A DISZKRÉT KIVÁLASZTÁSI MODELL BECSLÉSE COX-REGRESSZIÓVAL[1]

HAJDU OTTÓ
BME GTK

1 Bevezetés

A tanulmány az MDC (multinomial discrete choice) kiválasztási modell paramétereinek a standard pontbecslési eljárásait tekinti át egyfelől az eredmények értelmezését illetően, másfelől a modell gyakorlati alkalmazását segítendő azon esetre, mikor közvetlenül működő standard MDC programcsomag nem áll rendelkezésre.[2] A probléma klasszikus megoldása az irodalomban a túlélési modellek egyik alapmódszerének, az ún. Cox-regressziónak a felhasználásával történik, amelynek rugalmas, kontrollálható alkalmazása akkor is előnyt nyújthat, ha egyébként standard MDC program is rendelkezésre áll.[3]

Az MDC modellben egy I individuum (a döntést hozó személy) $g = 1, 2, \ldots, m_I$ számú lehetséges *alternatíva* halmazából egyet biztosan, és csak egyet kiválaszt. A választási alternatívák száma individuumonként nem szükségszerűen azonos. Nyilvánvaló, hogy a kiválasztás eredményét mind a döntéshozó individuális tulajdonságai, mind az alternatíva sajátosságai befolyásolják. A döntés kimenetét (eredményét) magyarázó változók tehát egyfelől lehetnek *individuális* jellegűek, másfelől *alternatíva-specifikusak*. Az előbbieket x, az utóbbiakat Z jelöli a tanulmányban. Bár a modell paramétereinek a becslése mindkét típus esetén a maximum likelihood (ML) elven alapul, jellegét tekintve különbözik x, vagy Z típusú magyarázó változók használata esetén. Előbb e két almodell becslésének a különbözősége kerül tárgyalásra. Ezt követően, mivel a reális döntési helyzet mind x, mind Z egyidejű figyelembe vételét (,,vegyes" modell alkalmazását) igényli, a becslési eljárást az általános, mindkét típust magában foglaló esetre is bemutatjuk.

Az alternatíva-specifikus magyarázó változók kezelése, és így a ,,vegyes" modell becslése standard módon a statisztikai szoftverek többségében közvetlenül nem érhető el. Ennek okán a tanulmány —egy illusztratív példa számszerűsítésén át— útmutatást ad arra vonatkozóan, hogy a ,,vegyes" modell adatállományát milyen struktúrában kell rögzíteni (a modellt hogyan kell paraméterezni) annak érdekében, hogy paramétereinek a becslése a standard *Cox-regresszióval* megoldhatóvá váljon.

---

[1]Beérkezett: 2005. június 21. E-mail: `ohajdu@finance.bme.hu`

[2]Például a Systat 11.0, vagy a SAS program szolgáltat közvetlenül hívható MDC modult, viszont például az SPSS 13.0 programban az eljárás csak közvetetten oldható meg.

[3]Lásd például Kuhfeld (2003), vagy a SAS program MDC és QLIM eljárásait.

2 A polichotom logit modell esete

A polichotom (multinomiális) logit modell (PL) akkor alkalmazandó, ha a döntés diszkrét kimenetét (a kiválasztás eredményét) magyarázó változó *individuális* jellegű, és az alternatívák köre mindenkire azonos: $g = 1, 2, \ldots, m$. Jelölje x_I a döntést hozó I individuum valamely jellemzőjét: példánkban az életkorát. Ekkor annak a valószínűsége, hogy a választható alternatívák közül éppen a $C \in \{1, 2, \ldots, m\}$ alternatívát választja ki az I személy:

$$P_{CI} = \frac{P_{CI}}{\sum_{g=1}^{m} P_{gI}} = \frac{P_{CI}/P_{MI}}{\sum_{g=1}^{m} P_{gI}/P_{mI}} = \frac{\text{odds}_I(C:m)}{\sum_{g=1}^{m} \text{odds}_I(g:m)} = \frac{e^{\alpha_C+\beta_C x_I}}{\sum_{g=1}^{m} e^{\alpha_g+\beta_g x_I}} ,$$

ahol az $\text{odds}_I(C:m)$ valószínűségarány annak az esélye, hogy az I személy a C alternatívát preferálja az m referencia alternatívával szemben ($\text{odds}_I(m:m) = 1$), és az $\ln(\text{odds}) = \textit{logit}$ mennyiség a magyarázó változó lineáris függvénye az $\alpha_m = \beta_m = 0$ megkötés mellett.

Ha a döntéshozó előtt csak két alternatíva áll, akkor $m = 2$ mellett az ún. *dichotom*, vagy bináris logit modellt kapjuk.

A modell szerint mind az α_g tengelymetszet, mind a β_g parciális regressziós paraméter *alternatíva-specifikus*, miközben adott individuum x_I jellemzője ugyanaz bármely alternatíva esetén. Látható, hogy m számú alternatíva mellett a paraméterek $m-1$ számú körét definiáljuk. Ha az x változó értéke egységnyit emelkedik, akkor a $C:m$ viszonylatú *odds* az e^{β_C} *odds-ratio* (OR) faktorral szorzódik.

A regressziós paramétereket kézenfekvő a maximum likelihood (ML) módszerrel becsülni. Jelölje a $C_1, C_2, \ldots, C_n$ minta $I = 1, 2, \ldots, n$ individuum megfigyelt, független döntéseit: $C_I \in \{1, 2, \ldots, m\}$. A minta maximálandó likelihood függvénye:

$$L = \prod_{I=1}^{n} \prod_{g=1}^{m} (P_{gI})^{S_{gI}} \to \max ,$$

ahol $S_{gI} = 1$, ha a g alternatívát az I személy kiválasztotta, egyébként $S_{gI} = 0$. A paraméterek referencia-függők, a valószínűségek viszont nem. A valószínűséget a

$$P_{CI} = \frac{1}{\sum_{g=1}^{m} e^{(\alpha_g-\alpha_C)+(\beta_g-\beta_C)x_I}}$$

formában írva látszik, hogy a különféle alternatívák kiválasztási valószínűségei közötti különbség adott x mellett csak a paraméterek alternatíva-specifikus jellegéből származik.

Illusztratív példánkban $I = 1, 2, \ldots, 21$ személynek az utazásuk módját illető választásait az életkorukkal magyarázzuk, tengelymetszet szerepeltetése mellett. A lehetséges három mód: *autó* (A), *repülő* (R) és *vonat* (V). A három mód közül egyet és csak egyet választ mindenki. A három mód mellett változónként rendre 2 koefficienst (két tengelymetszetet és két életkormeredekséget) becslünk úgy, hogy a *vonat* (V) a referencia alternatíva. A koefficiensek ML becsléseit az *1. tábla*, az adatokat és az eredményeket pedig a *2. tábla* közli. A táblában a „C" megnevezésű oszlop az illető választási döntését közli, az S_A, S_R és S_V indikátor jellegű oszlopok pedig a döntés statusa szerint veszik fel az 1, illetve a 0 értéket.

Például az $I = 1$ egyén, aki egyébként 32 éves, a repülős utat választotta, ezért

$$\text{odds}(R/V \mid 32) = e^{2.7212-0.05\cdot 32} = 3.068 \ ,$$

majd a repülős út kiválasztásának a valószínűsége

$$P_{R1} = \frac{3.068}{2.168 + 3.068 + 1} = 0.492 \ .$$

Ez a valószínűség bárkit, aki 32 éves, egyöntetűen jellemez! Az így számított 21 darab P_C valószínűség $L = 0.492 \cdot 0.483 \cdot \ldots \cdot 0.421$ szorzata a 2. táblában az 1. tábla koefficiensei mellett maximális, a megfelelő $-2 \ln L$ „*goodness-of-fit*" statisztika értéke pedig 42.18.[4]

Változó	Koefficiens		
	Autó	Repülő	Vonat
Tengelymetszet	3.0449	2.7212	0
Életkor	-0.0710	-0.0500	0

1. táblázat. ML becslés az életkor-koefficiensekre

Megfigyelés (I)	Életkor	C	S_A	S_R	S_V	$odds_A$	$odds_R$	$odds_V$	P_C
1	32	R	0	1	0	2.168	3.068	1	0.492
2	13	A	1	0	0	8.351	7.934	1	0.483
3	41	V	0	0	1	1.145	1.956	1	0.244
4	41	V	0	0	1	1.145	1.956	1	0.244
5	47	A	1	0	0	0.748	1.449	1	0.234
6	24	R	0	1	0	3.826	4.577	1	0.487
7	27	A	1	0	0	3.092	3.940	1	0.385
8	21	R	0	1	0	4.733	5.318	1	0.481
9	23	A	1	0	0	4.107	4.812	1	0.414
10	30	R	0	1	0	2.499	3.391	1	0.492
11	58	R	0	1	0	0.343	0.836	1	0.384
12	36	V	0	0	1	1.633	2.512	1	0.194
13	43	A	1	0	0	0.993	1.770	1	0.264
14	33	R	0	1	0	2.020	2.919	1	0.491
15	30	R	0	1	0	2.499	3.391	1	0.492
16	28	R	0	1	0	2.880	3.748	1	0.491
17	44	R	0	1	0	0.925	1.684	1	0.467
18	37	V	0	0	1	1.521	2.390	1	0.204
19	45	A	1	0	0	0.862	1.602	1	0.249
20	35	R	0	1	0	1.753	2.641	1	0.490
21	22	A	1	0	0	4.409	5.059	1	0.421

2. táblázat. Utazási mód választása adott életkor (év) mellett

[4] A maximálást a paraméterek tekintetében —az 1–5. táblák esetén— a MS Office 2003 Excel-Solver moduljával végeztük el. Ezzel pontbecslést nyertünk a paraméterekre.

A fenti modell becslése statisztikai szoftverekben standard módon, közvetlenül elérhető úgy, hogy 21 megfigyelés (case) mellett az *Életkor* a kovariáns és C az eredményváltozó.

Ha több, rendre $x_1, x_2, \ldots, x_k$ magyarázó változót tartalmaz a modell, akkor a regressziós paraméterek köre is megfelelően bővül, de a modell lényegileg változatlan marad. Például három magyarázó változót használva (*Életkor, Jövedelem, Nem*) három alternatívához a két tengelymetszet mellett még $3 * 2 = 6$ meredekséget is becsülnünk kell. Ha pedig olyan nominális magyarázó változóval bővítjük a modellt, melynek kettőnél több kimenete van (lakóhely szerint Budapest, Többi_város, Község), akkor az indikátor változók száma kettővel (B, Tv), a paraméterek száma pedig $2 * 2 = 4$-gyel emelkedik. Mint látható, a PL modell nem takarékos a paraméterekkel, és a paraméterek értelmezése a mindenkori referencia alternatíva viszonylatát igényli.

3 A feltételes logit modell esete

A feltételes (conditional) logit modell (CL) akkor alkalmazandó, ha a döntés kimenetét magyarázó változó *alternatíva-specifikus*, annak valamely tulajdonságát írja le. Jelölje Z az alternatívák egy jellemzőjét: példánkban az utazás időigényét. Megfigyelésünk most nem az individuumra, hanem az összes előforduló Z_{gI} $(I = 1, 2, \ldots, n;\ g = 1, 2, \ldots, m_I)$ utazási időre (órában mérve) irányul. A megfigyelt eseteket az individuumok n számú rétegre bontják, és adott rétegen belül egy alternatíva kiválasztásra kerül, a többi nem.

A $21 * 3 = 63$ megfigyelést (esetet) 21 rétegre bontva a *3. tábla* tartalmazza. Az egyes alternatívák időigényét adott individuum esetén az AI, RI és VI oszlopok azonosítják, de magyarázó változónk csak egy van, az *utazás időigénye*. Valamennyi *esetet* szemlélve a döntés kimenetét binárisan kódoljuk: $S_{gI} = 1$, ha a g alternatíva kiválasztásra került az I rétegben, és $S_{gI} = 0$, ha nem. Az eredményváltozó megfelelő rétegzett értéke tehát: S_{gI}. A kiválasztás előrejelzése így egy *rétegzett, dichotom* logit modell alkalmazására vezetett. E modell paramétere alternatíva-független, globális, minden megfigyelésre egyformán érvényes. E paraméterek becslése a CL modell speciális alkalmazásával valósítható meg.

