A HOZZÁRENDELÉSI PROBLÉMA MEGOLDÁSA KÉT MEGKÖZELÍTÉSBEN

TWO DIFFERENT APPROACHES TO SOLVING THE ASSIGNMENT PROBLEM

Kiss László
Budapesti Műszaki Főiskola, Rejtő Sándor Könnyűipari és Környezetmérnöki Kar, Médiatechnológiai Intézet

KIVONAT

Főiskolánkon több éve oktatunk operációkutatást Matematikai programozás néven a műszaki menedzserek számára, illetve a jelen félévben indult környezetinformatikai szakirányon is szerepel ez a tantárgy. Ebbe az oktatásba kapcsolódtam be nemrégen én is, s találkoztam a hozzárendelési problémával újra, egyetemi tanulmányaim óta első alkalommal. A tanítás során a diákok szemszögéből nézve több kérdés is felmerült bennem a megoldás módszerével kapcsolatosan:

- *Mindegy-e, hogy a probléma mátrixában előbb a sorok minimumait vonjuk ki a sorokból és azután az oszlopok minimumait az oszlopokból, vagy megfordítva?*
- *Hogyan lehet független zéró rendszert találni?*
- *Ha nincs független zéró rendszer, akkor hogyan találunk minimális lefedő vonalrendszert?*
- *Hány vonalból áll ez a rendszer, és mi a kapcsolata a megtalált már független zérókkal?*
- *Miért a vonallal nem lefedett elemek minimumát vonjuk ki a mátrix e részéből, s adjuk hozzá a kétszeresen vonallal lefedettekhez? Ez a módszer milyen módon hat az optimális megoldásra? Egyáltalán megengedett-e?*
- *Miért fejeződik be véges számú lépésben a megoldási algoritmus?*

Ezeket a kérdéseket szeretném körüljárni; különösképpen a független zéró rendszer és a minimális lefedő vonalrendszer meghatározásának problémáját. Készítettem - két, alapvetően különböző gondolatmenetnek megfelelően - számítógépes megoldásokat is, amikből szintén bemutatnék néhányat.

ABSTRACT

At Budapest Polytechnic, a subject about operational research - under the name "Mathematical programming" - has been part of the technical management curriculum for several years, and it has also been introduced for those choosing the newly launched speciality "Environment informatics". Having recently joined the team responsible for this subject, I came across the assignment problem for the first time since my graduate studies. While giving lectures, I thought of a number of questions a student may be inclined to ask regarding the method used to solve this problem:

- *Does it make a difference if we subtract row minimums from rows first and column minimums from columns later, or vice versa?*
- *How can a set of independent zeroes be found?*
- *If such a set does not exist, how do we obtain a minimal set of covering lines?*

- *How many lines are there in this set, and how does it relate to the set of independent zeroes already found?*
- *Why do we subtract the minimum of values not yet covered from this part of the matrix, and add the same value to those covered twice? How does this affect the optimal solution? Is it even acceptable to do this?*
- *Can we be sure that the algorithm only requires a finite number of steps?*

During my presentation I'll cover the above questions, focusing especially on finding independent zeroes and minimal sets of covering lines. I have prepared software solutions based on two different approaches; these will also form part of the presentation.

KULCSSZAVAK/ KEYWORDS
Operációkutatás, programozás, algoritmusok
Operational research, programming, algorithms

BEVEZETÉS

Amint az összefoglalásból is érzékelhető, a kulcsprobléma számomra egy-egy téma oktatásának mikéntje. Egyetemista koromban, illetve később az oktatásban tapasztaltak során egyre inkább érlelődött bennem a gondolat, hogy szükséges az adott probléma teljes körüljárására törekedni. Ezalatt azt értem, hogy az alkalmazott módszert nem elég csak részben megértetni a hallgatósággal, hanem minden felmerült kérdésre a lehető legrészletesebb és – talán még hangsúlyozottabban – a legszemléletesebb választ kell adni. Előadásomban erre szeretnék példát hozni.

