott probléma ezt nem elégíti ki, vagyis, ha az összes rendelkezésre álló és szükségelt mennyiség nem egyenlő egymással, akkor egy fiktív — pl. Y jelű — helyhez (feladathoz, szállítóeszközhöz) kell rendelni a különbséget, vagyis a fennmaradó (α_y, ill. β_y) mennyiséget. Ezzel a fenti követelménynek eleget teszünk. A fiktív helyhez tartozó c_{iy} vagy c_{yj} fajlagos jellemzők értékeit pedig zérónak vesszük fel.

A szállítási feladatok lineáris programozása önmagában biztosítja, hogy ily módon az ésszerűtlen szállításokat kiküszöböljük, mert e módszer matematikai feltételeinek kielégítése kizárja a keresztszállításokat, az ismétlődő szállításokat és a túl hosszú szállításokat is.

A szállítási feladatoknak a tényleges körülményekből fakadó különböző korlátozásait viszonylag egyszerű számításra vonhatjuk, ha a korlátozásoknak megfelelően módosítjuk az ideális viszonyok feltételezésével készített első modellünket. Korlátozott lehet pl. bizonyos szállítási útvonalak kapacitása, előfordulhat, hogy bizonyos szállítóeszköz bizonyos feladatra nem osztható be, lehetséges, hogy a rendelkezésre álló és a szükségelt mennyiségek nincsenek egyensúlyban stb. Tehát lehetséges és fontos is a modellt úgy megszerkeszteni és módosítani, hogy az a valóságos viszonyokat minél jobban fedje.

Az eddigiekből már látható, hogy egy szállítási feladat egzakt matematikai megoldása megkívánja a probléma jelentkezése után annak általános megfogalmazását, összetevőinek, céljainak, a célok elérésére szolgáló alternatíváknak feltárását. Mérlegelni kell a feladat jellemzőinek jelentőségét. Meg kell állapítani az egymással nem lineáris összefüggésben levő, esetleg ellentétesen működő, vagy egymástól független és el nem hanyagolható jellemzők népgazdasági szintű hatékonyságának mértékét. Megvizsgálva az egyes jellemzők minimális vagy maximális értékét biztosító eljárásokat (megoldásokat) a többi jellemző függvényében is, megkaphatjuk az egyes célkitűzéseknek megfelelő eljárások kombinált hatékonyságát és az ily módon súlyozott eljárások közül, figyelembe véve azok eredményességét és a döntés célját, kell az optimális megoldást kiválasztanunk. Természetes, hogy ehhez könnyen kezelhető matematikai módszer kidolgozása, esetleg analóg számológép megtervezése és elkészítése is szükséges. Igen nagy méretű vagy rendszeresen ismétlődő és percek alatt megoldandó kisebb feladatoknál pedig a kézi számolás már nem megfelelő, ezeknek megoldásához elektronikus számológép szükséges. A gépi számítás megkívánja a gép nyelvének ismeretét, s a feladatnak a gépre való programozását, vagyis a gép számára érthető lépésekre bontását.

A továbbiakban pusztán a matematikai módszerekkel foglalkozunk.

*

Maradjunk az [1] jelű irodalomban kidolgozott számpéldánál, adjunk azonban ennek teljesen általános tartalmat. Állapodjunk meg a következő elnevezésekben.

Az (1) és (2) egyenleteket a *kombinációs matrix*ban foglaljuk össze. A sorok és oszlopok indexei kiindulási és rendeltetési *címeket* jelentenek. A cím jelenthet helyet, de jelenthet szállítóeszközt, útvonalat és egyebet is. A kombinációs matrix négyszögei, cellái, vagyis a matrix komponensei az „$\mathbf{a}_{ij}$" *eljárás-vektorok,* melyek bizonyos feladatoknál útvonalat, míg más feladatoknál beosztást vagy egyebet reprezentálnak. Természetesen ez a matrix transzponálható, vagyis sorai és oszlopai egymással felcserélhetők, mert közömbös, hogy az (1) egyenleteket a sorokban és a (2) egyenleteket az oszlopokban vagy pedig fordítva írjuk le. Erre a sorrendre azonban az alább következő matrixok értelmezésénél figyelemmel kell lenni.

Mint az [1] és [2]-ben ismertettük, az $\mathbf{a}_{ij}$-vektorok képezik az **A** *együttható matrix* oszlopvektorait. A kombinációs matrix külső sora és oszlopa a címekhez szükségelt és rendelkezésre álló mennyiségeket, vagyis az (1), (2) egyenletek α_i és β_j adott számértékeit tartalmazza, nevezzük ezeket a sorok és oszlopok *multiplicitásának.* Ezek az adott számértékek képezik az (5) matrixegyenlet jobb oldalán levő **b** vektor elemeit. A kombinációs matrixba egy lehetséges megoldást, vagyis kombinációt írunk be, oly módon, hogy bizonyos eljárásvektorokhoz az (1), (2) egyenletrendszert kielégítő „x_{ij}" *tervszámokat* rendelünk. Ezek a tervszámok a (5) matrixegyenlet lehetséges megoldását jelentő **x** vektor komponensei. A minimummá teendő (4) függvény c_{ij} szorzói a feladatnak az egyes eljárás-vektorokra vonatkozó *fajlagos jellemzői,* melyek tartalmáról a bevezetésben már megemlékeztünk. Ezeket a c_{ij} értékeket a C *paramétermatrixban* foglaljuk össze. A megoldás szempontjából közömbös, hogy a fajlagos jellemzőket milyen egységben adjuk meg. $\bar{c}_{ij}$-al jelöljük a *közvetett fajlagos jellemzőket.* Ezeknek a fiktív értékeknek, mint látni fogjuk, az optimumkeresésnél van döntő szerepük. A „$c_{ij}-\bar{c}_{ij}$" különbséget, amellyel a kombinációs matrixban levő megoldást értékeljük, *javítófaktornak* nevezzük. A bennünket érdeklő nem-pozitív (illetve, ha negatív előjellel vesszük a c_{ij} paramétereket, akkor nem-negatív) értékű javítófaktorok megkeresése a számítás legfárasztóbb, de egyben legfontosabb része.

Számpéldánkat az 1. és 2. táblázat írja le.

Az [1]-ben ismertetett szimplex módszer matematikai lényege, hogy az **A** együttható-matrix által reprezentált „$m + n - 1$" dimenziós térben „$m + n - 1$" lineárisan független vektor választásával megalkotott bázis vektorait rendre kicseréljük a tér alkalmasan választott egyéb vektoraival és ily módon a kiindulásul választott bázist addig változtatjuk, míg a bázishoz tartozó lehetséges megoldás még javítható, vagyis amíg a lehetséges megoldások halmazában az optimális megoldást meg nem találtuk, mely bizonyíthatóan [3] ilyen *bázishoz tartozó megoldás,* és ezért az iteráció véges számú

1. táblázat

Kombinációs matrix

Rendeltetés / Kiindulás		Cím 1	2	3	4	n = 5	a_i
Cím	1	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	200
	2	x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	80
	3	x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	130
	m = 4	x_{41}	x_{42}	x_{43}	x_{44}	x_{45}	90
β_j		30	210	60	80	120	500

2. táblázat

C paramétermatrix

Rendeltetés / Kiindulás		Cím 1	2	3	4	n = 5
Cím	1	6	3	5	1	7
	2	3	7	4	4	1
	3	5	2	3	1	6
	m = 4	3	5	2	3	2

lépéssel elvégezhető. Mint ismeretes, az **A** matrixnak „$m + n$" sora és „$m \cdot n$" oszlopa van. Mivel a (6) egyenlőség szerint

$$\sum_{i=1}^{m} \alpha_i = \sum_{j=1}^{n} \beta_j,$$

ami azt jelenti, hogy az (1) és (2) egyenletrendszerben csak „$m + n - 1$" egyenlet független egymástól, és mert az együttható-matrix sorai az egyenletek baloldali együtthatóit tartalmazzák, könnyen belátható, hogy az „$m + n$" sorvektor között csak „$m + n - 1$" lineárisan független vektor van, akkor pedig ugyanannyi a lineárisan független oszlopvektorok száma is. Így tehát a különböző kombinációkban választott „$m + n - 1$" lineárisan független oszlopvektor lineáris kombinációjaként az összes többi oszlopvektor előállítható. Innen kapta a módszer az elnevezését is, ugyanis a bázissal kifejezhető vektorok halmazát *szimplex*nek nevezzük [4].

Az „$m \cdot n$" oszlopvektorból ugyan véges, de igen nagy számú bázist választhatunk. Ezeknek végigvizsgálására azonban nincs szükség, mert a szimplex táblázatok lehetőséget adnak arra, hogy minden helyesen elvégzett javítással egy olyan $\mathbf{a}_{ij}$ vektort vigyünk be a bázisba, mely az optimális megoldás bázisában is szerepelni fog, vagyis a legrosszabb esetben is

„$m + n - 1$" lépéssel

célhoz érünk. Erre azért van lehetőség, mert a bázisba mindig csak olyan új vektort vonunk be, amelynek javítófaktora bizonyos feltételnek eleget tesz.

A szimplex-táblázat megadja valamennyi $\mathbf{a}_{ij}$ eljárásvektornak kifejezését az adott bázisban. Ebből a kifejezésből pedig, az abban szereplő bázisvektorok együtthatóival megszorozva az illető bázisvektorokhoz tartozó c_{kl} fajlagos jellemzők értékét, a vizsgált $\mathbf{a}_{ij}$ vektornak a bázisra vonatkoztatott közvetett jellemzőjét, a $\bar{c}_{ij}$ értékét kapjuk.