A CL modell lényegét segít megvilágítani, ha előbb külön az $I = 1$ individuumot (réteget) tekintjük, aki a repülőt választotta, tehát esetében a háromelemű bináris döntési szekvencia: $\mathbf{d}_R = [0, 1, 0]$. Ennek likelihoodja a dichotom logit modell alapján:

$$L_{RI} = \frac{1}{1 + e^{\alpha + \theta Z_{AI}}} \cdot \frac{e^{\alpha + \theta Z_{RI}}}{1 + e^{\alpha + \theta Z_{RI}}} \cdot \frac{1}{1 + e^{\alpha + \theta Z_{VI}}} \,,$$

ahol α és θ *globális* paraméterek. A döntéshozó azonban választhatott volna másképpen is. Ragaszkodva ahhoz, hogy csak egy alternatívát választhat, a további lehetőségei rendre a $\mathbf{d}_A = [1, 0, 0]$ és a $\mathbf{d}_V = [0, 0, 1]$ szekvenciák, melyek likelihoodjai rendre:

$$L_{AI} = \frac{e^{\alpha+\theta Z_{AI}}}{1+e^{\alpha+\theta Z_{AI}}} \cdot \frac{1}{1+e^{\alpha+\theta Z_{RI}}} \cdot \frac{1}{1+e^{\alpha+\theta Z_{VI}}} ,$$

$$L_{VI} = \frac{1}{1+e^{\alpha+\theta Z_{AI}}} \cdot \frac{1}{1+e^{\alpha+\theta Z_{RI}}} \cdot \frac{e^{\alpha+\theta Z_{VI}}}{1+e^{\alpha+\theta Z_{VI}}} .$$

Ezek birtokában a $C \in \{A, R, V\}$ alternatíva kiválasztásának a feltételes valószínűsége az I egyén esetében a megfelelő likelihood statisztikai *megoszlása* a három likelihood összegében, általában pedig:

$$P_{CI} = \frac{L_{CI}}{\sum_{g=1}^{m_I} L_{gI}} = \frac{e^{\theta Z_{CI}}}{\sum_{g=1}^{m_I} e^{\theta Z_{gI}}} = \frac{1}{\sum_{g=1}^{m_I} e^{\theta(Z_{gI}-Z_{CI})}} .$$

Vegyük észre, hogy a likelihoodok közös nevezője és a globális tengelymetszet eliminálódik a valószínűségből, ezért szerepeltetésük fölösleges.[5] A magyarázó változó egységnyi abszolút növekményének a kiválasztási valószínűségre gyakorolt hatása a magyarázó Z változó alternatívák közötti ingadozásától függ, és konstans.

Végül a θ paraméter tekintetében maximálandó likelihood a rétegen belüli kiválasztási valószínűségek szorzata:

$$L = \prod_{I=1}^{n} \prod_{g=1}^{m_I} (P_{gI})^{S_{gI}} \to \max .$$

Az egyetlen változónkhoz tartozó θ paraméter ML becslése $\hat{\theta} = -0.26549$. Így az $I = 1$ egyén esetén a repülős út kiválasztásának a valószínűsége:

$$P_{RI} = \frac{0.303}{0.07 + 0.303 + 0.062} = 0.697 .$$

Az ily módon kalkulált 21 darab P_C valószínűség L szorzata a *3. táblában* a fenti koefficiens mellett maximális, és a $-2 \ln L$ statisztika értéke 33.629.

Ha több, rendre $Z_1, Z_2, \ldots, Z_q$ magyarázó változót tartalmaz a modell, akkor a regressziós paraméterek köre is megfelelően bővül.

[5] A feltételes logit P_{CI} valószínűségének a nevezője azért tartalmaz annyi összeadandót, ahány alternatíva van, mert az individuum csak egy alternatívát választhat ki, így az egy darab 1 összes permutációinak a száma megegyezik az alternatívák számával. Ha háromnál több alternatíva közül egynél többet is választhatunk, például kettőt, akkor az összes olyan permutációk száma melyek a szekvenciában két helyen tartalmaznak 1 értéket, már megsokszorozódik.

Réteg (I)	AI	RI	VI	S_A	S_R	S_V	$e^{\theta AI}$	$e^{\theta RI}$	$e^{\theta VI}$	P_C
1	10	4.5	10.5	0	1	0	0.070	0.303	0.062	0.697
2	5.5	4	7.5	1	0	0	0.232	0.346	0.137	0.325
3	4.5	6	5.5	0	0	1	0.303	0.203	0.232	0.314
4	3.5	2	5	0	0	1	0.395	0.588	0.265	0.212
5	1.5	4.5	4	1	0	0	0.671	0.303	0.346	0.509
6	10.5	3	10.5	0	1	0	0.062	0.451	0.062	0.786
7	7	3	9	1	0	0	0.156	0.451	0.092	0.223
8	9	3.5	9	0	1	0	0.092	0.395	0.092	0.683
9	4	5	5.5	1	0	0	0.346	0.265	0.232	0.410
10	22	4.5	22.5	0	1	0	0.003	0.303	0.003	0.982
11	7.5	5.5	10	0	1	0	0.137	0.232	0.070	0.529
12	11.5	3.5	11.5	0	0	1	0.047	0.395	0.047	0.096
13	3.5	4.5	4.5	1	0	0	0.395	0.303	0.303	0.395
14	12	3	11	0	1	0	0.041	0.451	0.054	0.826
15	18	5.5	20	0	1	0	0.008	0.232	0.005	0.946
16	23	5.5	21.5	0	1	0	0.002	0.232	0.003	0.977
17	4	3	4.5	0	1	0	0.346	0.451	0.303	0.410
18	5	2.5	7	0	0	1	0.265	0.515	0.156	0.167
19	3.5	2	7	1	0	0	0.395	0.588	0.156	0.347
20	12.5	3.5	15.5	0	1	0	0.036	0.395	0.016	0.883
21	1.5	4	2	1	0	0	0.671	0.346	0.588	0.418

3. táblázat. Utazási mód választása az utazási idő (óra) függvényében

4 A „vegyes" modell alkalmazása

A valósághű alkalmazás mind a választó individuum, mind a választandó alternatíva jegyeit figyelembe veszi. Ekkor a C alternatíva I egyén által való kiválasztásának a valószínűsége:

$$P_{CI} = \frac{e^{\alpha_C + \beta_C x_I + \theta Z_{CI}}}{\sum_{g=1}^{m} e^{\alpha_g + \beta_g x_I + \theta Z_{gI}}} \quad | \, \alpha_m = \beta_m = 0 \, .$$

A ML becsléssel nyert koefficienseket a *4. tábla* közli, melyekre az *5. táblában* foglalt 21 db P_C valószínűség szorzata maximális: a $-2 \ln L$ statisztika értéke 27.46433.

Változó	Koefficiens			
	Globális	Autó	Repülő	Vonat
Tengelymetszet		2.5007	-2.7792	0
Életkor		-0.07830	0.0169	0
Utazási idő	-0.6085			

4. táblázat. A vegyes modell ML koefficiensei

I	AI	RI	VI	Kor	C	$2.5+$ $-.078Kor+$ $-.608AI$	$-2.779+$ $.017Kor+$ $-.608RI$	$-.608VI$	P_C
1	10	4.5	10.5	32	R	−6.089	−4.975	−6.389	0.636
2	5.5	4	7.5	13	A	−1.864	−4.993	−4.564	0.900
3	4.5	6	5.5	41	V	−3.446	−5.735	−3.347	0.501
4	3.5	2	5	41	V	−2.838	−3.301	−3.042	0.333
5	1.5	4.5	4	47	A	−2.090	−4.721	−2.434	0.561
6	10.5	3	10.5	24	R	−5.766	−4.197	−6.389	0.757
7	7	3	9	27	A	−3.871	−4.147	−5.476	0.510
8	9	3.5	9	21	R	−4.619	−4.553	−5.476	0.428
9	4	5	5.5	23	A	−1.733	−5.431	−3.347	0.817
10	22	4.5	22.5	30	R	−13.233	−5.009	−13.691	0.999
11	7.5	5.5	10	58	R	−6.602	−5.143	−6.085	0.616
12	11.5	3.5	11.5	36	V	−7.314	−4.299	−6.997	0.060
13	3.5	4.5	4.5	43	A	−2.994	−4.788	−2.738	0.407
14	12	3	11	33	R	−7.384	−4.045	−6.693	0.904
15	18	5.5	20	30	R	−10.799	−5.618	−12.169	0.993
16	23	5.5	21.5	28	R	−13.685	−5.652	−13.082	0.999
17	4	3	4.5	44	R	−3.377	−3.858	−2.738	0.176
18	5	2.5	7	37	V	−3.437	−3.673	−4.259	0.197
19	3.5	2	7	45	A	−3.151	−3.234	−4.259	0.444
20	12.5	3.5	15.5	35	R	−7.844	−4.316	−9.431	0.966
21	1.5	4	2	22	A	−0.134	−4.840	−1.217	0.742

5. táblázat. Utazási mód választása az életkor (év) és az utazási idő (óra) függvényében

A vegyes modellben annak a valószínűsége, hogy az $I = 1$ személy a repülős utat választja:

$$P_{R1} = \frac{e^{-4.975}}{e^{-6.089} + e^{-4.975} + e^{-6.389}} = 0.636 \; .$$

Mint látható, a vegyes modell individuális változójának paramétere alternatíva-specifikus, míg az alternatíva-specifikus változó paramétere globális. Ez nehézséget okoz akkor, ha *szimultán* becslésükre (az alkalmazott statisztikai szoftver adottsága miatt) a polichotom logit módszer és a feltételes logit módszer csak szeparáltan hívható fel. Kézenfekvő megoldás az individuális változót is globalizálni.

5 A vegyes modell globális paraméterezése

Tekintsük az $i = 1, 2, \ldots, nm$ egyedi választási (utazási) lehetőségeket, melyeket az individuumok n rétegbe sorolnak. Definiáljuk az X *globális* változó értékeit az X_i $(i = 1, 2, \ldots, nm)$ módon, ahol i egy (g, I) párosítást képvisel. Azonosítsa továbbá a D_g globális *indikátor* változó 1 értékkel a g alternatívát, értéke egyébként 0. Így, az indikátor változók felhasználásával X hatása a kiválasztásra a $\gamma_0 + \gamma_1 X_i$ modell szerint alakul, ahol a γ globális paraméterek alternatívafüggők, az alábbiaknak megfelelően:

$$\gamma_0 = \sum_{g=1}^{m} \alpha_g D_g \; , \quad \gamma_1 = \sum_{g=1}^{m} \beta_g D_g \; , \quad (\alpha_m = \beta_m = 0) \; .$$

Így a globális X változó lineáris hatása:

$$\sum_{g=1}^{m} \alpha_g D_g + \sum_{g=1}^{m} \beta_g (D_g X_i) \ .$$

Alkossák most a **Z** változók körét egyfelől az eredetileg is Z jellegű változó, másfelől az alternatívát azonosító D_g változók, végül ezen indikátor változóknak az X változóval vett $D_g * X = D_g X$ *interakciói.* A C utazási mód kiválasztásának a valószínűsége az I individuum által (az életkort és az utazás idejét egyidejűleg figyelembe véve):

$$P_{CI} = \frac{e^{\alpha_C D_C + \beta_C (D_C X_{CI}) + \theta Z_{CI}}}{\sum_{g=1}^{m} e^{\alpha_g D_g + \beta_g (D_g X_{gI}) + \theta Z_{gI}}} \quad (\alpha_m = \beta_m = 0) \ ,$$

ahol a paraméterek becslése a feltételes likelihood maximálását igényli.

A fenti módon definiált, példabeni adatainkat a *6. tábla* írja le. E struktúrában adott individuum választási halmaza egy önálló réteget (strata) alkot, melyen belül mindegyik alternatíva egy önálló megfigyelést (sort) igényel. Az adatállományban a sorok száma $21 * 3 = 63$, a rétegek száma 21, és mindegyik réteg 3 alternatívát tartalmaz. Az S indikátor változó azt jelzi, hogy az alternatíva kiválasztásra került vagy sem. A D_A és D_R indikátor változók rendre az ,,autós” és a ,,repülős” utat azonosítják, miközben a ,,vonatutazás” a *referencia* alternatíva. (A t oszlop tartalma a következő fejezetben kerül definiálásra.)

Ha a paraméterbecsléshez feltételes maximum likelihood program nem áll rendelkezésre, akkor a probléma *túlélési* modellként való megfogalmazása nyújt megfelelő eredményt. Ennek során minden kiválasztást mint bekövetkezett eseményt, a ki nem választásokat pedig mint később bekövetkezendő eseményeket kezeljük, majd az ,,eseményig” tartó időtartam alakulását modellezzük magyarázó változók ismerete mellett.

Ennek egyik eszköze a *Cox*-regresszió, mely speciális körülmények között a CL modell megoldását nyújtja (Kuhfeld (2003)).