A HOZZÁRENDELÉSI PROBLÉMA

Adott egy nemnegatív egész számokat tartalmazó négyzetes mátrix, aminek minden sorából egy-egy elemet választunk úgy, hogy azok közül semelyik kettő ne legyen egy oszlopban. Keressük a lehetséges választások közül azt vagy azokat, amelyek esetén a választott számok összege minimális.

A HOZZÁRENDELÉSI PROBLÉMA MEGOLDÁSÁNAK MÓDSZERE

Levonjuk a sorok minimumait a sorokból, majd az oszlopok minimumait az oszlopokból, és keresünk – amennyiben lehetséges - egy független zéró rendszert, azaz minden sorban egy zérót úgy, hogy közülük semelyik kettő ne legyen azonos oszlopban. Ha találtunk ilyet, akkor az eredeti mátrixban a nekik megfelelő helyen lévő elemek adják a feladat egy megoldását, és összegük a keresett minimum. Ha nem találtunk független zéró rendszert, akkor választunk minimális számú sort és oszlopot, hogy rajtuk kívül más sorban vagy oszlopban ne legyen zéró (minimális lefedő vonalrendszer). A nem lefedett elemek minimumával csökkentjük azokat, hozzáadjuk a kétszeresen lefedettekhez, és visszatérünk a független zéró rendszer kereséséhez.

A FELVETETT KÉRDÉSEK TÁRGYALÁSA

Legyen a problémában szereplő mátrix sorainak és oszlopainak száma N. A kérdéseket a válaszadás lehetséges sorrendjében fogalmazzuk meg.

1. kérdés: szabad-e egy sor vagy oszlop minden eleméhez ugyanazt a számot hozzáadni? Befolyásolja-e ez a megoldást?

Tegyük fel, hogy adott a feladatnak egy megoldása, M. Legyen az M-ben szereplő számok összege S. Ha valamelyik sorhoz vagy oszlophoz egy k számot hozzáadunk (természetesen k lehet negatív is), akkor, mivel az adott sorból vagy oszlopból M csupán egyetlen elemet tartalmaz, ezért a benne lévő elemek összege S+k-ra változik. Ha az így keletkezett mátrixban létezne egy M* megoldás, amire az elemek összege kisebb, mint M-ben, azaz kisebb, mint S+k, akkor ugyanabból a sorból illetve oszlopból, amelyikhez k-t hozzáadtuk, azt levonva

M*-ban az elemek összege (-k)-val változik, s kisebb lesz, mint S. Ez ellentmond annak a feltevésünknek, miszerint M megoldása volt a problémának.

Megállapíthatjuk tehát, hogy ha egy sorhoz vagy oszlophoz ugyanazt a számot hozzáadjuk, akkor a megoldás nem, csak a benne lévő elemek összege változik.

Megengedett tehát a sorok és oszlopok csökkentése a minimumaikkal.

2. kérdés: megengedett-e a lefedetlen elemek minimumával csökkenteni azokat, és hozzáadni a kétszeresen lefedettekhez?

Vonjuk ki a lefedetlenek minimumát minden sor elemeiből, és adjuk hozzá a lefedett sorokban, illetve oszlopokban lévő elemekhez. Látható, hogy ezzel minden elem a kérdésben megfogalmazottaknak megfelelően változik, így a módszerünk nem befolyásolja a feladat megoldását.

3. kérdés: véges-e az algoritmus? Mindegy-e, hogy előbb a sorokból, és azután az oszlopokból vonjuk le azok minimumát, vagy fordítva?

Mivel a mátrixunk nemnegatív egész számokat tartalmaz, ezért a benne lévő elemek összege véges. A 2. kérdésben megfogalmazott és alkalmazott módszer tehát annyiszor N-szer vonja le a minimumot, ahány sora van a mátrixnak (N), és csak annyiszor N-szer adja hozzá, amennyi a lefedett sorok és oszlopok együttes száma (<N, később belátjuk), tehát a mátrixban az elemek összértéke csökken, s így az algoritmus véges számú lépésben véget ér. (Meg kell jegyezni, hogy a mátrix az algoritmus végrehajtása során mindig nemnegatív elemeket tartalmaz!)