Az adott bázishoz tartozó $\mathbf{x}$ megoldás (az ún. bázismegoldás) elbírálása úgy történik, hogy összehasonlítjuk valamennyi $\mathbf{a}_{ij}$ eljárásvektorhoz tartozó $\bar{c}_{ij}$ és c_{ij} értékeket és ha valamely eljárásvektornál a közvetett fajlagos jellemző nagyobb, mint a fajlagos jellemző, vagyis, ha

$$c_{ij} < \bar{c}_{ij},$$

úgy azt az eljárásvektort be kell vonni a bázisba, mert ily módon a (4) függvény értéke csökkenni fog. A bázismegoldás egyben lehetséges megoldás is, ha eleget tesz az (1), (2) és (3) összefüggéseknek. Ha a bázismegoldásban valamely eljárásvektorhoz negatív x_{ij} mennyiség tartozik, úgy azt a vektort egy megfelelő másik vektorral a bázisban ki kell cserélni.

A bázisban szereplő vektorok megfelelő lineáris kombinációit, mint tudjuk, nagyon könnyen előállíthatjuk, mert szállítási feladatoknál az $\mathbf{A}$ együttható matrix mindig azonos szerkezetű és egyszerűen felírható, így pedig a matrix $\mathbf{a}_{ij}$ oszlopvektorai, az eljárásvektorok is közvetlenül felírhatók, ugyanis az $\mathbf{a}_{ij}$ vektornak az „i"-ik és „$m + j$"-ik komponense „+ 1" a többi „0". Eszerint feladatunk együtthatómatrixa

$$\underset{(m+n,\ m\cdot n)}{\mathbf{A}} = \begin{bmatrix} 1\,1\,1\,1\,1 & 0\,0\,0\,0\,0 & 0\,0\,0\,0\,0 & 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 & 1\,1\,1\,1\,1 & 0\,0\,0\,0\,0 & 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 & 0\,0\,0\,0\,0 & 1\,1\,1\,1\,1 & 0\,0\,0\,0\,0 \\ 0\,0\,0\,0\,0 & 0\,0\,0\,0\,0 & 0\,0\,0\,0\,0 & 1\,1\,1\,1\,1 \\ & & & \\ 1\,0\,0\,0\,0 & 1\,0\,0\,0\,0 & 1\,0\,0\,0\,0 & 1\,0\,0\,0\,0 \\ 0\,1\,0\,0\,0 & 0\,1\,0\,0\,0 & 0\,1\,0\,0\,0 & 0\,1\,0\,0\,0 \\ 0\,0\,1\,0\,0 & 0\,0\,1\,0\,0 & 0\,0\,1\,0\,0 & 0\,0\,1\,0\,0 \\ 0\,0\,0\,1\,0 & 0\,0\,0\,1\,0 & 0\,0\,0\,1\,0 & 0\,0\,0\,1\,0 \\ 0\,0\,0\,0\,1 & 0\,0\,0\,0\,1 & 0\,0\,0\,0\,1 & 0\,0\,0\,0\,1 \end{bmatrix}$$

Ezért a lineáris kombinációban azok a bázisvektorok kapnak „+ 1" együtthatót, melyeknek első vagy második indexére a vizsgált vektor indexében közvetlenül vagy közvetve (amikor 3 bázisvektor lineáris kombinációjával nem lehet kifejezni a vektort) szükség van, az a bázisvektor kap „−1" együtthatót, melynek két indexe az előbbi bázisvektorok felesleges indexeit tartalmazza, a többi bázisvektor szorzója „0". Ebből az is következik, hogy a lineáris kombinációban nem-zérus együtthatóval páratlan számú bázisvektor szerepel, mert egy vektor kifejezéséhez három másik vektor kell, ezek közül két vektor egy-egy „+1" komponense megegyezik a vizsgált vektoréval (két „+1" komponens nem egyezhet meg, mert akkor ugyanaz a vektor lenne) egy vektor pedig a két felesleges „+1" komponenst tartalmazza. Amennyiben az egyik egyező „+1" komponensű vektor nem szerepel a bázisban, azt újabb három bázisvektor lineáris kombinációjával lehet csak előállítani, akkor már a vizsgált vektor lineáris kombinációjában öt nem-zérus együtthatójú bázisvektor szerepel stb. Ebből nyilvánvaló, hogy a lineáris kombinációban szereplő együtthatók összege mindig „+1".

Azokat az eljárásvektorokat, melyek a lineáris kombinációban „+1" együtthatóval szerepelnek, a vizsgált — tehát a bázisban nem szereplő — eljárásvektorhoz tartozó kicserélhető eljárásvektoroknak nevezzük, mert ezek közül akármelyiket kicserélhetjük a vizsgált vektorral és így újabb bázisokat kapunk, vagyis ha a vizsgált vektort a bázisba be akarjuk vonni, akkor a kicserélhető vektorok közül valamelyiket ki kell onnét emelni.

A továbbiakban a bázisban szereplő $\mathbf{a}_{ij}$ vektorokat és a hozzájuk tartozó c_{ij} és x_{ij} értékeket „kl" indexszel jelöljük.

A fenti gondolatmenetet még egyszerűbb ábrán megérteni. Ugyanis az optimális megoldást azon lehetséges megoldások között keressük, melyek egy nyitott poligonnal ábrázolhatók. A poligon oldalai a bázishoz tartozó, vagyis a megoldásban felhasznált eljárásvektorok. Közismert, hogy „$m + n$" pont (ez itt címet jelent) összekötéséhez minimálisan „$m + n - 1$" sokszögoldal szükséges és egyben elégséges is.

A [2]-ben a megoldáspoligonok ábrázolásával részletesen foglalkoztunk. Egyszerű sorrendi szabályok betartásával a vizsgált megoldás akkor is egyértelműen ábrázolható, ha a címek, vagyis a sokszögoldalak végpontjai nem helyeket jelölnek. Egy ilyen minimális oldalú megoldáspoligonon bármely kiindulási címről bármely rendeltetési címre csak egyféle módon juthatunk el, az így bejárt sokszögoldalak pedig azonosak a lineáris kombinációban szereplő (nem-zérus együtthatójú) eljárásvektorokkal. A poligonoldalak nyíliránya mindig a kiinduló címtől a rendeltetési cím felé mutat. Így egy — a megoldásban nem szereplő — eljárásvektort vizsgálva, azokat az oldalakat (bázisvektorokat) vesszük pozitív előjellel, melyeken a körüljárás során nyílirányban haladunk, míg ellenkező értelmű haladásnál az oldalt negatív előjellel vesszük. Ha ugyanilyen előjellel véve összegezzük a bejárt oldalakhoz (eljárásvektorhoz) tartozó c_{ij} fajlagos jellemzőket, a vizsgált eljárásvektor közvetett jellemzőjét kapjuk. Megfigyelhetjük majd, hogy az ilyen körüljárás során a nyílirány állandóan változik, amint a sarokpontok is felváltva hol kiindulási, hol rendeltetési címet jelentenek és természetes, hogy a körüljárás sokszögvonala három, vagy annál nagyobb páratlan számú — megoldásban szereplő — sokszögoldalt használ fel. A sokszögvonalnak azokat az oldalait, melyeken a nyílirány értelme megegyezik a vizsgált — megoldásban nem szereplő — oldal nyílirányának értelmével, az előzőknek megfelelően kicserélhető oldalaknak nevezzük, mert a vizsgált oldalt bármely kicserélhető oldal elhagyásával egyidejűleg bevonva, újból összefüggő, nyitott sokszögvonalat kapunk.

Így a szállítási feladatok (az ún. Hitchcock-probléma) lineáris programozása a szimplex módszer lényegesen egyszerűsített formájában is elvégezhető. Ugyanis az előbbi körüljárásokat nemcsak a megoldáspoligonon, hanem a paramétermatrixban is elvégezhetjük. Az ezen alapuló számítástechnika disztribuciós módszer néven terjedt el (az angol irodalom a „szállítási technika" elnevezést használja). Ez a módszer fölöslegessé teszi a bázisvektorok lineáris kombinációját, ahelyett mindjárt a közvetett jellemzőket állítjuk elő.

Egy jól kezelhető ilyen módszert már kidolgoztunk a [2]-ben. Kimutattuk, hogy egy bázishoz tartozó megoldás a paramétermatrixban úgy jelentkezik, hogy azt sakktáblának tekintve, a felhasznált „$m + n - 1$" számú négyszög (eljárásvektor) bástyamozgással végigjárható. Az ilyen megoldás felhasználhat olyan eljárásvektorokat is, amelyeken „0" értékű tervszám szerepel, ilyenkor degenerált programról beszélünk. A felhasznált négyszögek bástyamozgással való végigjárása azonos a megoldáspoligon végigjárásával.

Míg a megoldáspoligon sarokpontjai a címek, a bástyamozgás sarokpontjai a lehetséges megoldásban felhasznált eljárásvektorok. A vizsgált megoldásban fel nem használt eljárásvektorok közvetett fajlagos jellemzőjét úgy kapjuk meg, hogy a javítófaktorok matrixában a kérdéses négyszögről (eljárásvektor) elindulva olyan bástyamozgásokkal haladunk, melynek töréspontja csak a megoldásban szereplő eljárásvektor (négyszög) lehet és így vissza kell érkeznünk a kérdéses négyszögbe. Ez a körüljárás a javítófaktorok matrixában is csak egyféleképpen végezhető el. Az első sarokponthoz tartozó eljárásvektor fajlagos jellemzője pozitív, másodiké negatív, majd tovább is pozitív és negatív váltogatják egymást. Ily módon az összes közvetett fajlagos jellemző előállítható. A megoldásban szereplő eljárásvektorokra vonatkozólag

$$c_{kl} = \bar{c}_{kl} \text{ érvényes.}$$

A jobb áttekinthetőség okából a megoldások értékelésére olyan táblázatot szerkesztünk, melyen a kombinációs matrixot és a paramétermatrixot összevonjuk. Ebbe a táblázatba írjuk be a vizsgált megoldást és itt állítjuk elő a javítófaktorokat is.