I	Mód	Idő	Kor	S	t	D_A	D_R	$D_A * Kor$	$D_R * Kor$
1	A	10.0	32	0	2	1	0	32	0
1	R	4.5	32	1	1	0	1	0	32
1	V	10.5	32	0	2	0	0	0	0
2	A	5.5	13	1	1	1	0	13	0
2	R	4.0	13	0	2	0	1	0	13
2	V	7.5	13	0	2	0	0	0	0
3	A	4.5	41	0	2	1	0	41	0
3	R	6.0	41	0	2	0	1	0	41
3	V	5.5	41	1	1	0	0	0	0
4	A	3.5	41	0	2	1	0	41	0
4	R	2.0	41	0	2	0	1	0	41
4	V	5.0	41	1	1	0	0	0	0
5	A	1.5	47	1	1	1	0	47	0
5	R	4.5	47	0	2	0	1	0	47
5	V	4.0	47	0	2	0	0	0	0

6. táblázat. Vegyes modell interakciókkal

I	Mód	Idő	Kor	S	t	D_A	D_R	$D_A * Kor$	$D_R * Kor$
6	A	10.5	24	0	2	1	0	24	0
6	R	3.0	24	1	1	0	1	0	24
6	V	10.5	24	0	2	0	0	0	0
7	A	7.0	27	1	1	1	0	27	0
7	R	3.0	27	0	2	0	1	0	27
7	V	9.0	27	0	2	0	0	0	0
8	A	9.0	21	0	2	1	0	21	0
8	R	3.5	21	1	1	0	1	0	21
8	V	9.0	21	0	2	0	0	0	0
9	A	4.0	23	1	1	1	0	23	0
9	R	5.0	23	0	2	0	1	0	23
9	V	5.5	23	0	2	0	0	0	0
10	A	22.0	30	0	2	1	0	30	0
10	R	4.5	30	1	1	0	1	0	30
10	V	22.5	30	0	2	0	0	0	0
11	A	7.5	58	0	2	1	0	58	0
11	R	5.5	58	1	1	0	1	0	58
11	V	10.0	58	0	2	0	0	0	0
12	A	11.5	36	0	2	1	0	36	0
12	R	3.5	36	0	2	0	1	0	36
12	V	11.5	36	1	1	0	0	0	0
13	A	3.5	43	1	1	1	0	43	0
13	R	4.5	43	0	2	0	1	0	43
13	V	4.5	43	0	2	0	0	0	0
14	A	12.0	33	0	2	1	0	33	0
14	R	3.0	33	1	1	0	1	0	33
14	V	11.0	33	0	2	0	0	0	0
15	A	18.0	30	0	2	1	0	30	0
15	R	5.5	30	1	1	0	1	0	30
15	V	20.0	30	0	2	0	0	0	0
16	A	23.0	28	0	2	1	0	28	0
16	R	5.5	28	1	1	0	1	0	28
16	V	21.5	28	0	2	0	0	0	0
17	A	4.0	44	0	2	1	0	44	0
17	R	3.0	44	1	1	0	1	0	44
17	V	4.5	44	0	2	0	0	0	0
18	A	5.0	37	0	2	1	0	37	0
18	R	2.5	37	0	2	0	1	0	37
18	V	7.0	37	1	1	0	0	0	0
19	A	3.5	45	1	1	1	0	45	0
19	R	2.0	45	0	2	0	1	0	45
19	V	7.0	45	0	2	0	0	0	0
20	A	12.5	35	0	2	1	0	35	0
20	R	3.5	35	1	1	0	1	0	35
20	V	15.5	35	0	2	0	0	0	0
21	A	1.5	22	1	1	1	0	22	0
21	R	4.0	22	0	2	0	1	0	22
21	V	2.0	22	0	2	0	0	0	0

6. táblázat. Vegyes modell interakciókkal (folyt.)

6 A Cox-regresszió: „proportional hazards”

Jelölje t a vizsgált „*esemény*” bekövetkezéséig a megfigyelés (folyamat) kezdetétől eltelt idő hosszát: „*event time*”. E periódus változó időtartamát a modell szerint a Z_1, Z_2, ... magyarázó változók szintjei indokolják, és t_j a megfigyelt időtartamok *növekvő* rangsorában a j-edik, miközben f_j annak a gyakorisága, hogy t_j eltelt idő mellett a vizsgált eseményt hányszor észleltük:[6]

$$t_{1(f_1)} < t_{2(f_2)} < \ldots < t_{j(f_j)} < \ldots t_{k(f_k)} \ .$$

Ha egy individuum —akinél a folyamat már elindult, de— valami ok folytán kikerül a megfigyelési körből az *esemény bekövetkezése nélkül*, akkor az illető megfigyelést *cenzorált* (censored) esetként kezeljük. Jelölje továbbá R_j mindazon indexek által alkotott kockázati csoportot, akik közvetlen a t_j időt megelőzőleg ki vannak téve az esemény kockázatának. E kockázati körben az „event time" *legalább* t_j, és a t_j mellett cenzorált esetek tagjai e kockázati csoportnak. Ekkor annak feltételes valószínűsége, hogy valamely individuum megéli a t_j időt, de utána az esemény rögtön bekövetkezik, nem más, mint a „*hazard-ratio*" megoszlása:

$$P_Z = \frac{e^{\beta^T Z}}{\sum_{l \in R_j} e^{\beta^T Z_l}} \ .$$

E valószínűségek szorzata valamennyi t időre (súlyozottan felírva) a Breslow-féle likelihood függvényt adja:

$$L(\beta) = \prod_{j=1}^{k} \frac{e^{\beta^T Z_j^+}}{\left(\sum_{l \in R_j} e^{\beta^T Z_l}\right)^{f_j}} \to \max \ ,$$

ahol Z_j^+ a megfelelő magyarázó változó összegzését jelöli mindazokra, akiknél az esemény a t_j időpontban bekövetkezett. (A súlyozatlan eset, mikor $f_j = 1$ minden j-re, speciálisan a Cox-féle parciális likelihood függvényt eredményezi.) Ha a minta $I = 1, 2, \ldots, n$ rétegre van bontva, akkor a Breslow-likelihood egyszerűen a rétegen belüli likelihoodok szorzata:

$$L(\beta) = \prod_{I=1}^{n} L_I(\beta) \ .$$

Ahhoz, hogy a Breslow-likelihood a feltételes logit likelihoodjával ekvivalens legyen, az alábbiak szükségesek:

1. A megfigyeléseket az individuumok szerinti rétegekre (strata) bontjuk,
2. a kiválasztott C alternatívához a status változóban $S = 1$ (event) értéket, a ki nem választott alternatívákhoz pedig az $S = 0$ (censored) értéket rendeljük,
3. a kiválasztott C alternatívához mindig $t = 1$ „event time", a ki nem választott alternatívákhoz pedig egy nagyobb (későbbi), de egyöntetűen $t = 2$ „censored time" értéket rendelünk,

[6] A túlélési modell néhány alapfogalmát lásd a Függelékben!

4. mivel a diszkrét kiválasztási modellben a „t" *változó* adott értéke szükségszerűen többször fordul elő, ezért ha e kötések (ties) kezelésére az alkalmazott szoftverben opcionálisan más típus is választható (lásd SAS), akkor kifejezetten a Breslow-likelihood választandó.

A 6. tábla adataira a Cox-regressziót alkalmazva visszakapjuk a korábban már megismert (Excel-Solver) megoldásokat, az alábbiak szerint:

1. rétegképző „strata" változó: „*Individuum*",
2. a „status" változó: S,
3. az „event time" változó: t,
4. a kovariánsok: *Utazási idő*, D_A, D_R, $D_A * Kor$, $D_R * Kor$.

A (B) pontbecslések mellett aszimptotikus standard hibákat (SE), parciális Wald-statisztikákat, szignifikancia-értékeket, $\exp(B)$ „*hazard-ratio*" értékeket és 95%-os konfidencia intervallumokat is nyerünk. Az SPSS programmal kapott eredményeket a *7. tábla* közli.

Eszerint 5 százalékos szignifikancia szinten csak az utazás időtartama hat szignifikánsan a választásra. Továbbmenve, ha az utazás 1 órával tovább tart, akkor a kérdéses utazási mód kiválasztásának az esélye $100 \cdot (1 - 0.544) = 45.6$ százalékkal csökken. A többi paraméter tesztelése és az $\exp(B)$ „*hazard-ratio*" értelmezése analóg.

Az előzőekben modellenként rendre közöltük a Likelihood Ratio típusú *goodness-of-fit* statisztikák $-2 \ln L$ értékeit. A háromféle modell úgy ítélendő meg, hogy a tökéletesen illeszkedő *szaturált modell* esetén $-2 \ln L = 0$, míg a kovariánst nem tartalmazó „intercept only" ún. *null-modell* esetén $-2 \ln L = 46.142$. E határok között az egyes változók lépésenkénti szelektálására is lehetőség nyílik, melynek eredményeit a *8. tábla* közli.

Változó	B	SE	Wald	df	p-value	$\exp(B)$	Lower	Upper
Utazási idő	-.608	.271	5.031	1	.025	.544	.320	.926
D_A	2.501	2.396	1.089	1	.297	12.191	.111	1334.724
D_R	-2.779	3.529	.620	1	.431	.062	.000	62.686
$D_A * Kor$	-.078	.063	1.527	1	.217	.925	.817	1.047
$D_R * Kor$	.017	.074	.052	1	.820	1.017	.879	1.177

7. táblázat. A vegyes modell paraméterbecslése a Cox-regresszióból

Bevont változó	$-2 \ln L$	$Chi2$	df	p-value	$Chi2$_változás	df	p-value
1.: Utazási idő	33.629	11.988	1	.001	12.513	1	0.000
2.: D_R	30.284	13.522	2	.001	3.345	1	0.067
3.: $D_R * Kor$	29.266	13.940	3	.003	1.018	1	0.313
4.: $D_A * Kor$	28.739	13.966	4	.007	0.527	1	0.468
5.: D_A	27.464	15.361	5	.009	1.274	1	0.259

8. táblázat. A likelihood-arány javulása változóról változóra

Első lépésben csak az *Utazási idő*, az utolsó lépésben pedig mind az öt magyarázó változó a modellben szerepel. A *null*-modelltől való eltávolodást mérő Chi2 statisztika még a legbővebb modellt is szignifikánsnak ítéli 1%-os szinten, bár a *df* szabadsági fok a modell komplexitásának növekedésével gyorsabban nőtt, mint ahogy a $-2\ln L$ célfüggvény csökkent. A Chi2 lépésenkénti változását tesztelve látszik, hogy az utolsó három lépésben bevont tényező modellből való kihagyása megfontolandó.

7 Függetlenség az irreleváns alternatíváktól

Az eddigi modellek mindegyike azon a feltevésen alapult, hogy az alternatívák választása független egymástól: „Independence from Irrelevant Alternatives" (IIA). Ez alatt az értendő, hogy adott megfigyelésre bármely két alternatíva kiválasztási valószínűségének az egymáshoz való OR (odds-ratio) aránya független bármely más alternatívától. E feltevés lehet helytálló, lehet irreális, viszont fenntartása vagy elvetése statisztikai tesztet igényel.

Esetünkben az utazási *módok* és az utazási *idők* kölcsönhatásai (interakciói) valamint az utazási *módok egymás közti* kapcsolatainak az utazási időre gyakorolt hatása vizsgálható.

Az alternatívák és időigényeik interakcióit megfogalmazó modell

$$\sum_{g=1}^{m-1} \alpha_g D_g + \sum_{g=1}^{m} \theta_g D_g Z \ ,$$

melyet tovább bővítve alternatívaközi interakciók hozzáadásával

$$\sum_{g=1}^{m-1} \alpha_g D_g + \sum_{g=1}^{m} \theta_g D_g Z + \sum_{g=2}^{m} \delta_g D_g Z_{g-1} + \delta_1 D_1 Z_m$$

adódik. Láthatóan az alternatívák közötti kapcsolatok tesztelését most a szomszédos alternatívák vizsgálatára egyszerűsítettük. A két egymásba ágyazott modell közötti választás a paraméterek egy csoportjára vonatkozó hipotézis tesztelését igényli:

$$\delta_1 = \delta_2 = \ldots = \delta_m = 0 \ .$$

Az újonnan bevezetett magyarázó változókat —a korábban definiált változókkal együtt— a *9. tábla* tartalmazza. Például $D_R * AI$ a repülős alternatíva indikátor változójának és az autóval való utazási időnek a szorzata (repülőn utazva tart addig az út, mint egyébként autón). Az eredményeket a tágabb, meg nem szorított modellre a *10. tábla*, a szűkített modellre pedig a *11. tábla* tartalmazza.