Készítettem egy alkalmazást, amellyel azt vizsgáltam sok-sok futtatással, hogy hogyan változik egy-egy mátrixban a sor- illetve oszlopminimumok levonásával az elemek összértéke. A gyakorlat azt mutatta, hogy ez a szám a kiindulási mátrixtól függ, és teljesen esetlegesen hat az algoritmus folytatására, azaz mindegy, melyiket hajtjuk végre először.

4. kérdés: hogyan találjunk független zéró rendszert a csökkentett elemeket tartalmazó mátrixban? Ha nincs független zéró rendszer, akkor milyen módon határozzuk meg a lefedő vonalrendszert?

Előfordul, hogy ezeket a kérdéseket egyszerűen nem tárgyalják az oktatás során.

Ha igen, akkor sok esetben az általam józan paraszti ésszel való megközelítésnek nevezett módszer kerül terítékre. Ez abban áll, hogy keressünk olyan sorokat vagy oszlopokat, amelyekben egyetlen nulla található. Nyilvánvalóan célszerű ezeket választani. Az így választott nullák sorait és oszlopait kizárjuk, és a maradékkal dolgozunk tovább. Ha már ilyen sor, illetve oszlop nincsen, akkor valójában megáll a tudomány, és mindenféle ötletek szerint történik a további nullák választása. Az egyik lehetséges ötlet, hogy a legtöbb nullát tartalmazó sorból választjuk az elsőt. Ez a megoldás szerencsés esetekben lehet célravezető, de semmiképpen nem mondható általánosan alkalmazható algoritmusnak. Az 1. táblázatban a már sor- és oszlopminimumokkal csökkentett mátrixban például jól működik ez a módszer, mert a 2. és a 4. sorban egy-egy nulla van, ezért azokat választjuk. Ezzel az 5. sorban is csak egy választható nulla marad, tehát azt is választjuk. Áttérve az oszlopokra, az 1. és a 2. oszlopban is egy-egy nulla van, így azokat választva megtaláltuk megoldást.

1. táblázat. Van független zéró rendszer.

	1	2	3	4	5
1	0	10	19	0	0
2	3	15	7	0	7
3	1	0	18	21	0
4	8	3	0	2	9
5	1	2	0	20	0

A 2. táblázatban látható mátrixban az 1. sorban, majd az 5. és 7. oszlopban választhatunk ezzel a módszerrel nullát, és azután teljes a tanácstalanság.

Még problémásabb a helyzet akkor, ha nem találunk független zéró rendszert (mint ahogy ez a 2. táblázatban látható mátrixban nincs is), és meg kell határozni a minimális lefedő vonalrendszert. Ebben a gondolatkörben az a módszer, hogy újfent megkeressük a sorokban az egyedülálló nullákat, és lefedjük azok oszlopait, illetve az oszlopokban egyedülállókat, és lefedjük azok sorait. Ez alapvetően jó gondolat, hiszen minden nullát, így az egyedül állókat is le kell fedni, és nyilvánvalóan jobb, ha nem azt a vonalat fedjük le, amelyben egyedül vannak. A baj csupán az, hogy ismét tanácstalanul állunk abban a helyzetben, ha több nulla van egy sorban vagy oszlopban.

2. táblázat. Nincs független zéró rendszer. Hibás a lefedő vonalrendszer.