A [2]-ban már bemutattuk, hogy a közvetett fajlagos jellemzők előállításához a fenti körüljárásokat nem szükséges elvégezni, mert azok matrixában bármely két sor azonos oszlopban levő tagjai és bármely két oszlop azonos sorban levő tagjai azonosan különböznek egymástól. Ezek a különbségek pedig a megoldásban szereplő eljárásvektorokhoz tartozó c_{kl} értékek segítségével előállítható.

Belátható, hogy az „$m + n - 1$" eljárásvektort felhasználó megoldások a fentiek szerint a közvetett jellemzők matrixát egyértelműen meghatározzák. Vagyis ha egy „m, n" típusú null-matrixot úgy transzformálunk (sorait és oszlopait úgy növeljük), hogy a megoldásban felhasznált négyszögekbe éppen a c_{kl} érték kerüljön, akkor előállítottuk a vizsgált megoldáshoz tartozó közvetett jellemzők matrixát. Ugyanis minden c_{kl} előállítható két tag összegeként

$$c_{kl} = u_i + v_j \,, \tag{7}$$

ahol u_i $(i = k)$ az „i"-ik sorhoz és v_j $(j = l)$ az „j"-ik oszlophoz rendelt *címszám,* melyek közül az egész matrixban csak egy vehető fel tetszőlegesen. Még jobban érzékelhető ez a megoldáspoligonon, amelynek oldalaihoz tartozó c_{kl} értékeket szétbontjuk az illető sokszögoldal kezdő és végpontjára, vagyis a címekre. Mivel a megoldáspoligon szintén csak egyféleképpen járható végig, ha egy ponthoz, vagyis címhez egy címszámot rendelünk (egy kiindulási címhez „u_i"-t vagy egy rendeltetési címhez „v_j"-t), ebből a megoldáspoligonon haladva az összes többi címszám kiadódik.

A címszámokkal pedig a megoldásra vonatkoztatott összes közvetett jellemzőt előállíthatjuk, ugyanis

$$c_{ij} = u_i + v_j \,. \tag{8}$$

A (8) összefüggést *Dantzig* már régebben közölte [5], mégis a közvetett jellemzők előállításának a fenti technikáját először *Henderson* és *Schlaifer,* majd mások csak évekkel később publikálták [6] anélkül, hogy annak magyarázatát adták volna. Pedig az együttható matrix és az annak oszlopvektorait képező eljárásvektorok általunk magyarázott struktúrájával ennek a módszernek helyessége is igazolható.

Legyen ugyanis

$$c_{kl} = \mathbf{z}^* \cdot \mathbf{a}_{kl}\,, \tag{9}$$

ahol $\mathbf{z}^*$ sorvektor komponensei a keresett u_i és v_j értékek, tehát

$$\mathbf{z}^* = [u_1, u_2, \ldots, u_m\,;\; v_1, v_2, \ldots, v_n]\,.$$

A (9)-ből a $\mathbf{z}^*$ sorvektor egy komponensének önkényes felvétele után a többi komponens egyértelműen meghatározható, ha az $\mathbf{a}_{kl}$ vektorokat és c_{kl} fajlagos jellemzőket olyan sorrendben tesszük be a (9) összefüggésbe, hogy minden következő tag „k" vagy „l" indexe azonos legyen az előző tag „k" vagy „l" indexével, ily módon egyszerre mindig csak egy „u_i" vagy „v_j" értéket kell az egyenletből megállapitani.

Az összes közvetett fajlagos jellemző a sorvektorral kifejezhető, mert

$$\bar{c}_{ij} = \mathbf{z}^* \cdot \mathbf{a}_{ij}\,. \tag{10}$$

Az $\mathbf{a}_{ij}$ vektorok előállíthatók az $\mathbf{a}_{kl}$ bázisvektorok lineáris kombinációjaként. A $\bar{c}_{ij}$ értékeket pedig úgy kapjuk, hogy a lineáris kombinációban szereplő $\mathbf{a}_{kl}$ vektorok együtthatóival ($+1$ és -1) vesszük a hozzájuk tartozó c_{kl} értékeket. Jelöljük ezeket az együtthatókat γ_φ-vel és az $\mathbf{a}_{ij}$ vektor kifejezését

$$\mathbf{a}_{ij} = \sum_\varphi \gamma_\varphi \cdot \mathbf{a}_{kl,\varphi} \text{ összeggel,}$$

és hasonlóan a közvetett fajlagos jellemzőt

$$c_{ij} = \sum_\varphi \gamma_\varphi c_{kl,\varphi} \text{ összeggel,}$$

ahol φ a körbejárással érintett sokszögoldalak, illetve a bástyamozgással érintett sarokpontok sorrendjét jelenti. Akkor (9) behelyettesítésével

$$\sum_\varphi \gamma_\varphi c_{kl,\varphi} = \mathbf{z}^* \sum_\varphi \gamma_\varphi \mathbf{a}_{kl,\varphi} = \mathbf{z}^* \mathbf{a}_{ij}\,,$$

és mivel az $\mathbf{a}_{ij}$ vektorok szerkezetéből következik, hogy

$$\mathbf{z}^* \cdot \mathbf{a}_{ij} = u_i + v_j$$

skalárösszeggel, ezt a (10)-be helyettesítve igazoltuk (8) összefüggést, vagyis

$$c_{ij} = u_i + v_j\,.$$

A közvetett jellemzők matrixa pedig — vagyis a már említett nullmatrix transzformációja matematikai szimbólumokkal — két diád összegeként írható le.

$$\bar{C} = \mathbf{u} \cdot \mathbf{e}^*_{(n)} + \mathbf{e}_{(m)} \cdot \mathbf{v}^* \tag{11}$$

ahol $\mathbf{u} = \begin{bmatrix} u_1 \\ u_2 \\ \cdot \\ \cdot \\ \cdot \\ u_m \end{bmatrix}$, $\mathbf{v}^* = [v_1, v_2, \ldots, v_n]$

$$\text{és}\quad \mathbf{e}^*_{(n)} = \sum_{j=1}^{n} \mathbf{e}^*_j, \qquad \mathbf{e}_{(m)} = \sum_{i=1}^{m} \mathbf{e}_i$$

olyan n-ed, illetve m-ed rendű vektor, melynek minden eleme „1".

Ezekután lássuk az 1. és 2. táblázatban megfogalmazott számpélda megoldását.

*

A 3. táblázat négyszögeinek bal felső sarkaiban a paramétermatrix elemeit, a matrixon kívüleső jobbszélső oszlopban és a legalsó sorban pedig az adott α_i és β_j számértékeket, a multiplicitásokat írjuk fel.

Ezután szerkesszük meg a kiinduló megoldásunkat, vagyis az első kombinációt. Bizonyos szabályok betartásával már a kiindulásnál igen kedvezően befolyásolhatjuk a számítás további menetét, jelentősen megrövidíthetjük az optimális megoldáshoz vezető utat. Jelöljük meg minden sor és oszlop legkisebb c_{ij} értékét és elsősorban az így (a 3. táblázatban *-gal) megjelölt négyszögeket (eljárásvektorokat) használjuk fel az első kombinációban. Kerüljük a nagy c_{ij} értékű négyszögeket. Ha egy sorban (oszlopban) több azonos nagyságú fajlagos jellemző között kell döntenünk, azt válasszuk, melynek oszlopában (sorában) a nagyobb c_{ij} értékek vannak. Keressük a kétszeresen kedvező négyszögeket (ezeknek c_{ij} értékei sorukban és oszlopukban is a legkisebb értéket képviselik), ezeket külön (3. táblázatban két *-gal) jelöljük meg és az első kombinációba feltétlenül vegyük be, illetve ilyen négyszögből induljunk is el. Ezen szabályok betartásával készítettük el a 3. táblázatban vastag számokkal feltüntetett — és összefüggő bástyamozgással bejárható — kiinduló megoldást.

3. táblázat

Első kombináció és a javítófaktorok matrixa

$(-u_i)$ \\ v_j	6	3	4	1	4	
0	6 **30**— 0	3 **90**+ 0	5 +	1 * **80** 0	7 +	**200**
3	3 * 0	7 +	4 +	4 +	1 * * **80** 0	**80**
1	5 0	2 * **120**— 0	3 **10**+ 0	1 * +	6 +	**130**
2	3 * ○ —1	5 +	2 * **50**— 0	3 +	2 * **40** 0	**90**
	30	**210**	**60**	**80**	**120**	**500**

Ezután a (7) összefüggésnek megfelelően számítsuk ki az „u_i" és „v_j" értékeket.