A két modell különbsége a $-2\ln L$ statisztika tekintetében $27.153-24.781 = 2.372$, mely $df = 3$ szabadsági fok mellett nem szignifikáns (a két modell 3 paraméterben különbözik). A három alternatívaközi hatással történő bővítés tehát nem javítja jelentősen a likelihood kritériumot, így az egyéb alternatíváktól való függetlenség hipotézise jelen minta esetén fenntartható.

Felhívjuk a figyelmet végül, hogy a Cox-regresszió (Breslow-likelihood) alkalmazása a kiválasztónak megengedi, hogy választási halmazából egyidejűleg ne csak egy, hanem több alternatívát is kiválasszon: adott individuum mellett ennek megfelelően jelenik meg többször a *status* változó $S = 1$ értékkel, $t = 1$ „event time” érték mellett.

I	Mód	UI	S	t	D_A	D_R	D_V	D_AUI	D_RUI	D_VUI	D_RAI	D_VRI	D_AVI
1	A	10.0	N	2	1	0	0	10.0	.0	.0	.0	.0	10.5
1	R	4.5	I	1	0	1	0	.0	4.5	.0	10.0	.0	.0
1	V	10.5	N	2	0	0	1	.0	.0	10.5	.0	4.5	.0
2	A	5.5	I	1	1	0	0	5.5	.0	.0	.0	.0	7.5
2	R	4.0	N	2	0	1	0	.0	4.0	.0	5.5	.0	.0
2	V	7.5	N	2	0	0	1	.0	.0	7.5	.0	4.0	.0
3	A	4.5	N	2	1	0	0	4.5	.0	.0	.0	.0	5.5
3	R	6.0	N	2	0	1	0	.0	6.0	.0	4.5	.0	.0
3	V	5.5	I	1	0	0	1	.0	.0	5.5	.0	6.0	.0
4	A	3.5	N	2	1	0	0	3.5	.0	.0	.0	.0	5.0
4	R	2.0	N	2	0	1	0	.0	2.0	.0	3.5	.0	.0
4	V	5.0	I	1	0	0	1	.0	.0	5.0	.0	2.0	.0
5	A	1.5	I	1	1	0	0	1.5	.0	.0	.0	.0	4.0
5	R	4.5	N	2	0	1	0	.0	4.5	.0	1.5	.0	.0
5	V	4.0	N	2	0	0	1	.0	.0	4.0	.0	4.5	.0
6	A	10.5	N	2	1	0	0	10.5	.0	.0	.0	.0	10.5
6	R	3.0	I	1	0	1	0	.0	3.0	.0	10.5	.0	.0
6	V	10.5	N	2	0	0	1	.0	.0	10.5	.0	3.0	.0
7	A	7.0	I	1	1	0	0	7.0	.0	.0	.0	.0	9.0
7	R	3.0	N	2	0	1	0	.0	3.0	.0	7.0	.0	.0
7	V	9.0	N	2	0	0	1	.0	.0	9.0	.0	3.0	.0
8	A	9.0	N	2	1	0	0	9.0	.0	.0	.0	.0	9.0
8	R	3.5	I	1	0	1	0	.0	3.5	.0	9.0	.0	.0
8	V	9.0	N	2	0	0	1	.0	.0	9.0	.0	3.5	.0
9	A	4.0	I	1	1	0	0	4.0	.0	.0	.0	.0	5.5
9	R	5.0	N	2	0	1	0	.0	5.0	.0	4.0	.0	.0
9	V	5.5	N	2	0	0	1	.0	.0	5.5	.0	5.0	.0
10	A	22.0	N	2	1	0	0	22.0	.0	.0	.0	.0	22.5
10	R	4.5	I	1	0	1	0	.0	4.5	.0	22.0	.0	.0
10	V	22.5	N	2	0	0	1	.0	.0	22.5	.0	4.5	.0
11	A	7.5	N	2	1	0	0	7.5	.0	.0	.0	.0	10.0
11	R	5.5	I	1	0	1	0	.0	5.5	.0	7.5	.0	.0
11	V	10.0	N	2	0	0	1	.0	.0	10.0	.0	5.5	.0
12	A	11.5	N	2	1	0	0	11.5	.0	.0	.0	.0	11.5
12	R	3.5	N	2	0	1	0	.0	3.5	.0	11.5	.0	.0
12	V	11.5	I	1	0	0	1	.0	.0	11.5	.0	3.5	.0
13	A	3.5	I	1	1	0	0	3.5	.0	.0	.0	.0	4.5
13	R	4.5	N	2	0	1	0	.0	4.5	.0	3.5	.0	.0
13	V	4.5	N	2	0	0	1	.0	.0	4.5	.0	4.5	.0
14	A	12.0	N	2	1	0	0	12.0	.0	.0	.0	.0	11.0
14	R	3.0	I	1	0	1	0	.0	3.0	.0	12.0	.0	.0
14	V	11.0	N	2	0	0	1	.0	.0	11.0	.0	3.0	.0

9. táblázat. Irreleváns alternatívák függetlenségvizsgálata

I	Mód	UI	S	t	D_A	D_R	D_V	D_AUI	D_RUI	D_VUI	D_RAI	D_VRI	D_AVI
15	A	18.0	N	2	1	0	0	18.0	.0	.0	.0	.0	20.0
15	R	5.5	I	1	0	1	0	.0	5.5	.0	18.0	.0	.0
15	V	20.0	N	2	0	0	1	.0	.0	20.0	.0	5.5	.0
16	A	23.0	N	2	1	0	0	23.0	.0	.0	.0	.0	21.5
16	R	5.5	I	1	0	1	0	.0	5.5	.0	23.0	.0	.0
16	V	21.5	N	2	0	0	1	.0	.0	21.5	.0	5.5	.0
17	A	4.0	N	2	1	0	0	4.0	.0	.0	.0	.0	4.5
17	R	3.0	I	1	0	1	0	.0	3.0	.0	4.0	.0	.0
17	V	4.5	N	2	0	0	1	.0	.0	4.5	.0	3.0	.0
18	A	5.0	N	2	1	0	0	5.0	.0	.0	.0	.0	7.0
18	R	2.5	N	2	0	1	0	.0	2.5	.0	5.0	.0	.0
18	V	7.0	I	1	0	0	1	.0	.0	7.0	.0	2.5	.0
19	A	3.5	I	1	1	0	0	3.5	.0	.0	.0	.0	7.0
19	R	2.0	N	2	0	1	0	.0	2.0	.0	3.5	.0	.0
19	V	7.0	N	2	0	0	1	.0	.0	7.0	.0	2.0	.0
20	A	12.5	N	2	1	0	0	12.5	.0	.0	.0	.0	15.5
20	R	3.5	I	1	0	1	0	.0	3.5	.0	12.5	.0	.0
20	V	15.5	N	2	0	0	1	.0	.0	15.5	.0	3.5	.0
21	A	1.5	I	1	1	0	0	1.5	.0	.0	.0	.0	2.0
21	R	4.0	N	2	0	1	0	.0	4.0	.0	1.5	.0	.0
21	V	2.0	N	2	0	0	1	.0	.0	2.0	.0	4.0	.0

9. táblázat. Irreleváns alternatívák függetlenségvizsgálata (folyt.)

Változó	B	SE	Wald	*df*	p-value	exp(B)
D_A	-0.738	3.059	0.058	1	0.809	0.478
D_R	-3.624	3.480	1.084	1	0.298	0.027
$D_A * UI$	-2.234	1.899	1.384	1	0.239	0.107
$D_R * UI$	-0.101	0.686	0.022	1	0.883	0.904
$D_V * UI$	0.098	0.701	0.019	1	0.889	1.103
$D_R * AI$	0.445	0.686	0.421	1	0.517	1.560
$D_V * RI$	-0.532	0.635	0.703	1	0.402	0.587
$D_A * VI$	1.663	1.512	1.210	1	0.271	5.275

-2 Log Likelihood=24.781

10. táblázat. Az alternatívák kereszthatásai

Változó	B	SE	Wald	*df*	p-value	exp(B)
D_A	1.716	1.805	0.904	1	0.342	5.561
D_R	-3.601	3.306	1.187	1	0.276	0.027
$D_A * UI$	-0.795	0.363	4.795	1	0.029	0.451
$D_R * UI$	0.122	0.590	0.043	1	0.837	1.129
$D_V * UI$	-0.422	0.257	2.687	1	0.101	0.656

-2 Log Likelihood=27.153

11. táblázat. Redukált modell alternatíva-közi interakciók nélkül

8 Függelék

A túlélési „survival" függvény (t, T: idő)

$$S(t) = \Pr(T \geq t) = 1 - F(t) \ ,$$

a „hazard rate" függvény:

$$h(t) = \lim_{\Delta t \to 0} \frac{\Pr(t \le T \le t + \Delta t \mid T \ge t)}{\Delta t} = \frac{f(t)}{S(t)} \,,$$

Cox-„proportional hazards":

$$h(t \mid x) = \frac{f(t \mid x)}{S(t \mid x)} = h_0(t) e^{x^T \beta} \,,$$

ahol $h_0(t)$ a „base-line hazard", és a rétegzett „hazard" függvény:

$$h_g(t \mid x) = h_{0g}(t) e^{x^T \beta} \,.$$

9 Összefoglalás

A tanulmány a diszkrét kiválasztási modell paraméterbecslésének a mikéntjét mutatja be illusztratív példán attól függően, hogy csak az individuum, csak az alternatívák, vagy mindkettő jegyeire egyidejűleg áll rendelkezésre információ. Az individuális változó megfelelő értékét valamennyi döntéshez hozzárendelve, e globalizált változó paramétere közös modellben becsülhető az alternatíva-specifikus változó egyébként is globális paraméterével. Az eljárás feltételes likelihood maximálását igényli, mely a standard Cox-regresszióra visszavezethető. A tanulmány ennek előnyeire irányítja a figyelmet. A Cox-regresszió alkalmazása révén mintavételi következtetésekre, modellszelekcióra is lehetőség nyílik, továbbá vizsgálható az IIA (independence from irrelevant alternatives) hipotézis, az alternatívákhoz kötődő interakciók paramétereinek szeparált, vagy csoportos tesztelésével.

Irodalom

1. Agresti, A. (2002) *Categorical Data Analysis.* Second Edition, Wiley, New York
2. Greene, W. H. (2003) *Econometric Analysis.* Prantice Hall
3. Guadagni, P. M., Little, J. D. C. (1983) A Logit Model of Brand Choice Calibrated on Scanner Data, *Marketing Science*, Volume 2, Issue 3, 203–238
4. Hajdu, O. (2003) *Többváltozós statisztikai számítások*, KSH, Budapest
5. Hajdu, O. (2004) A csődesemény logit regressziójának kismintás problémái, *Statisztikai Szemle*, 82. évf. 4. sz. 390–422
6. Kleinbaum, D. G., Klein, M. (2002) *Logistic Regression.* Second Edition, Springer, New York
7. Kuhfeld, W. F. (2003) *Marketing Research Methods in SAS.* SAS Institue Inc., NC, USA
8. Norusis, M. J. (2006) *SPSS 13.0.* Advanced Statistical Procedures Companion, Prentice Hall

9. The MDC procedure: http://support.sas.com/rnd/app/papers/mdc.pdf
10. The QLIM procedure: http://support.sas.com/rnd/app/papers/qlim.pdf

PARAMETER ESTIMATION FOR MULTINOMIAL DISCRETE CHOICE MODEL USING COX-REGRESSION

Considering the so-called ,,multinomial discrete choice" model the focus of this paper is on the estimation problem of the parameters. Especially, the basic question arises how to carry out the point and interval estimation of the parameters when the model is mixed i.e. includes both individual and choice-specific explanatory variables while a standard MDC computer program is not available for use. The basic idea behind the solution is the use of the Cox-proportional hazards method of survival analysis which is available in any standard statistical package and provided a data structure satisfying certain special requirements it yields the MDC solutions desired. The paper describes the features of the data set to be analysed.

SZEMÉLYES VÉLETLENSZÁM GENERÁLÁS[1]

FERENCZI ZOLTÁN – NEMETZ TIBOR
Győri Széchenyi István Egyetem – MTA Rényi Alfréd Mat. Int.

Digitális dokumentumok, szerződések aláírásához az aláíró személyétől függő, általa generált véletlen véletlenszámokat célszerű használni. Véletlen számok generálásának erről az oldaláról számolunk be a jelen munkában.