	1	2	3	4	5	6	7	8
1	0	599	337	602	638	911	668	139
2	406	0	245	476	598	0	0	0
3	400	302	0	354	409	0	574	133
4	370	317	224	0	411	0	644	83
5	341	551	310	460	0	0	0	0
6	563	0	0	0	350	763	407	96
7	372	0	240	0	477	730	628	122
8	385	0	228	0	496	741	644	105

Készítettem egy alkalmazást, amelyik a fenti gondolatmenet alapján működik. Amikor több nulla van egy sorban, akkor kiválasztja az elsőt, és miután annak sorában és oszlopában a további nullákat (-1)-re állította, újra visszatér ahhoz, hogy van-e olyan sor vagy oszlop, amiben csupán egyetlen nulla található. Az összes lehetséges megoldást megkeresi és kiírja. Ez az alkalmazás természetesen mindig tökéletesen működik, ha a sor- és oszlopcsökkentések után van független zéró rendszer a mátrixban. Ha nincs, akkor a lefedő vonalakat a fent leírtak alapján határozza meg, amennyiben egyetlen nullát talált a sorban vagy oszlopban. Ha több nulla volt, akkor kiválasztja az elsőt, és lefedi annak oszlopát, s a sorában és oszlopában levő többi nullát (-1)-re állítja, majd újra próbálkozik azzal, hogy van-e valamelyik sorban vagy oszlopban egyetlen nulla. Ha nem talált ilyet, akkor ismét az első lehetséges nullát választva folytatja a módszert mindaddig, amíg marad nulla a mátrixban. A lefedett sorok és oszlopok színezésekor visszaírja a (-1)-eket nullára.

Az alkalmazás az esetek nagy részében jól is működik, de előfordul olyan, hogy nem sikerül minden nullát lefedni, mint például a 2. táblázatban szereplő mátrix esetén. Az algoritmus javítható (igaz, felmerül, hogy van-e értelme), hiszen az így le nem fedett nulla sorában vagy oszlopában lévő választott nulla (ilyen biztosan van, különben őt magát választhatnánk) lefedő vonalát kell megváltoztatnunk (ha sorát fedtük le, akkor oszlopát, ha oszlopát fedtük le, akkor sorát kell lefednünk). Ekkor azonban újabb lefedési hibába ütközhetünk, és újra javítanunk szükséges. Látható, ez egy bizonytalan módszer, és adott esetben rendkívül hosszadalmas is lehet. A 2. táblázatban ez a javítás egy lépésben megoldható, ha a mátrix 6. sorának 3. oszlopában lévő elemének, mint kiválasztott nullának nem a sorát, hanem az oszlopát fedjük le. Ismerve az általam írt alkalmazást, annak működése közben „kézzel” belejavíthatunk.

Mi tehát a „helyes” módszer a független zéró rendszer és a minimális lefedő vonalrendszer problémájának megoldására?

Ennek a kérdésnek a megválaszolásához transzformálnunk kell a mátrixunkat páros gráffá.

Hogy a kapcsolatot megteremthessük a gráfelméleti ismeretek, valamint a probléma megoldása és algoritmizálása között, néhány fogalmat ismernünk szükséges.

A páros gráf olyan gráf, amelynek csúcsai két halmazba sorolhatóak úgy, hogy azonos halmazbeli pontok között nem halad él. Két élt függetlennek nevezünk, ha nincs közös végpontjuk. Párosításnak nevezzük a páros gráf éleinek egy olyan részhalmazát, amelyben bármely két él független. A párosítás teljes, ha a páros gráf két csúcshalmaza közül legalább az egyiknek minden pontjából indul a párosításnak éle.

Legyenek a sorok és oszlopok egy páros gráf két csúcshalmaza, és a gráf azon pontjait kössük össze éllel, melyeknek megfelelő sorok és oszlopok találkozásában nulla elem van. A gráfunknak így mindkét csúcshalmaza ugyanakkora elemszámú. Független zérók független éleket, azaz párosítást jelentenek, és viszont. A keresett független zéró-rendszer tehát ebben az ábrázolásban egy teljes párosítás. (Megjegyzendő, hogy ez egy olyan páros gráf, aminek minden pontjából indul ki él.)