A javítófaktorok meghatározását kényelmesebbé tehetjük azáltal, ha a táblázat bal szélén a „$-u_i$" értékeket írjuk fel, mert így a sorban összeadást, az oszlopban pedig kivonást kell elvégeznünk. Ugyanis

$$c_{ij} - \bar{c}_{ij} = [(-u_i) + c_{ij}] - v_j, \tag{12}$$

tehát a sorban elvégezzük a zárójelben levő összeadást és az összegből az oszlopban kivonjuk a „v_j"-t. Az „u_i" és „v_j" értékek meghatározásánál ennek megfelelően úgy járunk el, hogy a megoldáshoz nem tartozó „c_{ij}" értékeket figyelmen kívül hagyva, megkeressük a legnagyobb „c_{kl}" értéket, és annak sorához rendeljük a „$(-u_i) = 0$" címszámot. Akkor „$v_j = c_{kl}$". A többi címszámot — a megoldásban felhasznált négyszögeket végigjárva — az alábbi összefüggésekből kapjuk

$$(-u_i) = v_j - c_{kl}$$

$$v_j = (-u_i) + c_{kl}, \text{ ahol } i = k \text{ és } j = l.$$

Ha további kényelmi szempontból azt kívánjuk, hogy az összes „$(-u_i)$" és „v_j" nem-negatív szám legyen, akkor a „$(-u_i)$" és „v_j" értékek között található legnagyobb negatív szám abszolút értékével növeljük meg az összes „$(-u_i)$" és „v_j" címszámot. Ezt megtehetjük, mert

$$u_i + v_j = -[(-u_i) + |\varkappa|] + (v_j + |\varkappa|).$$

Az első kombinációban a

$$\max c_{kl} = c_{11} = 6,$$

ehhez felvesszük $(-u_1) = 0$ komponenst, mivel $v_j = (-u_i) + c_{kl}$, az első sorból $v_j = (-u_1) + c_{1l}$ alapján (ahol $j = l$) $v_1 = 6$, $v_2 = 3$ és $v_4 = 1$; a 2. oszlopban továbbhaladva, mivel $(-u_i) = v_j - c_{kl}$,

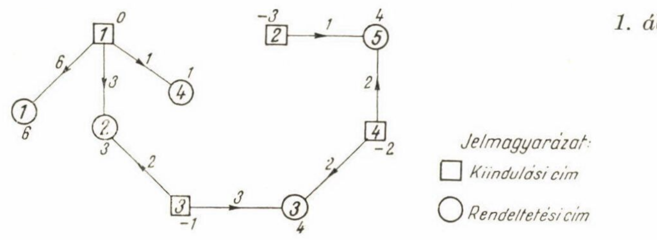
$(-u_3) = 3 - 2 = 1$ értéket kapjuk, míg a 3. sorból
$v_3 = 1 + 3 = 4$, a 3. oszlopból
$(-u_4) = 4 - 2 = 2$, a 4. sorból
$v_5 = 2 + 2 = 4$ és az 5. oszlopból
$(-u_2) = 4 - 1 = 3.$

Így meghatároztuk az összes „$-u_i$" és „v_j" értéket és ezzel a közvetett fajlagos jellemző matrixát is. A $\bar{c}_{ij}$ értékeket azonban szükségtelen előállítani, mert a javítófaktorokat, vagyis a „$c_{ij} - \bar{c}_{ij}$" különbségeket a (12)-ből közvetlenül megkapjuk. Ha

$[(-u_i) + c_{ij}] > v_j$, akkor a négyszög bal alsó sarkába „+" jelet teszünk, ha

$[(-u_i) + c_{ij}] = v_j$, akkor „0"-át írunk és ha
$[(-u_i) + c_{ij}] < v_j$, akkor a különbség értékét feltüntetjük.

Felrajzolunk az első kombináció ábrázolására egy hozzátartozó megoldáspoligont, hogy az olvasó korábbi megállapításainkat ellenőrizni tudja.



1. ábra.

A sokszögoldalak (eljárásvektorok) mellett feltüntettük a megfelelő c_{kl} fajlagos jellemzőket, a címek mellett pedig az u_i, és v_j értékeket.

A 3. táblázaton három üres négyszögben találunk nem-pozitív értékű javítófaktort. Ezen négyszögek, illetve az általuk reprezentált eljárásvektorok közül azt vonjuk be a következő megoldásba, mellyel a vizsgált jellemző, tehát a (4) függvény első kombinációhoz tartozó értékét jobban csökkentenénk.

A vizsgált jellemző az első kombinációban

$$\sum_{i=1}^{m} \sum_{j=1}^{n} c_{ij}\, x_{ij} = 6 \cdot 30 + 3 \cdot 90 + 1 \cdot 80 + 1 \cdot 80 + 2 \cdot 120 + \\ + 3 \cdot 10 + 2 \cdot 50 + 2 \cdot 40 = 1060$$

és ez az összeg a következő kombinációban a bevonandó eljárásvektor javító faktorának és az illető eljárásvektorhoz rendelendő x_{ij} tervszámnak szorzatértékével fog megváltozni. Ezért az új kombinációba bevonandó négyszögtől (eljárásvektortól) azt kívánjuk, hogy

$$|(c_{ij} - \bar{c}_{ij})\, x_{ij}| = \text{maximum legyen,}$$

ahol $(c_{ij} - \bar{c}_{ij}) \leqslant 0$. Ily módon gyorsabban közelítjük meg az optimumot. Ezért a tekintetbe jöhető (nagyobb negatív értékű) javítófaktorok eljárásvektoraihoz meg kell keresni a hozzárendelhető x_{ij} tervszámot. Mivel az új eljárásvektor bevonása esetén a vizsgált megoldásból az egyik kicserélhető eljárásvektort ki kell emelni, először is meg kell keresni a megfelelő *kicserélhető négyszögeket*. Tehát a következő kombináció számára tekintetbe jöhető négyszögekhez el kell végezni a bástyamozgásszerű körüljárást, mert a körüljárás páratlan sorszámú sarokpontjai — vagyis, amelyek c_{kl} értékei a közvetett fajlagos jellemző előállításában pozitív előjelet kapnának — lesznek az illető négyszög kicserélhető négyszögei. Válasszuk ki ezek közül azt a négyszöget, melyben a legkisebb c_{kl} tervszám szerepel és ez lesz a vizsgált négyszöghöz tartozó *kicserélendő négyszög*. Ugyanis a kicserélendő négyszög x_{kl} tervszámát az összes kicserélhető négyszög c_{kl} tervszámából le kell vonni és a körüljárás többi sarokpontján levő x_{kl} értékhez, beleértve a felhasználandó négyszöget is, azt hozzá kell adni.

Ily módon kerülhetjük csak el, hogy az új megoldásban sem forduljanak elő negatív x_{kl} mennyiségek, vagyis a (3) feltétel kielégítését így biztosítjuk. Az pedig, hogy a kicserélendő négyszög (eljárásvektor) x_{kl} mennyiségét átvive az újonnan felhasználandó négyszögbe, a páratlan sorszámú sarokpontokon le kell vonni, a párosakon pedig hozzá kell adni, az (1) és (2) feltételből következik. Ugyanis, ha a kombinációs matrix sorában vagy oszlopában egy értéket megnövelünk, akkor ugyanabban a sorban vagy oszlopban egy másik értéket csökkenteni kell és ugyanígy fordítva.

Ily módon a számbajöhető négyszögek kicserélendő négyszögeit megtalálhatjuk és kiszámíthatjuk a hozzájuk tartozó

$$(c_{ij} - \bar{c}_{ij})\ x_{kl}$$

szorzatokat.

Ebből következik, hogy azoknak a négyszögeknek bevonásával, melyeknek javítófaktora zéró értékű (vagyis ahol $c_{ij} = \bar{c}_{ij}$), a (4) függvény értéke nem fog változni. Ezért példánkban csak a $\mathbf{a}_{41}$ eljárásvektor (négyszög) jön tekintetbe. Tegyünk a körüljárás páratlan sorszámú sarokpontjain szereplő x_{kl} értékek után egy „—" jelet, a páros sarokpontokon „+" jelet és az $\mathbf{a}_{41}$ négyszögébe rajzoljunk egy kört. A „—" jelű négyszögek közül ez lesz a kicserélendő négyszög, melyben a legkisebb x_{kl} érték szerepel. Ez jelenleg az $\mathbf{a}_{11}$, ahol $x_{11} = 30$. Akkor a (4) függvény értéke a javítás után

$$1060 - 1 \cdot 30 = 1030 \text{ lesz.}$$

Ezután könnyű lesz a második kombinációt megszerkeszteni, amit a 4. táblázatban tüntettünk fel. A javítófaktorok előállítását az előbbiek szerint itt is elvégezzük.

4. táblázat

Második kombináció és a javítófaktorok matrixa

$(-u_i)$ \ v_j	5	3	4	1	4	
0	6 +	3 **120** 0	5 +	1 **80** 0	7 +	**200**
3	3 +	7 +	4 +	4 +	1 **80** 0	**80**
1	5 +	2 **90** 0	3 **40** 0	1 +	6 +	**130**
2	3 **30** 0	5 +	2 **20** 0	3 +	2 **40** 0	**90**
	30	**210**	**60**	**80**	**120**	**500**

Ebben a matrixban már nincs egyetlen negatív értékű javítófaktor sem, vagyis a második megoldás tovább már nem javítható, tehát *optimális megoldás,* mégpedig mivel a javítófaktor egyetlen fel nem használt négyszögben sem zéró, ez az egyetlen optimális megoldás. Ha a javítófaktorok matrixában

„$m+n-1$”-nél több zéró elem van, az annyit jelent, hogy több azonos függvény-értékű megoldás létezik, de ha most is csak azokat tekintjük optimálisnak, melyek „$m+n-1$” négyszöget használnak fel, úgy ilyen megoldás általában, amint azt az [1]-ben megmutattuk, kevés van, és ezek könnyen megszerkeszthetők. Ilyenkor az egyenértékű optimális megoldások közül más szempontok figyelembevételével választhatjuk ki a legkedvezőbb megoldást.

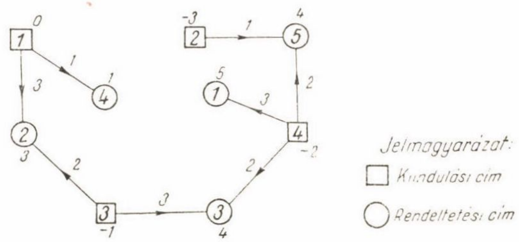
Ellenőrizhető, hogy az optimális megoldáshoz tartozó függvényérték valóban

$$\sum_{i=1}^{m}\sum_{j=1}^{n} c_{ij}\, x_{ij} = 1030.$$

Az olvasó a bemutatott példa megoldásán keresztül az (1)-ben a szimplex-módszert, most pedig a distribuciós módszert ismerhette meg.

Fejtegetéseink alátámasztására a 2. ábrán bemutatjuk a második kombinációhoz tartozó fiktív megoldáspoligont.