Hagyományos és digitális dokumentum

A közelmúlt tinta-papír, írógép-papír világában törvény szabályozta az okmányok formáját és tartalmát. Törvényi előírás határozta meg, hogy mikor szabályos egy aláírás, egy cégszerű aláírás, amelyet közjegyző hitelesíthet a törvény számára. Az elektronikus hírközlés világában ugyanilyen törvényi szabályozásra került sor. A hagyományos papír- alapú világban az aláírás azonosításának, hitelesítésének több ,,kelléke" is volt, így maga az aláírás, pecsét, tanúk, illetve a közjegyző. Az aláírás célja a hitelesítés. Az, hogy igazolja: *a jogügylet meghatározott tartalommal, adott helyen és időben létrejött.* Az aláírás tehát azonosítható, és egyetlen személyhez kapcsolódik. A közjegyző feladata a hitelesítés, illetve hiteles példány őrzése és arról másolat kiadása.

Ugyanez a feladat az elektronikus társadalomban, az elektronikus üzletvitel világában kihívást jelent: *az értékesítési folyamat elektronikus eszközök, internet segítségével zajlik,* ami számos megoldandó kérdést vet fel, kezdve a személyes adatok védelmétől a tartalom biztonságáig, a hozzáférési jogokig, és persze a ,,digitális közjegyző" által hitelesített elektronikus aláírásig.

Elvárásunk egy digitális dokumentummal, okirattal szemben az, hogy akkor legyen érvényes, ha teljesül, hogy

- Szerzője és tartalma letagadhatatlan
- Megállapítható, hogy mikor készült (keltezés-igazolás).
- Sem szerzője, sem más nem változtathatja meg (sértetlenség).
- Jogtalan hozzáférő ne tudja elolvasni (bizalmasság).
- Meg lehessen győződni arról, hogy az készítette, akinek vallja magát (hitelesség).

Megoldás: sűrítmény (digitális lenyomat) készítés.

[1]Beérkezett: 2005. szeptember 18. E-mail: `z.ferenczi@invitel.hu`.

Sűrítmény (digitális lenyomat)

A digitális hírközlés során többnyire hosszabb dokumentumok cserélnek gazdát. Ezek egy részében nem követelmény, hogy azok tartalma titokban maradjon. Sokkal gyakrabban és erősebben jelenik meg az a követelmény, hogy senki ne piszkálhasson bele a dokumentumba, vagy ha belenyúlnak, akkor az legyen gyorsan felismerhető. Erre a célra hosszú dokumentumokból rövid lenyomatokat állítanak elő, amelyek nagyon erősen függnek az eredeti dokumentumoktól. Ezek a rövid stringek olyanok, hogy gyakorlatilag lehetetlen hozzájuk olyan új dokumentumot konstruálni, amelyiknek ugyanez a lenyomata. Ily módon lehetetlen két olyan dokumentumot konstruálni, amelyeknek azonos a lenyomata. Ha a dokumentumban legalább egy bitet megváltoztatunk, akkor a megfelelő lenyomatok sok bitben fognak különbözni. A lenyomatokat úgy definiálják, hogy előállításuk könnyű feladat legyen. Azt a leképezést, amely egy ilyen leképezést előállít, *hash függvény*nek nevezik. A hash függvény lényegében véve egy olyan transzformáció, amely egy tetszőleges hosszú szöveg digitális ujjlenyomatát készíti el. Az ujjlenyomat fix hosszú bitsorozat, amely jellemző az adott szövegre, és amely *digitális aláírás protokoll* szerves alkotórésze.

A hash függvényeket eredetileg adattárolásra és adatbankokban való keresésre dolgozták ki. Lényegében az ott elért eredményeket használják fel a kriptográfiában is, ahol azonban további feltételekre is figyelemmel kell lenni.

A hash függvénnyel szembeni elvárásokat a következő követelmények összegezik:

1. Gyakorlatilag lehetetlen egy adott outputhoz olyan dokumentumot konstruálni, amelyikhez a hash függvény ugyanazt az outputot rendeli.
2. Gyakorlatilag lehetetlen 2 olyan dokumentumot konstruálni, amelyek azonos hash értéket eredményeznek.
3. Ha legalább egy bitet egy dokumentumban megváltoztatunk, akkor a megfelelő hash értékek több bitben különböznek.
4. Elegendő hosszúságú bementi dokumentum esetén a *sűrítmény véletlenszerűen* viselkedik. 5-6 darab 32 bites szóból álló bemenet már „elegendően" hosszúnak tekinthető.

A 4. tulajdonság, a *véletlenszerűség* külön kommentárt igényel. Az irodalomban nem található „szép" hivatkozás arra, hogy ez a tulajdonság valóban teljesül. A konkrét alkalmazáshoz arra van szükség, hogy a sűrítmény *bármilyen eloszlású* bemeneti string esetén is egyenletes eloszlású stringnek legyen tekinthető. Igazából ennél kevesebb is elegendő, elég úgynevezett felcserélhető valószínűségi változókból álló bemeneti változókra igazolni (tehát egy adott sorozatnak ugyanakkora a valószínűsége, mint a sorrend bármely permutálásával adódó sorozatoknak). E mögött az állítás mögött egy határértéktétel áll: Ha (majdnem) független, (majdnem) azonos eloszlású valószínűségi változó sorozatot nézünk egy modulusban, és egy blokkon belüli valószínűségi változókat modulo aritmetikában összeadunk, akkor enyhe (mindig teljesülő) feltételek mellett az összeg határértékben (a blokkhossz növekedése esetén) egyenletes eloszlású lesz.

Véletlenszámok generálása

A kriptográfiában véletlenszám alatt heurisztikusan a következőt szokás érteni. Valamilyen k egész szám mellett tekintsük a $\{0, 1, \ldots, k-1\}$ értékkészletet. Tekintsük ebbe a halmazba tartozó számok olyan sorozatát, amelyeknek mindegyike lényegében ugyanannyiszor fordul elő a sorozatban, mégpedig bármelyik helyen bármelyik szám egyenlő eséllyel fordul elő (*egyenletesség*), függetlenül attól, hogy mi volt előtte, vagy mik jönnek utána (*függetlenség*). Az ilyen sorozatokat véletlenszámok sorozatának nevezik. Az értékkészlet k számosságának szokásos megválasztása $k = 2$, amikor *bináris véletlenszámokról* beszélünk. Ezek központi jelentőségét az információelméletben figyelhetjük meg. Ugyancsak gyakori a $k = 10$ választás, amikor *decimális véletlenszámokról* beszélünk. A $\{0, 1, \ldots, k-1\}$ értékkészlet helyett egy nyelv ábécéjét is lehet értékkészletnek tekinteni. Ekkor is véletlenszámokról beszélnek.

Véletlenszámok manuális előállításához kockát, pénzérmét, kártyát, sorsolást szoktak használni. Ez azonban egy rendkívül lassú folyamat. Ezért keresnek más lehetőségeket is. A 20. század elején a statisztikai következtetésekhez a véletlenszámokat manuálisan állították elő és táblázatolták. Az első „nagy” táblázatot az angol Tippet készítette 1927-ben: 40 ezer véletlen számjegyet állított elő népszámlálási adatokból.

Véletlenszámok fizikai generálása

Kriptográfiai kulcsok generálásakor a pszeudó véletlen számgenerátorok előnyei rendkívüli hátrányokká válnak. Ilyenkor különösen fontos, hogy

- az elkészült sorozat ne legyen reprodukálható
- „információtartalma” megegyezzen a méretével.

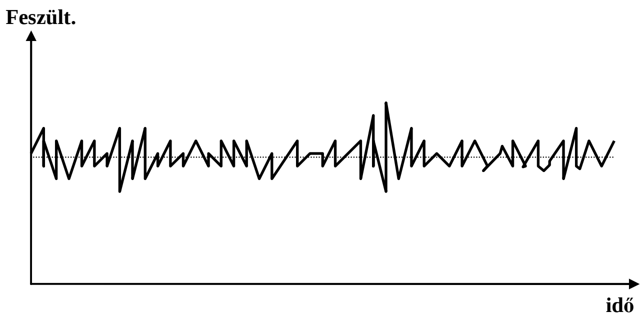
Fentiek miatt pusztán fizikai véletlenszám generátorokat használnak a rejtjelkulcsok készítésére. *A fizikai véletlenszám generátorok alapjául véletlen jeleket adó természeti jelenség szolgál.*

A gyakorlatban legtöbbször valamilyen rádiótechnikai eszköz a jelforrás. Segítségével építhető olyan számítógépes bővítőkártya, amelyben a fizikai folyamatot erősítik, mintavételezik és amely a kapott digitális jeleket átadja egy számítógépnek. A generált sorozatot folyamatosan ellenőrizni kell, mert a fizikai folyamatot befolyásolja a környezete, amely enyhébb esetben az eloszlás megváltozását, súlyosabb esetben degenerációt okoz.

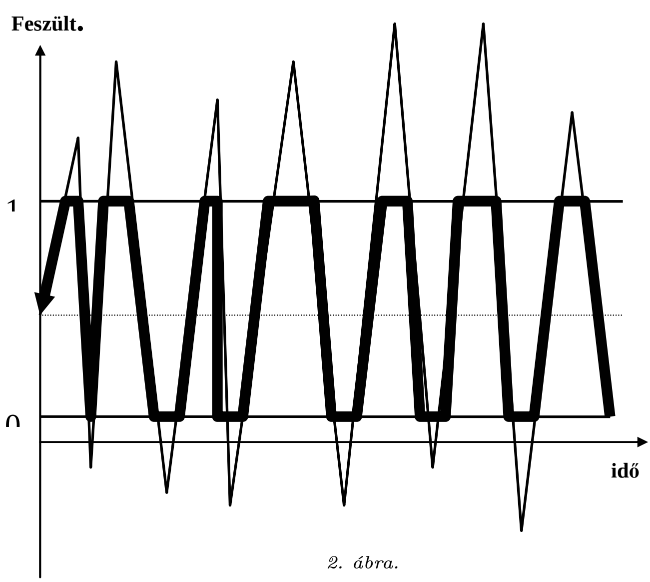
A következőkben leírjuk egy PC-be illeszthető fizikai generátorkártya felépítését. Ennek a fizikai véletlenszám generátornak az alapja, a zajforrás egy ún. *Zener dióda.* Az ezen átfolyó áram feszültsége alapvető fizikai törvényszerűségek miatt kismértékben ugyan, de véletlenszerűen ingadozik. Az ingadozás „frekvenciája” 0 és néhány száz (200) kHz között változik (*1. ábra*).

A kártyára épített elektronika ezt az ingadozást több fokozatban erősíti, majd a jel „közepén” szimmetrikusan kijelöl egy intervallumot, amelynek két szélső értéke lesz a 0 és az 1 érték (*2. ábra*).

A jelet ezután négyszögesítjük. Ezután kerül sor a mintavételezésre, amelynek frekvenciája néhány ezer bit másodpercenként, vagyis a mintavételezés frekvenciája nagyságrendekkel kisebb a kiinduló jel frekvenciájánál, ami a szomszédos minták függetlenségét biztosítja.



1. ábra.



2. ábra.

A generátorkártyán két független jelforrás és mintavevő rendszer működik, a két forrásból származó biteket modul 2 (xor) összeadva kapjuk azt a bitet, amely a generátor outputja lesz. A PC-be illesztett kártyához egy, a PC operációs rendszerén futó driver tartozik, amely statisztikai ellenőrzéseket is végez az outputra, s szélsőséges esetben riaszt. Szükséges még egy applikáció, amely segíti a kártya installálás utáni kalibrálását, és statisztikai teszteket végez, amelyek eredményét a felhasználó számára meg is jeleníti. Többszörösen hangsúlyozzuk *statisztikai tesztek folyamatos elvégzésének* szükségességét, lásd [3].

Pszeudo-véletlenszámok generálása

Az atombomba gyártásánál felmerült nagyon nagy mennyiségű véletlenszám előállításának az igénye. Fából vaskarika kívánalomként az is felmerült, hogy ezek reprodukálhatók legyenek. Neumann János 1944-ben javasolta azt, hogy erre a célra álvéletlen számokat használjanak. Valódi véletlenszámok helyett determinisztikus számsorozatot állítsanak elő, amelyek *statisztikailag ugyanúgy viselkednek, mintha valódi véletlen sorozatok lennének.* Ezeket a számsorozatokat ismételten elő lehet állítani, reprodukálni, ami a számítógépek hőskorában rendkívül fontos követelmény volt. Az algoritmikusan előállítható, véletlenszerűen viselkedő determinisztikus számsorozatokat *pszeudovéletlen* (álvéletlen) sorozatoknak nevezik.