Lefogó pontoknak nevezzük a páros gráfban azokat a pontokat, amelyekbe futó éleket elvágva az összes élet elvágtuk. Ez a fogalom a mátrixunk esetén azt jelenti, hogy kiválasztva a pontoknak megfelelő oszlopokat és sorokat, azok a mátrixban található összes nullát tartalmazzák, tehát lefedő vonalrendszerét alkotják a mátrixnak.

A maximális párosítás kérdésének problémáját Kőnig Dénes és Egerváry Jenő magyar matematikusok oldották meg az 1930-as években.

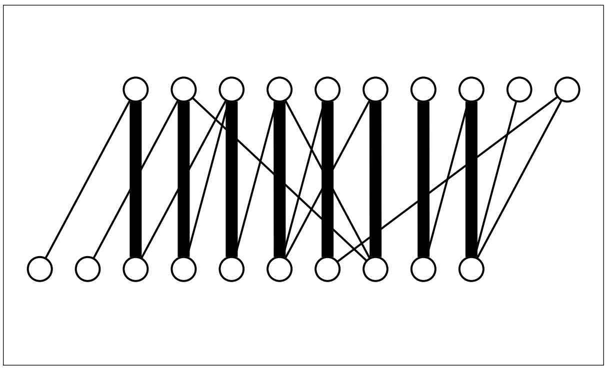
Kőnig Dénes tétele (1931): Bármely páros gráf független éleinek maximális száma egyenlő az éleket lefogó pontok minimális számával.

A tétel a mátrixra átfogalmazva azt mondja ki, hogy pontosan annyi kiválasztott nulla (független zéró) van, mint amennyi a lefedő vonalak száma. Mivel minden kiválasztott, azaz a sorok és oszlopok párosításában résztvevő nullán át kell mennie lefedő vonalnak, ebből következik, hogy minden ilyen nullán pontosan egy lefedő vonal megy át.

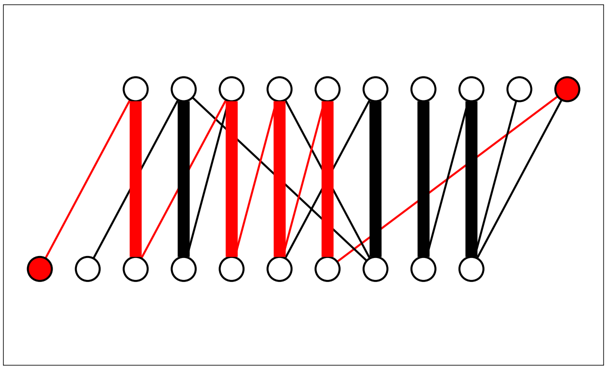
Már csak azt kellene meghatároznunk, hogy hogyan keressük a teljes párosítást, és ha nincs, akkor hogyan találjuk meg a lefedő vonalrendszert.

A Kőnig-tétel bizonyításában használjuk az alternáló erdő fogalmát. Ez az erdő olyan fákból áll, amelyek a páros gráf egyik ponthalmazából vett párosítatlan pontból indulnak, és minden páros indexű élük független, azaz részt vesz az addigi párosításban.

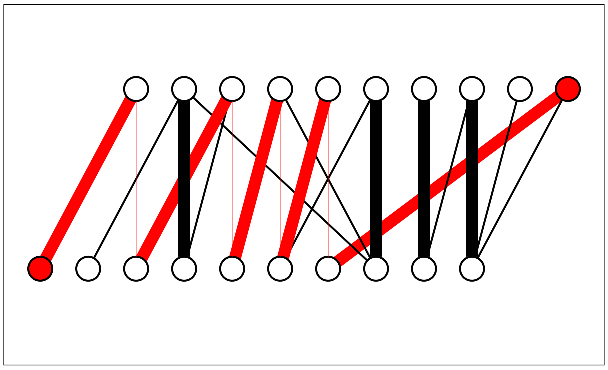
Nyilvánvaló, hogy ha egy ilyen fa részeként előálló tetszőleges útnak páratlan számú éle van, akkor a páratlan indexűeket bevéve a párosításba, és a párosakat elhagyva abból, növelhetjük a párosítások számát (lásd: 1., 2. és 3. ábra). A páratlan élszámú alternáló utak megtalálása tehát módszer a párosításunk javítására. Egy ilyen út a mátrixban úgy található meg, hogy olyan sorból indulunk ki, amelyben nincs kiválasztott nulla (a mátrixnak megfelelő páros gráfban a párosításban résztvevő él), és abban nullát találva, annak oszlopában megkeressük a kiválasztott nullát, majd a kiválasztott nulla sorában keresünk egy attól különböző nullát, és így tovább.