2. ábra



Érdemes a fajlagos jellemzők matrixát és az optimális megoldáshoz tartozó közvetett fajlagos jellemzők és javítófaktorok matrixát összehasonlítani, ezért az 5. táblázatban összefoglaltuk ezt a három matrixot. Minden négyszögben először a fajlagos jellemzők, majd a közvetett fajlagos jellemzők, és végül a javítófaktorok matrixának megfelelő elemeit tüntettük fel.

5. táblázat

A $[c_{ij}]$, $[\bar{c}_{ij}]$ és $[c_{ij}-\bar{c}_{ij}]$ matrixok

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

Az itteni $[c_{ij} - \bar{c}_{ij}]$ matrix elemei megegyeznek az [1]-ben a 3. f. táblázat utolsó sorának elemeivel, de az ottani értelmezés miatt ott ezek az elemek mind negatív előjellel szerepelnek.

Természetes, hogy az optimális megoldáshoz tartozó közvetett jellemzők matrixa csak olyan u_i és v_j komponensekkel transzformálható, melyek eleget tesznek az alábbi feltételeknek.

$$u_i + v_j \leqq c_{ij}\,, \tag{13}$$

ahol $i = 1, 2, \ldots, m$ és $j = 1, 2, \ldots, n$.

mert ellenkező esetben a megoldás még javítható lenne.

Vagyis, ha egy „m, n" típusú null-matrixot fokozatosan úgy transzformálunk, hogy elemei közül legalább „$m + n - 1$" egyenlő legyen a fajlagos jellemzők matrixának megfelelő elemeivel, ha tehát legalább „$m + n - 1$" elem eleget tesz a

$$c_{ij} = \bar{c}_{ij} \tag{14}$$

egyenlőségnek és a többi elem is kielégíti a (13) feltételt és ha a (14) feltételnek eleget tevő eljárásvektorok felhasználásával az (1), (2) egyenletrendszer kielégíthető, akkor ily módon, más sorrendben ugyan, de szintén az optimális megoldáshoz jutunk. Ha a transzformáció során a matrix valamely eleme eleget tesz a (14) egyenlőségnek, annak négyszögében egy — az (1), (2) feltételt kielégítő — x_{ij} tervszámot írunk és a transzformációt abban a sorban vagy oszlopban folytatjuk, ahol az (1) vagy (2) még nincs kielégítve. Ily módon az (1), (2) és (3) megszorításokkal haladhatunk az optimum felé.

Az itt tárgyalt elvek alapján a szállítási feladatokat elektronikus számológépekre programozhatjuk, ezzel azonban külön kívánunk foglalkozni.

*

Ezekután ismerkedjünk meg a Kőnig-féle graph-elméleti tétellel és az azon alapuló „magyar módszerrel". A szóban forgó tételt *Kőnig Dénes* graph-elméleti úton bizonyította be, majd *Egerváry Jenő* *Kőnig*-tételének új, matrix aritmetikai úton történő bizonyítását adta és általánosította is azt [7]. Azonban ez a tétel és bizonyításai hosszú ideig nem keltettek különösebb érdeklődést. A legutóbbi években *Kuhn* alkalmazta azt elsőnek a beosztási probléma elfajult esetére, amikor az (1) és (2) egyenletek a következő alakba mennek át

$$\sum_i x_{ij} = \sum_j x_{ij} = 1\,, \tag{15}$$

ahol $i, j = 1, 2, \ldots, n$ $(n \geqq 3)$ és

$$\text{az } x_{ij} = x_{ij}^2 \tag{16}$$

megszorítással előírják, hogy x_{ij} értéke csak „+1" vagy „0" lehet. Kuhn a Kőnig—Egerváry-féle tétel felhasználásával kidolgozta a fenti probléma megoldásának számítási algoritmusát [8]. Itt ezt a számítási algoritmust az összes szállítási probléma megoldására általánosítjuk. Kétségtelen azonban, hogy ennek a módszernek különös előnye a (15) és (16) feltételekkel korlátozható speciális beosztási problémáknál jelentkezik, mert ezek a problémák nem „$2n - 1$", hanem a (15) és (16)-ból következően csak „n" négyszöget használnak fel a kombinációs matrixban, ami itten természetesen kvadratikus alakú. Mi a magyar módszer számítási menetét és az 1. és 2. táblázattal megadott általános típusú és jelentésű számpéldán mutatjuk be.

Kőnig tétele számunkra így fogalmazható meg: bármely kvadratikus matrixra vonatkozóan, melynek elemei részben zérusok, részben zérustól különböző pozitív számok, a fedővonalak minimális száma egyenlő a független pontok maximális számával. A fedővonalak (sorok és oszlopok) rendszere valamennyi zéró elemet tartalmazza, míg független pontoknak nevezzük azokat a zéró elemeket, melyek közül nem esik kettő egy vonalba.

Az (1), (2) és (3) feltétellel körülhatárolt általános szállítási probléma mindig átalakítható a (15) és (16) feltételnek eleget tevő elfajult problémává. Ha például az 1. táblázatunkban megadott kombinációs matrixot és a 2. táblázatban megadott paramétermatrixot felbontjuk egy-egy olyan nagy kvadratikus matrixra, melyben azoknak minden kiindulási címe α_i-szer és minden rendeltetési címe β_j-szer fog szerepelni, akkor az elfajult problémához jutunk, melynek javítófaktor-matrixára a *Kőnig-tétel* alkalmazható. Hagyjuk meg azonban ezeket a matrixokat eredeti formájukban, csupán a paramétermatrix sorai mellé és oszlopai alá írjuk fel a megfelelő α_i és β_j multiplicitásokat, kifejezve ezzel, hogy az elfajult matrixban az illető sor vagy oszlop hányszor fordul elő. Akkor a *Kőnig-tétel* gyakorlatilag számunkra a következőt jelenti: („0" index-szel jelölve az adott fajlagos jellemzőket), a C_0 paramétermatrixot addig kell transzformálnunk, míg olyan C_p matrixhoz jutunk, melyben a fedővonalak minimális száma, vagyis a fedővonalak multiplicitásának minimális összege egyenlő lesz a sorok (illetve oszlopok) multiplicitásainak összegével, vagyis a terv összmennyiségével. Ez a C_p matrix „$m + n - 1$" vagy több zéró értékű elemet fog tartalmazni, és azonos az optimális megoldáshoz tartozó javítófaktorok matrixával. Ennek az eljárásnak alapját a $C_0 = [c_{ij}^{(0)}]$ paramétermatrixnak az a fontos tulajdonsága adja meg, hogy az abból képezett másik, így a $C_p = [c_{ij}^{(p)}]$ matrixnak megoldása is azonos a C_0 matrix megoldásával, ha

$$c_{ij}^{(p)} = c_{ij}^{(o)} - u_i - v_j\,, \qquad (17)$$

ahol u_i és v_j tetszés szerinti állandók, melyek eleget tesznek a következő — már ismert — összefüggésnek

$$c_{ij}^{(o)} \geqq u_i + v_j \qquad (18)$$

vagyis

$$c_{ij}^{(p)} \geqq 0\,.$$

Jelöljük a fedővonalak minimális rendszerét S-el és az ehhez tartozó és a multiplicitásukkal vett fedővonalak számát s-sel, akkor a C_p matrixra vonatkozóan

$s^{(p)} = M$, ahol M a programozásra kerülő, és a felvett egységekben megadott teljes mennyiséget jelenti. Mivel a kombinációs matrixban — általánosan kifejezve — az összes lehetőség egyenlő az összes szükséglettel

$$M = \sum_{i=1}^{m} \alpha_i = \sum_{j=1}^{n} \beta_j .$$

A C_0 matrix transzformációját a 6. táblázatban végezzük el, és a C_p matrixhoz elérkezve ezen a táblázaton fogjuk megszerkeszteni az optimális kombinációt is. Meg kell jegyezni, hogy a C_0 matrix redukálásával nyert

$C_1, C_2, \ldots, C_{p-1}$ matrixok nem az egyetlenek, melyeken keresztül a C_p-hez eljuthatunk, mert ezek a közbenső transzformációból származó matrixok attól függnek, hogy a fedővonal-rendszerek alternatívái közül melyiket választottuk. A 6. táblázat bal szélén felírt indexek $(0, 1, \ldots, p)$, jelzik, hogy e sorokban található elemek hányadik transzformált matrixhoz, vagyis hányadik lépéshez tartoznak. Az utolsó index a C_p matrix elemeit jelöli meg.