A Neumann János által javasolt eljárás nagyon egyszerű volt. Kiindulásként választott egy n jegyű számot kezdőértéknek. A pszeudo-véletlen sorozat következő számának előállításához az utolsónak generált n jegyű számot négyzetre emelte, majd az így nyert négyzet középső n jegyéből álló szám lett a sorozat új száma. (Az n jegy szükség esetén nullákkal kezdődhet.)

A négyzetközép módszernek ma már csak történelmi jelentősége van. Pszeudo véletlen számsorozatok előállítására a leggyakrabban a *lineáris kongruencia generátorokat* használják. Ezt a módszert D. A. Lehmer javasolta 1949-ben. Ehhez választanak egy $R(0)$ *egész kezdőértéket, egy A multiplikatív és egy B additív konstanst*, valamint egy nagy M *modulust.* Ezek segítségével az utolsónak előállított $R(N)$ számból kiszámítják az

$$R(N+1) = A * R(N) + B \pmod M$$

egész számot. Ebből egy további $U(N)$ álvéletlen számot M-mel való osztással kapunk. Ez az osztás egy bizonyos fajta standardizálást szolgál: az $U(N)$ számok mindig 0 és 1 közé esnek. A számítógépek RND generátora ilyen számok sorozatát szolgáltatja. Innen a $\{0, 1, \ldots, k-1\}$ halmazba eső egyenletes eloszlású egész számokat az

$$[X(N) = k * U(N)]$$

képlet ad meg (adja vissza).

A paraméterek megválasztásával és az előállított sorozat statisztikai vizsgálatával kapcsolatos kérdésekről D. Knuth (1987) 2. kötete ad kimerítő tájékoztatást.

Az algoritmikus eljárások gyors végrehajtására különböző célgépeket szerkesztettek, melyek valódi véletlen sorozatok helyett determinisztikus, de véletlenszerűen viselkedő sorozatokat alkalmaznak. Gyors hardver megoldást tesznek lehetővé a maximális periódusú lineáris visszacsatolt shift regiszterek (angol rövidítéssel LFSR).

Hibrid generátorok

Már a véletlenszámok tömeges előállításának hőskorában megszületett az az ötlet, hogy bármilyen nagy munkával is, de célszerű *egyszer* nagymennyiségű véletlenszámot előállítani, és ezeket közkinccsé téve belőlük bárki, bármikor tetszőlegesen sokat felhasználhat, lásd [2]. Természetesen ez nem egy megbízható megoldás a kriptográfusok számára. Hiszen éppen a legfontosabb követelmény, a kiismerhetetlenség sérül meg. Ezt kell orvosolni ahhoz, hogy ilyen típusú megoldás számukra is használhatóvá váljon. Ezt több lehetőség is segítheti:

- A CD-ROM tárolók megjelenése az általában egyetlen alkalmazáshoz szükséges véletlenszám mennyiség rendkívül sokszorosának megőrzését teszi lehetővé.
- A CD-ROM-okat abszolút biztonságos körülmények között lehet teleírni valódi véletlen (fizikai) generátorok segítségével, miközben a teleírás történhet szubjektív (személyi) beavatkozás nélkül.
- Véletlenszerű kezdettel kis blokkokat úgy lehet kiemelni a CD-ROM-ról, hogy még a kezelő sem képes felismerni, honnan származnak, tehát védelmet lehet biztosítani a saját kezelő személyzettel szemben.

Billentyűleütéssel generált véletlenszám

Javaslunk egy olyan fizikai véletlenszám generátort, amely az internettől függetlenül, „kézi” munkával, a feladat megengedte sebesség (lassúság) mellett állít elő véletlenszámokat. *Az újdonságot az adja meg, hogy az alkalmazást igyekszünk az ad-hoc elemektől megszabadítva, lehetőség szerint szabatossá tenni. Ezért ez újnak tekintendő generálási módszert is eredményez.*

Ebben az eljárásban a véletlenszerűséget a klaviatúrán két egymás után leütött billentyű leütése között eltelt idő adja meg. Ezt az eltelt időt mérjük meg a számítógép beépített órája által megengedett pontossággal. A nyert számot elosztjuk 16-tal, maradékos osztással. A maradékot tekintjük generált (hexadecimális) véletlenszámnak.

Kissé formálisabban a következő lesz az algoritmusunk (amelyben a számítógép pszeudo-véletlenszám generátorát is felhasználjuk).

Algoritmus

1. lépés. Megadunk egy nyomtatható (képernyőre írható) karakterekből álló $ABC\$$ stringet (vagy mátrixot). Legyen a stringben (mátrixban) szereplő jelek száma M, ami a program paramétere. A stringben szereplő karakterek ismeretén kívűl bizonytalansági faktort jelenthet a benne szereplő jelek sorrendje is.

2. lépés. A program deklarálja, hogy csak windos alatt futtatható. Közli, hogy a generált véletlenszámokat mindig az `rndout.doc` fájlba menti. Felszólítja a felhasználót, hogy gondoskodjon az esetleges ilyen nevű fájl tartalmának a mentéséről, mert ezt a program felülírja.

3. lépés. A program rákérdez a generálandó hexadecimális jegyek N számára. Ez a szám 1 és 500 közti tetszőleges egész lehet.

4. lépés. Állítsuk be a PC pszeudovéletlen generátorát egy véletlen kezdőértékre, és generáljunk egymás után N pszeudovéletlen számot az $1, 2, \ldots, M$ egész számok közül.

5. lépés. Minden egyes generált pszeudovéletlen egész esetén írjuk ki a képernyőre az $ABC\$$ string ennyiedik betűjét. (Ez szervezhető $N = 280$ esetén például 4 darab 70 betűs sorba). Ezzel kezdődik a véletlen-hexa generálás.

6. lépés. Üssük le a billentyűzeten egymás után a képernyőn levő N karaktert, miközben a PC méri és kódolja az egymás utáni leütések közti időt, és leírja (feljegyzi, tárolja) a generált hexadecimális számot. A kiírás megint szervezhető 4×70-es mátrixba, ahogy ezt a konkrét példánk teszi is.

Teljesen lényegtelen, hogy a számítógép milyen pszeudo-véletlenszám generátort alkalmaz. A program működésének bemutatására $N = 4 \times 70$ paraméter mellett megadunk egy *leütendő sorozatot*, majd a kiírt sorozat leütésekor létrejövő véletlenszám sorozatot.

```
3gkmh%g7wl g%zi3$6e=, wl3jz-bpfm t6ibcbp8nt 2=a,%p4=ma 24hrcrcp08 ym7yms5x59
ecx4d-tpzn 3$@h5tpa60 37pfakprwu a34-yx@0pv 0jh68$#e7! bbizueg,kw cynu.%r6$8
mesw%wxubn w6b3phq9pz d!a$%ny,x0 kwkzl!%=rf 03ap4xz9%3 za4yrct3o@ xmg0bu@d!1
su=a%cyv!2 .,9uton6yx oo9@x$zcph gmazktp3q3 t1zhw%!t-j ok,t$dp8,v r4wzg8%,xo
```

A leütéssel generált véletlenszám sorozat:

```
82E5F343C5 687D4A1574 F2D60BE321 5B105D9595 70551049C7 65D2EF56BB 9D1A8E27A7
7264AA6BC8 5E906AD00C C6BF053066 7012CC702A C0B118E671 A35D4EDED6 C0EB221A4C
9BBBE0043F 13272E40B2 8781BCCBC9 AF3294E75D 32086A97B2 145276860C 65EB2B9161
96CE07F282 427882DB76 DBC9E59E11 DCB0956BF8 CBF5559E5C 37CFB0CE5D 6ABB3668E0
```

A generált sorozat elemeinek *gyakoriság-eloszlása:*

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
22	17	22	11	12	24	23	19	14	15	12	25	21	14	19	10

1. táblázat.

A χ^2 próba-statisztika értéke: 21,4857. A 15 szabadsági fokhoz tartozó kritikus értékek

P	0,1	0,05	0,01
χ^2 érték	22,3	25,0	30,6

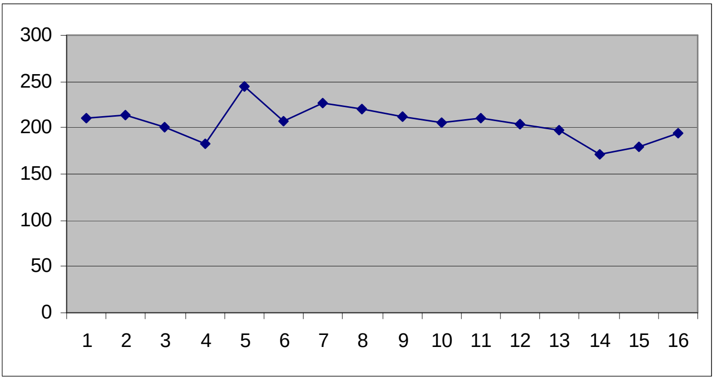
2. táblázat.

Az Excel program segítségével adatainkat grafikusan szemléltethetjük is. Illusztrációként megadunk egy további 4×70-es ,,valódi” véletlen ,,leütéses” mátrixnak megfelelő gyakorisági grafikont.

Az $N = 280$ elemű minta gyakoriság-eloszlása a 3. táblázatban, grafikonja pedig a *3. ábrán* látható.

0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
16	15	20	17	19	19	16	18	17	16	19	19	19	15	15	20

3. táblázat.



3. ábra. Gyakorisági poligon

Az empirikusan generált véletlen sorozatokat kötelező felhasználás előtt statisztikailag ellenőrizni. Erre a szokásos tesztek megfelelnek, de a generálás módját figyelembe vevő speciális tesztek is szükségesek lehetnek.

Ellenőrizendők a következő feltételezések:

- *Egyenletesség*: Bármely időpontban a 0-k és 1-ek addigi előfordulási száma várhatóan megegyezik.
- *Skálázhatóság*: Minden univerzálisan kiválasztott részsorozat is kielégíti a statisztikai teszteket.
- *Konzisztencia*: Egy pszeudo véletlen generátor statisztikailag nem függhet az algoritmus kezdőértékétől.

Összefoglalás

Digitális dokumentumok azonosítására használt algoritmusok véletlenszámokat alkalmaznak. Ezek a véletlenszámok az alkalmazótól és a dokumentumtól függenek. Ezért az alkalmazónak magának kell generálni ezeket. A dolgozat erre javasol egy olyan módszert, amely semmilyen eszközt nem igényel. A módszer azt használja ki, hogy egy PC billentyűzetén két billentyű leütése közt eltelt idő véletlenszerűen változik.

Irodalom

1. Knuth, D. E. (1987): *A számítógép-programozás művészete 2. Szeminumerikus algoritmusok.* Műszaki Kiadó, Budapest
2. Nemetz, T.–Papp, P. (1998): Hybrid Random Byte Generators, in: *Global IT Security*, Papp, Gy.-Posch, R. (ed), Proc. of the 14th Int. Conf. on Info. Security, Wien, 366–380
3. NIST (2001): *Special Publication 800-22* Statisztikai próbák a kriptográfiai alkalmazásoknál használt véletlen- és pszeudo-véletlenszám generátorokra. (Javított kiadás: 2001. május 15.)

GENERATING PERSONAL RANDOM NUMBERS

Algorithms for the identification of digital documents apply random numbers. These random numbers depend on the user and the document hence the user himself must generate them. In this paper we suggest an algorithm to this task which does not require any other tools. Our method exploits the observation that the length of the time interval between two presses of keys on a keyboard of a PC varies randomly.

KRITÉRIUMOK PÁROS ÖSSZEHASONLÍTÁS MÁTRIXOKRA[1]

KÉRI GERZSON
MTA SZTAKI

1 Bevezetés

Többszempontú döntési problémák kezelésének egyik kulcskérdése a páros összehasonlítás mátrixok elemzése, melynek eredményeképpen egy páros összehasonlítás mátrixot vagy elfogadunk, vagy pedig annak korrekcióját kérjük az értékelőtől. Az elemzés során alkalmas kritériumok alapján vizsgálhatjuk a páros összehasonlítás mátrixok következetességének a mértékét. A legelterjedtebb eljárás a Saaty [3] által javasolt módszer a *CR* következetlenségi hányados kiszámításán alapul, s e módszer szerint a páros összehasonlítás mátrixot akkor fogadjuk el jónak, ha *CR* értéke 0.1-nél kisebb. E módszert használja például az *Expert Choice* szoftver.