1. ábra. Páros gráf egy párosítása (a független élek vastagok).

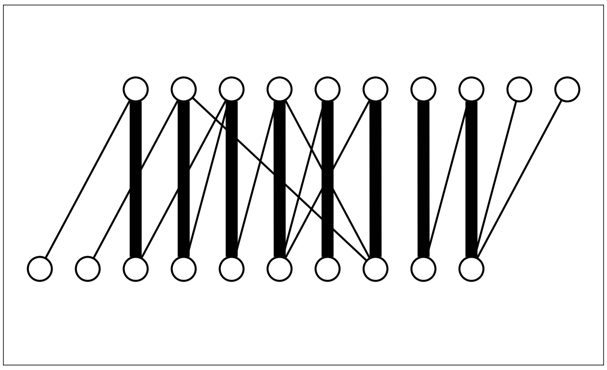


2. ábra. Javító (alternáló) út.

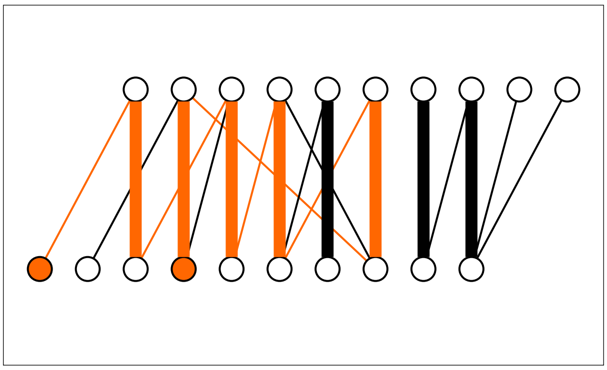


3. ábra. Javítás és így új párosítás.

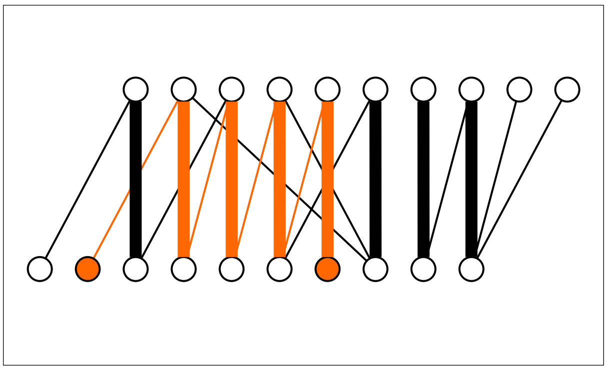
Ezeknek az utaknak a megkeresése lehetséges mind a mélységi, mind a szélességi keresés elvén. Különböző keresési algoritmusok alkalmazása esetén különböző megoldáshoz juthatunk.



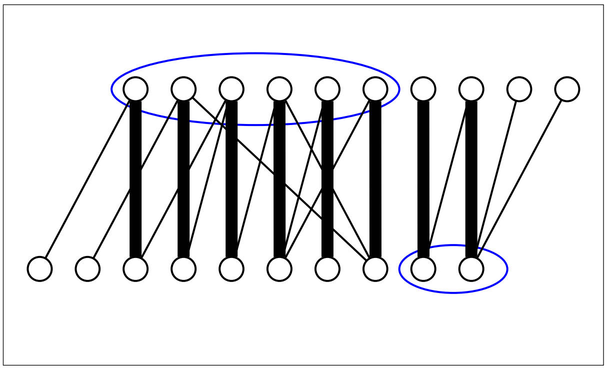
4. ábra. Páros gráf.