6. táblázat

A $C_{0,1,\ldots,p}$ matrix és az optimális kombináció

		4		3	4	4		2	3 4				
	0,	6		3	*	5		1	*	7			
	2, 1,	3		1		3		0		6			0
	3,	2		0	**120**	2		0	**80**	5		**200**	
	4,	1		0		1		0		4			
5	5,	1		0		1		0		3			
	0,	3		7		4		4		1	*		
2	2, 1,	0		5		2		3		0			0
3	3,	0		5		2		4		0	**80**	**80**	
4	4,	0		6		2		5		0			
5	5,	1		7		3		6		0			
	0,	5		2	*	3		1		6			
2	2, 1,	2		0		1		0		5			0
	3,	2		0	**90**	1	**40**	1		5		**130**	
	4,	1		0		0		1		4			
5	5,	1		0		0		1		3			
	0,	3	*	5		2	*	3		2	*		
2	2, 1,	0		3		0		2		1			0
3	3,	0	**30**	3		0	**20**	3		1	**40**	**90**	
	4,	0		4		0		4		1			
5	5,	0		4		0		4		0			
			30		**210**		**60**		**80**		**120**	**500**	
		3		2		2		1		1			

A C_0 matrix transzformálását a következő lépésekben hajtjuk végre:
1. lépés. Keressük meg a C_0 matrix minden oszlopának legkisebb elemét:

$$v_j^{(0)} = (\min c_{ij}^{(0)})$$

és azt írjuk az illető oszlop alá, majd vonjuk ki az oszlop összes eleméből. Így új, C_1 matrixot kapunk, melynek elemei

$$c_{ij}^{(1)} = c_{ij}^{(0)} - v_j^{(0)}, \text{ ahol}$$

$$i = 1, 2, \ldots, m \quad \text{és} \quad j = 1, 2, \ldots, n .$$

A $c_{ij}^{(1)}$ értékeket a $c_{ij}^{(0)}$ értékek alá írjuk és a táblázat bal szélén a $c_{ij}^{(1)}$ értékkel egyvonalban írjuk fel az „1" indexeket. Így táblázatunkban felírtuk a C_1 matrixot is. Ennek a matrixnak már minden oszlopában lesz legalább egy „0" eleme. Mint a 6. táblázat mutatja, példánkban

$$v_1^{(0)} = 3,\ v_2^{(0)} = 2,\ v_3^{(0)} = 2,\ v_4^{(0)} = 1 \text{ és } v_5^{(0)} = 1 \text{ volt.}$$

2. lépés. Megvizsgáljuk a C_1 matrix sorait és azoknak legkisebb elemét felírjuk a táblázat jobb szélén az illető sorok mellé. Az így kapott

$$u_i^{(1)} = (\min c_{ij}^{(1)})$$

értékeket kivonjuk az illető sorok minden eleméből, vagyis

$$c_{ij}^{(2)} = c_{ij}^{(1)} - u_i^{(1)} \text{ lesz.}$$

Az így kapott C_2 matrix, amelyet a C_1-hez hasonlóan szintén beírunk táblázatunkba, már minden oszlopában és minden sorában is tartalmaz legalább egy „0" elemet. Néha — mint most is — előfordul, hogy ezt már az 1. lépéssel elérjük. Példánkban ugyanis $u_i^{(1)} = 0$ $(i = 1, 2, \ldots, m)$. Ilyenkor a 2. lépésnek ez a része elmarad.

Ezután megkeressük azt a fedővonalrendszert, amely lefedi az összes „0" elemet és amelyre vonatkozóan a multiplicitásukkal vett fedővonalak száma a minimum. Példánkban az S_2 fedővonalrendszer a 2., 3. és 4. sort és a 4. oszlopot takarja. Az S_2 fedővonalainak száma

$$s^{(2)} = 80 + 130 + 90 + 80 = 380 .$$

A 6. táblázaton a S_2 fedővonal-rendszert a sorok mellé, ill. az oszlopok fölé húzott és 2-vel jelölt vonalakkal tüntettük fel.

Megjegyzendő, hogy amennyiben a minimális fedővonal-rendszer megválasztásában olyan hibát követünk el, hogy a hozzátartozó $s^{(k)}$ nem a tényleges minimum, mert létezik

$$s^{(k')} < s^{(k)},$$

ez a hiba a következő lépésekben automatikusan korrigálódik, de emiatt általában a lépések száma megnövekszik.

Annak eldöntése, hogy a transzformáció az oszlopokkal vagy sorokkal kezdődjék-e, aszerint történhet, hogy összehasonlítjuk a

$$\sum_{i=1}^{m} u_i^{(0)} \alpha_i \quad \text{és} \quad \sum_{j=1}^{n} v_j^{(0)} \beta_j$$

sorozatösszegeket. Amennyiben

$$\sum_{j=1}^{n} v_j^{(0)} \beta_j \geq \sum_{i=1}^{m} u_i^{(0)} \alpha_i \quad \text{akkor}$$

mint itt tettük, 1. lépésben a $v_j^{(0)}$-t vonjuk ki,

míg ha $$\sum_{j=1}^{n} v_j^{(0)} \beta_j < \sum_{i=1}^{m} u_i^{(0)} \alpha_i \quad \text{akkor}$$

1. lépésben az $u_i^{(0)}$-t célszerűbb kivonni.

Ezt betartva gyorsabban juthatunk a C_p matrixhoz.

Példánkban

$$\sum_{i=1}^{m} u_i^{(0)} \alpha_i = 1 \cdot 200 + 1 \cdot 80 + 1 \cdot 130 + 2 \cdot 90 = 590 \quad \text{és}$$

$$\sum_{j=1}^{n} v_j^{(0)} \beta_j = 3 \cdot 30 + 2 \cdot 210 + 2 \cdot 60 + 1 \cdot 80 + 1 \cdot 120 = 650$$

tehát helyes sorrendet alkalmaztunk.

Mivel a 2. lépés minimális fedővonalainak száma

$$s^{(2)} < M \qquad \text{mert}$$
$$380 < 500 \qquad \text{a}$$

C_2 matrixot tovább kell transzformálni.

3. lépés. Megkeressük a legkisebbet azon elemek közül, melyek nem fekszenek benne az S_2 fedővonal-rendszerben és ezt h_2-nek nevezzük. Vonjuk le h_2-őt a C_2 matrix minden olyan eleméből, melyet nem takar az S_2 fedővonal-rendszer és adjuk hozzá az S_2 fedővonalainak metszéspontjaiban levő elemekhez. Így nyerjük a C_3 matrixot, amit az előzőkhöz hasonlóan szintén beírunk táblázatunkba (3, index).

Jelölésekkel megmagyarázva, ebben a lépésben valamennyi sor valamennyi eleméből levonjuk

$u_i^{(2)'} = h_2$-őt, majd hozzáadjuk minden olyan sor elemeihez

$$u_i^{(2)''} = h_2\text{-őt},$$

és minden olyan oszlop elemeihez

$v_j^{(2)''} = h_2$-őt, melyeket az S_2 fedővonal-rendszer takar.

Példánkban $h_2 = 1$.

A C_3 matrix „0" elemeit fedő minimális S_3 (3-mal jelölt és multiplicitásukkal vett) fedővonalainak száma

$$s^{(3)} = 80 + 90 + 210 + 80 = 460$$

mivel $s^{(3)} < M$, a transzformációt a 3. lépésnek megfelelően addig ismételjük, míg $s^{(p)} = M$ lesz.

4. lépés. Az S_3 fedővonal-rendszer által nem takart legkisebb elem $h_3 = 1$. A C_4 matrix $c_{ij}^{(4)}$ elemeit a 6. táblázatban a $c_{ij}^{(3)}$ elemek alá írjuk

(4, index). Megkeressük a (④ jelű) minimális fedővonal-rendszert. Azt találjuk, hogy

$$s^{(4)} = 80 + 30 + 210 + 60 + 80 = 460 < M$$

5. lépés. Már magyarázat nélkül:

$$h_4 = 1$$

$$s^{(5)} = 200 + 80 + 130 + 90 = 500 = M, \text{ tehát}$$

a $C_5 = C_p$ matrix burkoltan már 500 független pontot tartalmaz, tehát az x_{ij} mennyiségek minden egysége számára kijelöl egy eljárásvektort, vagyis négyszöget. Ezeket a négyszögeket táblázatunkban egy csillaggal jelöljük meg.

Ezzel megtaláltuk az optimális megoldás eljárásvektorait. Mint az előzőkből már tudjuk, az optimális megoldás „$m + n - 1$" számú négyszöget használ fel. Példánkban

$$m + n - 1 = 4 + 5 - 1 = 8 \quad \text{és}$$

a C_5 matrix éppen annyi „0" elemet tartalmaz. Nem is tartalmazhat többet, mert már meggyőződtünk arról, hogy ennek a feladatnak csak egy optimális megoldása van. Láthatjuk, hogy a C_5 matrix azonos az 5. táblázatban leírt $[c_{ij} - c_{ij}]$ javítófaktorok matrixával.

Ezek után a 6. táblázatban megszerkesztjük az optimális megoldáshoz tartozó kombinációs matrixot. Megkeressük azokat az oszlopokat és sorokat, melyekben csak egy „0" elem van és ezeknek négyszögeihez az illető sor vagy oszlop teljes α_i, ill. β_j mennyiségét rendeljük. A többi négyszögben, ahol „$c_{ij}^{(5)} = 0$", a tervezett mennyiség már kiadódik, mert a sorok és oszlopok összege meg kell egyezzék azok multiplicitásával. Így olyan megoldást nyertünk, mely eleget tesz az (1), (2), (3) és (4) feltételének. Az egyes eljárásvektorhoz rendelt tervszámokat a 6. táblázatban vastagabb számokkal tüntettük fel.

Írjuk be a 7. táblázatba ezt a megoldást, mely azonosan jelentkezik a 4. és 6. táblázatban.

7. táblázat

Az optimális megoldás

Rendeltetés / Kiindulás		Cím 1	2	3	4	5	Összesen
Cím	1		120		80		200
	2					80	80
	3		90	40			130
	4	30		20		40	90
Összesen		30	210	60	80	120	500

A magyar módszerrel kapott megoldás jellemzőjének (4) függvényértéke a lépésekkel párhuzamosan is számítható

$$\begin{array}{llcr}
1.\ \text{lépésből} & \sum_{j=1}^{n} v_j^{(0)} \beta_j & = & 830 \\
2.\ \text{lépésből} & \sum_{i=1}^{m} u_i^{(1)} a_i & = & 0 \\
3.\ \text{lépésből} & h_2 (M - s^{(2)}) & = & 120 \\
4.\ \text{lépésből} & h_3 (M - s^{(3)}) & = & 40 \\
5.\ \text{lépésből} & h_4 (M - s^{(4)}) & = & 40 \\
\hline
 & & & 1030 = \text{maximum},
\end{array}$$

mert az S_5 már az összes sort vagy összes oszlopot takarja, tehát h_5 nem választható és $(M - s^{(5)}) = 0$, vagyis az összeg tovább nem növelhető.