Gass [1] már évekkel ezelőtt felhívta a figyelmet arra, hogy páros összehasonlításokkal végzett rangsorolás során célszerű törekedni arra, hogy három elemből álló intranzitív ciklusok lehetőleg ne, vagy minél kisebb számban forduljanak elő. E gondolathoz kapcsolódva Gass leszögezi, hogy általánosan elfogadott elv a fellépő intranzitivitások elemzésének és lehetőség szerinti kiküszöbölésének a kívánalma. Ezután az idézett [1] cikk a páros összehasonlítás mátrixokkal, az intranzitivitások vizsgálatával és kezelésével foglalkozik azon megszorító feltétel mellett, hogy minden páros összehasonlítás döntést eredményez valamelyik összehasonlított objektum javára, tehát nem fordulnak elő azonos értékűnek vagy összehasonlíthatatlannak talált objektumok. A páros összehasonlítás mátrixára vonatkozóan ez a feltétel úgy fogalmazható meg, hogy a főátlón kívül a mátrixban nem lehetnek 1 értékű mátrixelemek.

A döntési rendszerek gyakorlati alkalmazásában jártas szakértők általában abban is egyetértenek, hogy a gyakorlatban előforduló döntési problémák esetén a tranzitivitás biztosítása, vagyis az intranzitív hármasok teljes megszüntetése, általában nem követelhető meg. Gass és Standard [2] kifejti és indokolja, hogy többszempontú döntések és kvalitatív szempontok esetén az összehasonlításoknak nem feltétlenül kell tranzitívnak lenniük. A szerzők megvizsgálják és a tranzitivitás szempontjából elemzik több gyakorlati feladat páros összehasonlítás mátrixát.

Véleményem szerint a kirívó mértékű intranzitivitásokat nem okvetlenül kell elfogadni, ezért törekedni kell arra, hogy az értékelők képesek legyenek a durvább intranzitivitásokat észrevenni és korrigálni az értékelés során. Ezért fordult figyelmem az objektumhármasok vizsgálatára. A dolgozatban Gass

[1] Beérkezett: 2005. augusztusz 21.

[1] cikkéhez hasonlóan a páros összehasonlítás mátrixok gráf reprezentációja segítségével vizsgálom e mátrixokat és definiálom azok különböző típusait a 3x3 méretű részmátrixok tulajdonságai alapján. Az elvégzett elemzések és az azok alapján megfogalmazott eredmények jellegükben teljesen eltérnek az [1] cikk eredményeitől, és az is lényeges eltérés, hogy Gasstól eltérően nem teszem fel, hogy minden páros összehasonlítás döntést eredményez, tehát megengedek 1 értékű elemeket a mátrix főátlóján kívül is. Az AHP módszertant [3] nem elvetve, hanem azt kiegészítve, megfogalmazok néhány —egymástól többé-kevésbé eltérő— kritériumot, vizsgálom azok teljesülésének feltételét, továbbá útmutatási javaslatot adok a kritériumok alapján következetlennek talált páros összehasonlítás mátrixok javítására.

2 Értékelés páros összehasonlítással

Tekintsünk véges számú azonos típusú

$$O_1, O_2, \ldots, O_n$$

objektumot. Ezek lehetnek például egy döntési probléma alternatívái. Az objektumok meghatározott szempont szerinti, páros összehasonlítással végzett értékelése során az értékelő minden lehetséges $i, j \in \{1, 2, \ldots, n\}$ párra megállapítja, hogy az O_i, O_j objektumok közül a vizsgált szempontból melyik előnyösebb, mint a másik. A teljes körűen végrehajtott páros összehasonlítás eredménye egy

$$A = (a_{ij})_{i,j=1,2,\ldots,n}$$

mátrix, ahol $a_{ij} = u(> 1)$, ha az O_i objektum u-szor előnyösebb, mint O_j, illetve $a_{ij} = v(< 1)$, ha az O_i objektum v-szer kevésbé előnyös, mint O_j. Ha az értékelő nem tudja megmondani, hogy O_i és O_j közül melyik előnyösebb, akkor $a_{ij} = 1$.

Egy A páros összehasonlítás mátrixra nyilvánvalóan teljesülnie kell az $a_{ij} > 0$ és $a_{ji} = 1/a_{ij}$ $(i, j = 1, 2, \ldots, n)$ relációknak. Az ilyen tulajdonságú A mátrixokat pozitív reciprok mátrixoknak nevezzük.

Definíció. *Az A mátrix pozitív reciprok mátrix, ha minden eleme pozitív, a főátlóra szimmetrikus elemek pedig egymás reciprokai.*

A fenti definícióból már következik, hogy a főátlóban álló elemek értéke 1. Az is nyilvánvaló, hogy egy pozitív mátrix akkor és csak akkor pozitív reciprok, ha a mátrixelemek logaritmusaiból álló mátrix antiszimmetrikus.

3 Páros összehasonlítás mátrixok gráfja

Definíció. *Egy pozitív reciprok A mátrix gráfjának nevezzük azt az n szögpontú G_A irányított gráfot, melynek i-edik csúcsából a j-edik csúcsába akkor és csak akkor vezet irányított él, ha $a_{ij} > 1$.*

A G_A gráf $A_1, A_2, \ldots, A_n$ csúcsai az $O_1, O_2, \ldots, O_n$ objektumoknak felelnek meg, abban az értelemben, hogy A_i-ból A_j-be akkor és csak akkor vezet él, ha az O_i objektum előnyösebb, mint az O_j objektum.

Pozitív reciprok A mátrixból kiindulva mindig olyan G_A gráfhoz jutunk, melynek egyik csúcspontjából sem vezet él önmagához, továbbá az i-edik csúcspontból a j-edikbe vezető és a j-edik csúcspontból a i-edikbe vezető lehetséges élek közül legfeljebb egy lehet G_A-nak éle.

Az A mátrix a G_A gráfot egyértelműen meghatározza, ennek megfordítása azonban nem igaz, mert már 2×2-es vagy 3×3-as méretben is könnyű konstruálni olyan különböző pozitív reciprok mátrixokat, melyekhez ugyanaz a gráf tartozik, ilyenek például 2×2-es méretben az

$$\begin{pmatrix} 1 & 2 \\ 1/2 & 1 \end{pmatrix} \quad \text{és} \quad \begin{pmatrix} 1 & 3 \\ 1/3 & 1 \end{pmatrix}$$

mátrixok, 3×3-as méretben pedig az

$$\begin{pmatrix} 1 & 2 & 6 \\ 1/2 & 1 & 3 \\ 1/6 & 1/3 & 1 \end{pmatrix}, \quad \begin{pmatrix} 1 & 3 & 4 \\ 1/3 & 1 & 5 \\ 1/4 & 1/5 & 1 \end{pmatrix} \quad \text{és} \quad \begin{pmatrix} 1 & 3 & 3 \\ 1/3 & 1 & 3 \\ 1/3 & 1/3 & 1 \end{pmatrix}$$

mátrixok. A kölcsönösen egyértelmű megfeleltetést úgy tudjuk biztosítani, ha a G_A gráf éleihez súlyokat rendelünk: a gráf tetszőleges irányított élének a súlya az él i, j végpontjaihoz tartozó a_{ij} és a_{ji} közül az 1-nél nagyobb értékű mennyiség. Most tekintsük át a 2×2-es és 3×3-as méretű páros összehasonlítás mátrixok gráfjának lehetséges eseteit, az élek súlyait egyelőre figyelmen kívül hagyva.

2×2-es méret esetén csak 2 lehetőség van; az egyik esetben a G_A gráf üres ($a_{12} = 1$), a másik esetben egyetlen éle van ($a_{12} \neq 1$).

3×3-as-as méret esetén a G_A gráfra a következő esetek lehetségesek:

a) 3 irányított él – 2 eset: G_1 és G_2 az 1. ábrán;

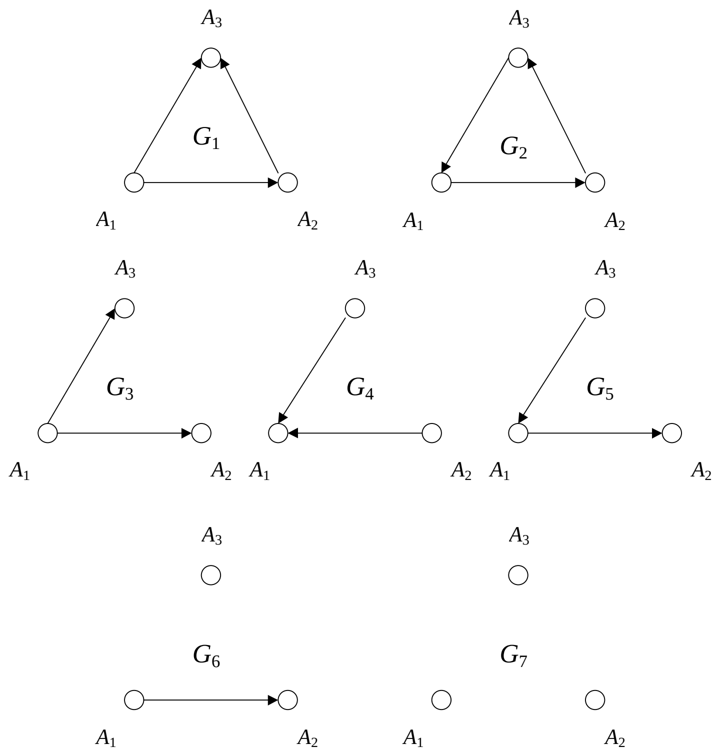
b) 2 irányított él – 3 eset: G_3, G_4 és G_5 az 1. ábrán;

c) 1 irányított él – 1 eset: G_6 az 1. ábrán;

d) 0 irányított él – 1 eset: G_7 az 1. ábrán.

Ennél nagyobb méretekre az esetek száma nagyon gyorsan növekszik, ezért nagyobb méretű döntési problémák esetén a páros összehasonlítási mátrixok gráfjában a 3 csúcsból álló részgráfok vizsgálatát javasoljuk a mátrixok elemzése során.

Itt, és a további tárgyalás során is, részgráf alatt mindig feszített részgráfot értünk, vagyis egy részgráf élei közé tartozónak tekintjük a kiindulásul vett gráf mindazon éleit, melyeknek mindkét csúcsa a részgráfhoz tartozik.



1. ábra.

4 Konzisztens és kvázi-konzisztens páros összehasonlítás mátrixok

Ideális esetben a páros összehasonlítás eredménye konzisztens, ami alatt azt értjük, hogy az eredményül kapott pozitív reciprok mátrix konzisztens a következő definíció értelmében.

Definíció. *Egy pozitív reciprok mátrix konzisztens, ha minden $i, j \in \{1, 2, \ldots, n\}$ párra – alkalmas pozitív $p_1, p_2, \ldots, p_n$ súlyokkal – teljesül $a_{ij} = p_i/p_j$.*

A fenti definícióból látszik, hogy konzisztens mátrix esetén az e mátrixot eredményező preferencia reláció aszimmetrikus, tranzitív és ráadásul negatívan is tranzitív: $p_i > p_j$, $p_j > p_k$ esetén mindig $p_i > p_k$, illetve $p_i \leq p_j$, $p_j \leq p_k$ esetén mindig $p_i \leq p_k$.

Kis n esetén ($n = 2, 3$, esetleg 4) még könnyű az n objektumot együttesen, ezek teljes rendszerét áttekintve "mérlegelni", s ennek alapján konzisztens összehasonlítás mátrix előállítását biztosítani, n növelésével azonban ez egyre nehezebbé válik. Az említettnél nagyobb méretek esetén tehát más módszerekre van szükség.

Példák konzisztens mátrixokra $n = 3$ esetén: Az 1. ábrán felsorolt gráfok közül G_2, G_5 és G_6 kivételével mindegyik tartozhat konzisztens mátrixhoz, például az

$$\begin{pmatrix} 1 & 2 & 6 \\ 1/2 & 1 & 3 \\ 1/6 & 1/3 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 2 & 2 \\ 1/2 & 1 & 1 \\ 1/2 & 1 & 1 \end{pmatrix}, \begin{pmatrix} 1 & 1/3 & 1/3 \\ 3 & 1 & 1 \\ 3 & 1 & 1 \end{pmatrix} \text{ és } \begin{pmatrix} 1 & 1 & 1 \\ 1 & 1 & 1 \\ 1 & 1 & 1 \end{pmatrix}$$

mátrixhoz rendre a G_1, G_3, G_4, illetve G_7 irányított gráf tartozik.