5. ábra. Alternáló út.



6. ábra. Alternáló út.



7. ábra. Lefogó ponthalmaz.

Amennyiben nem tudunk javítani, akkor az alternáló utak mind páros számú élből állnak, így végpontjuk a kiinduló ponthalmazban van. Ha most kiválasztjuk a másik ponthalmaz azon pontjait, amelyeket az alternáló erdő tartalmaz, akkor ezek éleit elvágva, látható, hogy a velük párosított, és az első ponthalmazban párosítatlan pontok összes élét elvágtuk, tehát az őket lefogó ponthalmazhoz jutottunk. Ha most ezekhez a pontokhoz hozzávesszük az első ponthalmaz maradék pontjait, akkor megkapjuk a teljes páros gráfot lefogó pontrendszert (lásd: 4., 5., 6. és 7. ábra.).

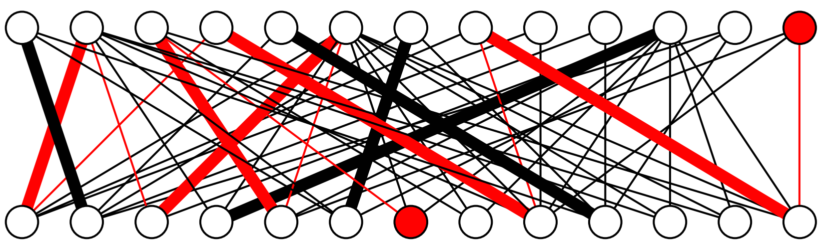
A mátrixunkban ez azt jelenti, hogy vesszük azokat az oszlopokat, amelyek részt vettek az alternáló erdőnek megfelelő bejárásban, és hozzávesszük azokat a sorokat, amelyek nem. Így ezeket lefedve eljutunk a helyes minimális lefedő vonalrendszerhez.

A fent leírt megoldás, mivel helyes és biztos elméleti ismereteken alapszik, mindig jó eredményre vezet.

Az alábbi táblázatok egy mátrixban mutatják a mélységi illetve a szélességi keresés módszere által megtalált javító utat (3. és 4. táblázat), illetve lefedő vonalrendszert (5. és 6. táblázat) egy-egy konkrét esetben. Utóbbi lehet azonos mindkét módszer esetén, hiszen bár más sorokat és oszlopokat kapcsoltunk össze a párosítási algoritmus különbözősége miatt, a mátrixban lévő nullák, azaz a megfelelő páros gráfban lévő élek azonosak, s így lehet azonos a mátrixban a lefedő vonalrendszer, illetve a páros gráfban a lefogó pontrendszer is.

Készítettem több különböző számítógépes alkalmazást is a mélységi, illetve a szélességi keresés elvét felhasználva. Az alkalmazások jól használhatók az egyes problémák megoldásán túl feladatok előkészítésére, és a hozzárendelési probléma szemléletes oktatására. Mivel az algoritmus annak bizonyos pontjain paramétertől függően megállítható, bármely konkrét feladat megoldása nyomon követhető, és ez a megértést nagymértékben segíti. A cikkben szereplő táblázatokat természetesen ezekkel az alkalmazásokkal állítottam elő.

Megjegyezném még, hogy a páros gráfok párosításának mátrixos szemléltetése sokszor előnyösebb és áttekinthetőbb, mint a pontokkal és élekkel ábrázolt gráfon. Példa erre 8. ábra, amelyben látható javító út azonos a 3. táblázatban ábrázolttal.



8. ábra. A 3. táblázat mátrixának megfelelő gráf javító útja mélységi kereséssel.

3. táblázat. (7,3) –ból kiinduló javító út, mélységi kereséssel.