Ugyanis ha a (18) egyenlőtlenséget a (4) függvénybe behelyettesítjük

$$\sum_i \sum_j c_{ij}^{(0)} x_{ij} \geqq \sum_i u_i a_i + \sum_j v_j \beta_j \tag{19}$$

és ebből látjuk, hogy kettős problémát oldottunk meg, mert a (19) egyenlőtlenség minimális baloldali értéke egyenlő a maximális jobboldali értékkel. Vagyis

$$\min \left[\sum_i \sum_j c_{ij}^{(0)} x_{ij}\right] = \max \left[\sum_i u_i a_i + \sum_j v_j \beta_j\right] \tag{20}$$

és a (4) függvény duálja

$$\sum_i u_i a_i + \sum_j v_j \beta_j = \max, \tag{21}$$

ahol u_i és v_j eleget tesz a (18) egyenlőtlenségnek és az

$$u_i = u_i^{(0)} + u_i^{(1)} + u_i^{(2)} + \cdots + u_i^{(p)},\ \text{valamint}$$

a $$v_j = v_j^{(0)} + v_j^{(1)} + v_j^{(2)} + \cdots + v_j^{(p)}\ \text{redukáló}$$

sorozatok.

Az $u_i^{(2)}, \ldots, u_i^{(p)}$ és a $v_j^{(2)}, \ldots, v_j^{(p)}$ jelentését a 3. lépésben magyaráztuk meg.

Az elfajult szállítási problémákon túlmenően ez a módszer mindig előnyösen használható olyan kisebb kombinációs matrixok megoldásánál, melyek hosszúkás alakúak, vagy melyeknél egy jó kiinduló megoldás felvétele nagy nehézséget okoz.

A minimális fedővonal-rendszer meghatározására nincs algoritmus, de egy kis gyakorlat után annak megkeresése már igen gyorsan elvégezhető.

Megemlítjük, hogy az 1. és 2. lépést általában olyankor is érdemes elvégezni, amikor disztribuciós módszerrel oldjuk meg a feladatot, mert így kisebb címszámokkal dolgozhatunk.

*

Az itt tárgyalt három módszer összehasonlításaként a következőket állapíthatjuk meg.

1. A *szimplex módszer* szállítási feladatoknál a használható $\mathbf{a}_{ij}$ eljárásvektorokból bázist képez és azt újabb $\mathbf{a}_{ij}$ vektorok bevonásával addig javítja, míg végül olyan bázishoz jut, amelyre vonatkoztatva a közvetett fajlagos jellemzők közül egyiknek értéke sem nagyobb a megfelelő tényleges fajlagos jellemzőnél, amikor is az összes javítófaktorra érvényes

$$c_{ij} - \bar{c}_{ij} \geqslant 0 ,$$

ahol $i = 1, 2, \ldots, m$ és $j = 1, 2, \ldots, n$.

2. A *disztribuciós módszer* „$m + n - 1$" eljárásvektorral kifejezhető és összefüggő, nyitott poligonnal ábrázolható lehetséges megoldást vesz fel és azt addig javítja, míg egy olyan megoldáshoz jut, amelynek fajlagos jellemzői által a $c_{kl} = \bar{c}_{kl} = u_i + v_j$ összefüggés alapján meghatározott u_i és v_j komponensekkel egy $\underset{(m,n)}{[0]}$ matrixot transzformálva, az így kapott matrix valamennyi eleme eleget tesz az alábbi egyenlőtlenségnek

$$u_i + v_j \leqslant c_{ij} .$$

3. A *magyar módszer* az előbbivel ellentétesen a fajlagos jellemzők matrixát addig transzformálja, míg végül a matrix zéró elemeit már csak olyan fedővonal-rendszerrel lehet letakarni, melyhez tartozó fedővonalak száma, vagyis multiplicitásainak összege a terv összmennyiségével egyenlő, tehát

$$s^{(p)} = M$$

és az u_i és v_j redukáló tényezők megfelelnek az alábbi egyenlőtlenségnek

$$c_{ij} - (u_i + v_j) \geqslant 0 .$$

*

A lineáris programozás módszereinek tanulmányozása után pár alkalmazást mutatunk be.

I. feladat. A. R. Frattasi olasz szerző nemrégen ismertetett egy egyszerű vasúti alkalmazást [9]. Nyolc vasúti állomás három pályaudvarról kapja az üres vagonokat, melyeknek *szétosztását* Frattasi a szimplex módszerrel végezte el.

8. táblázat

Napi kocsiszétosztás

Honnan \ Hova		1 Torino	2 S. Giuseppe Cairo	3 Savona	4 Alessandria	5 Casale	6 Vercelli	7 Trecate	8 Novara	Készlet összesen (a_i)
1	Torino	x_{11}	x_{12}	x_{13}	x_{14}	x_{15}	x_{16}	x_{17}	x_{18}	800
2	Alessandria	x_{21}	x_{22}	x_{23}	x_{24}	x_{25}	x_{26}	x_{27}	x_{28}	400
3	Novara	x_{31}	x_{32}	x_{33}	x_{34}	x_{35}	x_{36}	x_{37}	x_{38}	200
Szükséglet összesen : (β_j)		220	700	300	50	25	25	60	20	1400

9. táblázat

C_0 távolság-mátrix

Honnan \ Hova	1	2	3	4	5	6	7	8
1	—	124	146	91	101	79	110	101
2	91	86	108	—	35	56	78	67
3	101	151	172	67	54	24	11	—

10. táblázat

$C_{0,1,\ldots,4}$ matrix és az optimális kombinációk

		① ②				② ③	② ① ③	② ① ③	② ① ③	
	0,	0	124	146	91	101	79	110	101	
	1,	0	38	38	91	66	55	99	101	
	2,	0	19	19	72	47	55	99	101	
③	3,	0	0	0	53	47	55	99	101	**800**
④	4.	0	0(**350**)	0(**230**)	53	73	81	125	127	
		220	**580**							
	0,	91	86	108	0	35	56	78	67	
①	1,	91	0	0	0	0	32	67	67	
②	2,	110	0	0	0	0	51	86	86	
③	3,	129	0	0	0	19	70	105	105	**400**
④	4,	129	0(**350**)	0 (—)	0	45	96	131	131	
			120	**230**	**50**					
	0,	101	151	172	67	54	24	11	0	
	1,	101	65	64	67	19	0	0	0	
	2,	101	46	45	48	0	0	0	0	**200**
	3,	101	27	26	29	0	0	0	0	
④	4,	75	1	0	3	0	0	0	0	
				70		**25**	**25**	**60**	**20**	
		220	**700**	**300**	**50**	**25**	**25**	**60**	**20**	**1400**
		0	86	108	0	35	24	11	0	

Itt bemutatjuk, hogy ez a feladat a magyar módszerrel milyen gyorsan megoldható. A feladatot a 8. és 9. táblázattal ismertetjük. A 8. táblázat a rendelkezésre álló és az igényelt üres vagonok mennyiségét (készlet és szükséglet), a 9. táblázat pedig az állomások közötti vasúti távolságokat (km) sartalmazza. Az üres kocsi fuvarköltsége 30 líra/km és az olasz szerző a szétosztást úgy tervezte meg, hogy az összes fuvarköltség minimum legyen. Mivel itt a fuvarköltség arányos a távolsággal, ugyanazt az optimális programot kapjuk akkor is, ha a szállítási teljesítményt tesszük minimummá. A távolság-matrix transzformációját a 10. táblázatban végeztük el, az alábbi lépésekben.

1. lépés : $v_1 = 0,\ v_2 = 86,\ v_3 = 108,\ v_4 = 0,\ v_5 = 35,\ v_6 = 24,$
$v_7 = 11,\ v_8 = 0,$
$s^{(1)} = 725 < 1400$

2. lépés : $h_1 = 19$; $s^{(2)} = 750 < 1400$

3. lépés : $h_2 = 19$; $s^{(3)} = 1320 < 1400$

4. lépés : $h_3 = 26$; $s^{(4)} = 1400 = M$.

A C_4 matrix 11 zéró elemet tartalmaz, míg az „$m + n - 1 = 10$”. Megállapíthatjuk, hogy „$m + n - 1$” útvonal felhasználásával csak két optimális megoldás szerkeszthető, ezeket a 10. táblázatban vastag számokkal tüntettük fel, ahol a variáns megoldás eltérő értékeit zárójelbe írtuk. Természetesen mindkét program azonos — 121730 vagonkm — szállítási teljesítménnyel bonyolítható le.

II. feladat. A lineáris programozás érdekes alkalmazását ismerteti *Z. Rumjánceva* szovjet szerző [10] a repülőgépek *beosztási* problémájának megoldására. Egy repülőosztály, mely 40 db különböző nagy-típusú géppel rendelkezik, öt légiúton bonyolítja le a személy-, teher- és postaforgalmat.Ennek megfelelően és a közbenső légikikötőkben való várakozást is figyelembe véve, meghatározható tkm-ben minden légiút szállítási szükséglete. A különböző géptípusokhoz tartozó repülőgépek számából, az egy gépre jutó havi átlagos óraszámból és a gépek tkm/óra átlagos teljesítőképességéből meghatározható tkm-ben a különböző típusú csoportoknak szállítási kapacitása. A géptípusok kapacitását úgy kell kihasználni, vagyis a gépeket úgy kell *beosztani,* hogy a teljes szállítást minimális üzemelési költséggel lehessen teljesíteni. Ezért valamilyen egységben meg kell adni az egyes géptípusoknak az egyes légiutakhoz tartozó egy tkm-re eső fajlagos üzemelési önköltségét. A szovjet szerző az A-típusú repülőgép fajlagos önköltségét veszi egységnek. Mivel a szállítási szükséglet 31 millió tkm, a kapacitás pedig 33 mill. tkm, beállítható esetleges szükséglet kielégítésére egy Y jelű tartalék-légiút, amelyhez tartozó c_{ij} értékeket mi zérusnak vesszük fel.

11. táblázat

Repülőgépek beosztása

1(=)1 mill. tkm		Légi utak 1	2	3	4	5	γ	Szállítási kapacitás összesen
Repülőgép-típusok	A	*5,4*	*6,6*					12,0
	B	*4,6*						4,6
	C		*1,4*	*1,6*	*4,0*	*3,0*		10,0
	D			*4,4*			*2,0*	6,4
Szállítási szükséglet összesen		10,0	8,0	6,0	4,0	3,0	2,0	33,0

A feladatot és az optimális megoldást a 11. táblázat ismerteti, a 12. táblázat a fajlagos üzemelési önköltségeket tartalmazza. A 13. táblázatban disztribuciós módszerrel igazoltuk, hogy a 11. táblázat beosztási kombinációja tényleg az optimum. A 12. táblázat szerint a B-2, D-1 és D-2 beosztás nem vehető figyelembe, ezért ezeknek a beosztásoknak négyszögeit a javítófaktorok matrixában üresen hagytuk. A 13. táblázatból azt is megtudjuk, hogy még egy optimális beosztás lehetséges, ennek eltérő tervszámait zárójelben tüntettük fel.

C_0 költségmatrix

12. táblázat

Rep.-gépek típusa \ Légi utak	1	2	3	4	5	y
A	0,9	1,0	0,8	0,7	0,7	0
B	2,1	—	2,0	1,8	2,0	0
C	2,5	2,6	2,2	1,7	1,6	0
D	—	—	3,0	2,8	3,0	0

13. táblázat

Optimális kombináció és a javítófaktorok matrixa

$(-u_i)$ \ v_j	3,3	3,4	3,0	2,5	2,4	0	
2,4	0,9 **5,4** 0 **(4,0)**	1,0 **6,6** 0 **(80)**	0,8 +	0,7 +	0,7 +	0 +	**12,0**
1,2	2,1 **4,6** 0		2,0 +	1,8 +	2,0 +	0 +	**4,6**
0,8	2,5 **1,4** 0	2,6 **1,4** 0 **(—)**	2,2 **1.6** 0	1,7 **4,0** 0	1,6 **3,0** 0	0 +	**10,0**
0			3,0 **4,4** 0	2,8 +	3,0 +	0 **2,0** 0	**6,4**
	1,00	**8,0**	**6,0**	**4,0**	**3,0**	**2,0**	**33,0**

A felvett egységben kifejezve, a teljes szállítás üzemelési önköltsége

$$K_v = 53,08$$

III. feladat. Végül bemutatunk egy olyan alkalmazási lehetőséget, mely *elfajult beosztási problémaként* jelentkezik. Többször hallani azt a panaszt, hogy az autóbusz- és villamosvasúti-üzem dolgozói a forgalom kezdete előtt és a forgalom leállta után milyen nehezen, mennyi időveszteséggel jutnak el lakásukról a szolgálatbalépésük helyére, illetve a szolgálat befejezésének helyéről lakásukra.

Az optimális beosztás meghatározásával a közlekedési alkalmazottak munkája bizonyára hatékonyabbá tehető. Ehhez adott az egyes végállomásokon, telephelyeken a forgalom megindulásakor munkába álló dolgozók (pl. kalauzok) száma mint szükséglet, és ugyanezek a dolgozók lakhely szerinti bontásban, mint rendelkezésre álló személyek. A beosztást úgy tervezzük meg, hogy az összes munkába lépő dolgozó (a legkedvezőbb útvonalakon és szolgálati kocsikon) együttesen a minimális idő alatt jusson el munkahelyére. A beosztás fajlagos jellemzőinek tehát azokat az időszükségleteket vesszük, amelyek alatt az egyes dolgozók a különböző munkahelyekre eljuthatnak. Egy nagy városban ez a probléma már egy hatalmas matrix transzformálását teszi szükségessé, amire elektronikus számológép nélkül gondolni sem lehet. Egy kisméretű ilyen feladatot azonban a magyar módszerrel kézi számítás útján gyorsan megoldhatunk.

Dolgozók beosztása *14. táblázat*

Munkahely \ Lakhely	1	2	3	4	5	6	7	8	9	10	Szolgálatba lépő dolgozók száma
I					*1*	*1*			*1*		3
II							*1*			*1*	2
III			*1*								1
IV	*1*	*1*		*1*			*1*	*1*			5
Szolgálatba induló dolgozók száma	1	1	1	1	1	1	2	1	1	1	11

A 14. táblázat 4 szolgálatbalépési helyet és 10 lakhelyet tartalmaz 11 dolgozóval. A dolgozóknak az egyes munkahelyek eléréséhez szükséges időráfordítását a 15. táblázat percekben tartalmazza. Nyilvánvaló, hogy a lehetséges megoldások nem használhatnak fel

$$m + n - 1 = 13$$

beosztásvektort, amikor a feladat szerint 11 dolgozót kell beosztani.

C_0 fajlagos időszükséglet matrix *15. táblázat*

Munkahely \ Lakhely	1	2	3	4	5	6	7	8	9	10
I	41	61	53	57	17	10	98	8	13	79
II	20	65	23	38	74	24	19	97	50	14
III	91	53	3	64	66	39	98	24	81	28
IV	29	37	18	47	59	62	35	17	48	53

Ezért elfajult ez a probléma. A magyar módszer alkalmazása ilyenkor feltétlenül előnyös. A fajlagos jellemzők matrixát addig kell transzformálnunk, míg **11** független zérót kapunk, vagyis míg olyan C_p matrixhoz jutunk, melyben $s^{(p)} = M$. A matrix transzformálását a 16. táblázatban az alábbi lépésekben végeztük el.

1. lépés : $v_1 = 20,\ v_2 = 37,\ v_3 = 3,\ v_4 = 38,\ v_5 = 17,\ v_6 = 10,$
$v_7 = 19,\ v_8 = 8,\ v_9 = 13,\ v_{10} = 14$
$s^{(1)} = 7 < 11$

2. lépés : $h_1 = 9$; $s^{(2)} = 10 < 11$

3. lépés : $h_2 = 6$; $s^{(3)} = 10 < 11$

4. lépés : $h_3 = 1$; $s^{(4)} = 11 = M$

16. táblázat

$C_{0,1,\ldots,4}$ matrix és az optimális beosztás

		2 3	1 2 3	3	2 3				2 3			
	0,	41	61	53	57	17 *	10 *	98	8	13 *	79	
1	1,	21	24	50	19	0	0	79	0	0	65	
2	2,		33									**3**
3	3,	27	39		25	0	0		6	0		
4	4,	28	40	51	26	0 **1**	0 **1**		7	0 **1**		
	0,	20	65	23	38	74	24	19 *	97	50	14 *	
1	1,	0	28	20	0	57	14	0	89	37	0	
2	2,		37									**2**
3	3,	6	43		6			0	95		0	
4	4,	7	44	21	7			0 **1**	96		0 **1**	
	0,	91	53	3 *	64	66	39	89	24	81	28	
1	1,	71	16	0	26	49	29	70	16	68	14	
2	2,		25									**1**
	3,	77	31	0	32				22			
4	4,			0 **1**		48	28	69		67	13	
	0,	29 *	37 *	18	47 *	59	62	35 *	17 *	48	53	
	1,	9	0	15	9	42	52	16	9	35	39	
	2,	0		6	0	33	43	7	0	26	30	**5**
	3,	0	0	0	0	27	37	1	0	20	24	
4	4,	0 **1**	0 **1**	0	0 **1**	26	36	0 **1**	0 **1**	19	23	
		1	**1**	**1**	**1**	**1**	**1**	**2**	**1**	**1**	**1**	**11**
		20	37	3	38	17	10	19	8	13	14	

A független zérók négyszögét *-gal jelöltük meg. Megfigyelhetjük, hogy az $\mathbf{a}_{43}$ beosztásvektor négyszögében a javítófaktor ugyan zérus, de nem független, mert oszlopában már van egy független zéró. Tudniillik a sor vagy oszlop független zéró-elemeinek száma nem lehet nagyobb, mint a sor vagy oszlop multiplicitása. A megjelölt négyszögek felhasználásával a 16. táblázatban megszerkesztettük az optimális beosztást, melynek végrehajtása összesen 241 perc, kereken 4 óra időfelhasználást tesz szükségessé. Végül az optimális megoldást beírtuk a 14. táblázatba.

IRODALOM

[1.] JÁNDY G.: Szállítási feladatok lineáris programozása. Közlekedéstudományi Szemle, 1958. 6. szám.

[2.] JÁNDY G.: Tereprendezés optimális földszállításának tervezése. Mélyépítéstudományi Szemle, 1958. 7. szám.

[3.] KREKÓ B.—BACSKAY Z.: Bevezetés a lineáris programozásba. Budapest,1957.

[4.] L. SZ. PONTRJAGIN: Kombinatorikus topológia. Budapest, 1955.

[5.] T. C. KOOPMANS: Activity Analysis of Production and Allocation. New York, 1951.

[6.] C. W. CHURCHMAN, R. L. ACKOFF, E. L. ARNOFF, Introduction to Operations Research. New York, 1957.

[7.] KÖNIG D.: Graphok és matrixok.
EGERVÁRI J.: Matrixok kombinatorius tulajdonságairól. Matematikai és Fizikai Lapok. 1931. jan.—jún.

[8.] H. W. KUHN: The Hungarian Method for the Assignment Problem. Naval Research Logistics Quarterly 2 (1955).

[9.] A. R. FRATTASI: Sulla programmazione ,,ottima'' della distribuzione dei carri vuoti dai centri di concentramento ai centri utilizzatori. Ingegneria Ferroviaria, 1957. 12. szám.

[10.] Z. RUMJÁNCEVA: Linejnoje programmirovanije. Grazsdanszkaja Aviacija, 1958. 8. szám.