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	16	0	227	0	16	0	78	0	0	16	0	78	0
2	0	24	149	123	0	24	0	123	123	0	24	0	123
3	98	0	371	43	98	0	222	43	43	98	0	222	43
4	53	0	468	211	53	0	319	211	211	53	0	319	211
5	16	16	0	222	5	0	78	0	222	5	0	78	0
6	0	0	24	144	128	24	0	123	144	128	24	0	123
7	98	98	0	366	48	0	222	43	366	48	0	222	43
8	53	53	0	463	216	0	319	211	463	216	0	319	211
9	16	0	227	0	16	0	78	0	0	16	0	78	0
10	0	24	149	123	0	24	0	123	123	0	24	0	123
11	98	0	371	43	98	0	222	43	43	98	0	222	43
12	53	0	468	211	53	0	319	211	211	53	0	319	211
13	16	16	0	222	5	0	78	0	222	5	0	78	0

4. táblázat. (7,3) –ból kiinduló javító út, szélességi kereséssel.

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	16	0	227	0	16	0	78	0	0	16	0	78	0
2	0	24	149	123	0	24	0	123	123	0	24	0	123
3	98	0	371	43	98	0	222	43	43	98	0	222	43
4	53	0	468	211	53	0	319	211	211	53	0	319	211
5	16	16	0	222	5	0	78	0	222	5	0	78	0
6	0	0	24	144	128	24	0	123	144	128	24	0	123
7	98	98	0	366	48	0	222	43	366	48	0	222	43
8	53	53	0	463	216	0	319	211	463	216	0	319	211
9	16	0	227	0	16	0	78	0	0	16	0	78	0
10	0	24	149	123	0	24	0	123	123	0	24	0	123
11	98	0	371	43	98	0	222	43	43	98	0	222	43
12	53	0	468	211	53	0	319	211	211	53	0	319	211
13	16	16	0	222	5	0	78	0	222	5	0	78	0

5. táblázat. Lefedő vonalrendszer, mélységi keresés esetén.

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	16	0	227	0	16	0	78	0	0	16	0	78	0
2	0	24	149	123	0	24	0	123	123	0	24	0	123
3	98	0	371	43	98	0	222	43	43	98	0	222	43
4	53	0	468	211	53	0	319	211	211	53	0	319	211
5	16	16	0	222	5	0	78	0	222	5	0	78	0
6	0	0	24	144	128	24	0	123	144	128	24	0	123
7	98	98	0	366	48	0	222	43	366	48	0	222	43
8	53	53	0	463	216	0	319	211	463	216	0	319	211
9	16	0	227	0	16	0	78	0	0	16	0	78	0
10	0	24	149	123	0	24	0	123	123	0	24	0	123
11	98	0	371	43	98	0	222	43	43	98	0	222	43
12	53	0	468	211	53	0	319	211	211	53	0	319	211
13	16	16	0	222	5	0	78	0	222	5	0	78	0

6. táblázat. Lefedő vonalrendszer, szélességi keresés esetén.

	1	2	3	4	5	6	7	8	9	10	11	12	13
1	16	0	227	0	16	0	78	0	0	16	0	78	0
2	0	24	149	123	0	24	0	123	123	0	24	0	123
3	98	0	371	43	98	0	222	43	43	98	0	222	43
4	53	0	468	211	53	0	319	211	211	53	0	319	211
5	16	16	0	222	5	0	78	0	222	5	0	78	0
6	0	0	24	144	128	24	0	123	144	128	24	0	123
7	98	98	0	366	48	0	222	43	366	48	0	222	43
8	53	53	0	463	216	0	319	211	463	216	0	319	211
9	16	0	227	0	16	0	78	0	0	16	0	78	0
10	0	24	149	123	0	24	0	123	123	0	24	0	123
11	98	0	371	43	98	0	222	43	43	98	0	222	43
12	53	0	468	211	53	0	319	211	211	53	0	319	211
13	16	16	0	222	5	0	78	0	222	5	0	78	0

SZERZŐ

Kiss László, főiskolai docens, BMF RKK MTI, kiss.laszlo@rkk.bmf.hu