Könnyű belátni, hogy egy páros összehasonlítás mátrix akkor és csak akkor konzisztens, ha annak minden főátlóra szimmetrikus harmadrendű szubmátrixa konzisztens. Egy harmadrendű pozitív reciprok mátrix pedig – mint láttuk – akkor és csak akkor konzisztens, ha annak gráfja a G_1, G_3, G_4, G_7 gráfok valamelyikével izomorf. Következésképpen tetszőleges konzisztens mátrixhoz tartozó G_A gráf minden három szögpontú részgráfja G_1, G_3, G_4 és G_7 valamelyikével izomorf. Jó lenne ezt fordítva is tudni, vagyis felvetődik a kérdés: Igaz-e, hogy ha egy G irányított gráf minden harmadrendű részgráfja G_1, G_3, G_4, G_7 valamelyikével izomorf, akkor található olyan konzisztens pozitív reciprok mátrix, melynek gráfja az adott G gráf? A kérdésre „igen" a válasz, mert bebizonyítjuk a következő tételt.

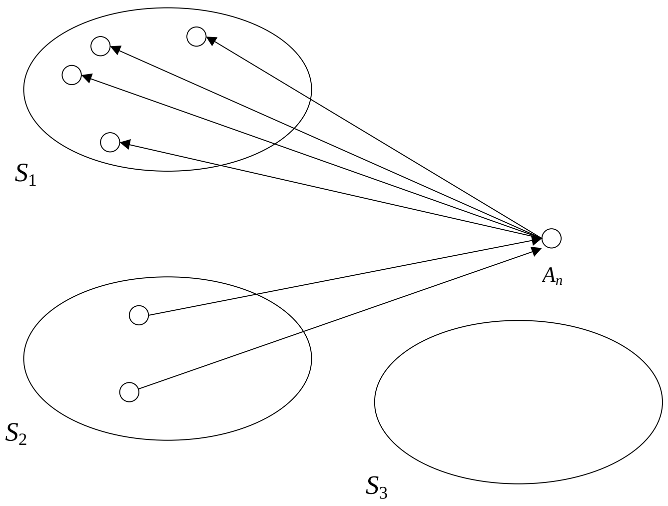
Tétel. *Ha G egy olyan irányított gráf, melynek bármely két csúcspontja között legfeljebb egy irányított él van és G minden harmadrendű részgráfja az 1. ábrán megadott G_1, G_3, G_4, G_7 valamelyikével izomorf, akkor a G gráf $A_1, A_2, \ldots, A_n$ csúcsaihoz lehet olyan pozitív $p_1, p_2, \ldots, p_n$ súlyokat rendelni, hogy ezekre $p_i > p_j$ akkor és csak akkor áll fenn, ha a G gráfban A_i-ből A_j-be vezet irányított él.*

Bizonyítás. A csúcsok számára vonatkozó teljes indukciót alkalmazunk. Ha a G gráfnak 3 csúcsa van, akkor a tétel állítása nyilvánvaló. Ha a G gráf csúcsainak száma $n > 3$, akkor tekintsük a G-ből egy tetszőleges csúcs és az e csúcsba érkező vagy e csúcsból induló élek elhagyásával keletkező G' részgráfot.

Az indukciós feltevés szerint léteznek olyan $p_1, p_2, \ldots, p_{n-1}$ pozitív számok, hogy ezekre $p_i > p_j$ akkor és csak akkor áll fenn, ha a G' gráfban A_i-ből A_j-be vezet irányított él.

A G' gráf $A_1, A_2, \ldots, A_{n-1}$ csúcsainak a halmazát bontsuk fel három diszjunkt halmaz: S_1, S_2 és S_3 egyesítésére a G-ben A_n-ből vagy A_n -be vezető élek létezése és iránya alapján, és legyen $A_i \in S_1$, ha A_n-ből A_i-be vezet él, $A_i \in S_2$, ha A_i-ből A_n-be vezet él, $A_i \in S_3$, ha sem A_n-ből A_i-be, sem A_i-ből A_n-be nem vezet él (l. 2. ábra).

Az így meghatározott S_1, S_2, S_3 halmazok bármelyike lehet üres halmaz is. A bizonyítást folytatva pontokba szedjük az egyes mozzanatokat.



2. ábra

1. Ha $A_i \in S_3$ és $A_j \in S_3$, akkor $p_i = p_j$. Ellenkező esetben ugyanis a G gráf A_i, A_j, A_n csúcsaihoz tartozó részgráf G_6-tal izomorf lenne.

2. Ha $A_i \in S_1$ és $A_j \in S_2$, akkor $p_j > p_i$. Ellenkező esetben a G gráf A_i, A_j, A_n csúcsaihoz tartozó részgráf G_2-vel vagy G_5-tel izomorf lenne.

3. Ha $A_i \in S_1$ és $A_j \in S_3$, akkor $p_j > p_i$. Ellenkező esetben a G gráf A_i, A_j, A_n csúcsaihoz tartozó részgráf G_5-tel vagy G_6-tal izomorf lenne.

4. Ha $A_i \in S_2$ és $A_j \in S_3$, akkor $p_i > p_j$. Ez az állítás az előbbihez hasonló módon bizonyítható.

5. Az előbbieket összefoglalva:

a) ha S_3 nem üres, akkor valamennyi S_3-beli csúcshoz azonos p_i súly tartozik;

b) bármely S_1-beli csúcs súlya határozottan kisebb bármely S_2-beli és S_3-beli csúcs súlyánál.

c) bármely S_2-beli csúcs súlya határozottan nagyobb bármely S_1-beli és S_3-beli csúcs súlyánál.

6. Ha az S_3 halmaz nem üres, akkor csak abban az esetben kapunk a tétel követelményének megfelelő $p_1, p_2, \ldots, p_n$ súlyrendszert, ha $p_n = p_k$, ahol $A_k \in S_3$.

7. Ha S_3 üres halmaz, de S_1 és S_2 nem üres, akkor A_n-hez olyan p_n súlyra van szükség, melyre

$$p_n > \max\{p_i : A_i \in S_1\}$$

és

$$p_n < \min\{p_i : A_i \in S_2\}.$$

Ilyen p_n érték létezik, mivel 5. szerint $\max\{p_i : A_i \in S_1\} < \min\{p_i : A_i \in S_2\}$.

8. Ha S_3 és S_2 üres halmaz, akkor legyen

$$p_n > \max\{p_i : A_i \in S_1\},$$

egyébként tetszőleges, ha pedig S_3 és S_1 üres halmaz, akkor legyen

$$p_n < \min\{p_i : A_i \in S_2\},$$

egyébként tetszőleges pozitív értékű szám.

A gondolatmenetből látható, hogy az indukciós feltevés alapján meghatározott $p_1, p_2, \ldots, p_{n-1}$ súlyokat a 6., 7. vagy 8. szerint választott p_n-nel kiegészítve a tétel követelményének megfelelő súlyrendszerhez jutunk. ∎

A bizonyításból az is látszik, hogy a G gráf csúcsainak ezek súlya alapján történő rendezése során az A_n csúcs egyértelműen sorolódik be vagy az S_3-hoz tartozó csúcsok mellé, ezekkel azonos súlyúnak (amennyiben az S_3 halmaz nem üres), vagy pedig az S_2-höz tartozó csúcsok után, de az S_1-hez tartozó csúcsok elé (ha az S_3 halmaz üres).

A következő definícióval bevezetjük a kvázi-konzisztens pozitív reciprok mátrix fogalmát, ami logikailag kapcsolódik az előbb bizonyított tételhez.

Definíció. *Egy pozitív reciprok A mátrix kvázi-konzisztens, ha található ugyanolyan méretű konzisztens $A^\star$ mátrix úgy, hogy tetszőleges i, j indexpárra teljesül, hogy $a_{ij} > 1$ (illetve $a_{ij} = 1, a_{ij} < 1$) esetén $a^\star_{ij} > 1$ (illetve $a^\star_{ij} = 1, a^\star_{ij} < 1$).*

Nyilvánvaló, hogy a 3×3-as méretű pozitív reciprok mátrixok közül azok és csak azok kvázi-konzisztensek, melyek gráfja G_1, G_3, G_4 vagy G_7 típusú. Ezért a fenti tételből következik, hogy ha egy pozitív reciprok mátrixban minden, a főátlóra szimmetrikusan elhelyezkedő, 3×3-as méretű szubmátrix kvázi-konzisztens, akkor a teljes mátrix is kvázi-konzisztens.

5 Tranzitív és erősen tranzitív páros összehasonlítás mátrixok

A következőkben bevezetjük a konzisztencia két lazább változatát. Közülük az elsőt pusztán a G_A irányított gráf alapján értelmezzük, ez tehát csak egy kvalitatív jellemzője az A mátrix következetességi fokának, a másodikhoz viszont figyelembe vesszük a G_A gráf irányított éleihez rendelt súlyokat is.

Ha a páros összehasonlítást végző értékelő azt mondja, hogy az O_i objektum előnyösebb, mint az O_j, ez utóbbi pedig előnyösebb, mint az O_k, akkor a logika szerint elvárhatjuk tőle, hogy az O_i és O_k objektumokat hasonlítva O_i-t is előnyösebbnek ítéli O_k-nál. Az ezen elvárásnak eleget tevő értékelések páros összehasonlítás mátrixát tranzitívnek nevezzük.

Definíció. *Egy pozitív reciprok A mátrix tranzitív, ha tetszőleges i, j, k indexhármasra teljesül, hogy $a_{ij} > 1$ és $a_{jk} > 1$ esetén mindig fennáll $a_{ik} > 1$ is.*

Tranzitív mátrixoknak a definícióból következő további nyilvánvaló tulajdonságai:

$$\begin{aligned}
a_{ij} &< 1 \text{ és } a_{jk} < 1 \text{ esetén mindig } a_{ik} < 1;\\
a_{ij} &= 1 \text{ és } a_{jk} > 1 \text{ esetén mindig } a_{ik} \geq 1;\\
a_{ij} &= 1 \text{ és } a_{jk} < 1 \text{ esetén mindig } a_{ik} \leq 1;\\
a_{ij} &> 1 \text{ és } a_{jk} = 1 \text{ esetén mindig } a_{ik} \geq 1;\\
a_{ij} &< 1 \text{ és } a_{jk} = 1 \text{ esetén mindig } a_{ik} \leq 1.
\end{aligned}$$

A logika alapján elvárható az is, hogy ha az értékelést végző személy ítélete szerint az O_i objektum előnyösebb, mint O_j, ez pedig u-szor előnyösebb, mint O_k, akkor O_i is legalább u-szor előnyösebb, mint O_k, vagyis $a_{ij} > 1$ -ből következik, hogy $a_{ik} \geq a_{jk}$. Ennek alapján bevezetjük a következő definíciót:

Definíció. *Egy pozitív reciprok A mátrix erősen tranzitív, ha tetszőleges olyan i, j indexek esetén, melyekre $a_{ij} > 1$, minden k-ra fennáll az $a_{ik} \geq a_{jk}$ egyenlőtlenség.*

Az erősen tranzitív mátrix fogalma már nem tisztán kvalitatív tulajdonságot fejez ki, ezért a mátrix gráfjára történő ekvivalens átfogalmazás most önmagában a G_A gráfra nem értelmezhető, hanem csak a —3. szakaszban leírt módon— él-súlyokkal ellátott gráfra.

Definíció. *Egy pozitív reciprok A mátrixból származtatható G_A gráf (illetve él-súlyokkal ellátott G_A gráf) tranzitív (illetve erősen tranzitív), ha maga az A mátrix tranzitív (illetve erősen tranzitív).*

A definició folytán nyilvánvaló, hogy egy pozitív reciprok mátrix akkor és csak akkor tranzitív (erősen tranzitív), ha minden harmadrendű, főátlóra szimmetrikus részmátrixa tranzitív (erősen tranzitív).

Az is nyilvánvaló a három mátrixosztály (konzisztens, tranzitív, illetve erősen tranzitív mátrixok) definíciója alapján, hogy minden konzisztens mátrix erősen tranzitív, és minden erősen tranzitív mátrix tranzitív.

A harmadrendű pozitív reciprok mátrixokhoz tartozó, az 1. ábrán felsorolt gráfokat végignézve látjuk, hogy közülük a G_1, G_3, G_4, G_6 és G_7 gráfok tranzitív, a G_2, és G_5 gráfok nem tranzitív mátrixhoz tartoznak. Kimondható tehát a következő állítás:

Tétel. *Egy G_A irányított gráf akkor és csak akkor tranzitív, ha a gráf három csúcsból álló részgráfjai között nem fordul elő sem G_2-vel, sem G_5-tel topológiai értelemben ekvivalens gráf.*

Eszerint a konzisztens, illetve tranzitív mátrixok gráfjai közötti eltérés úgy is megfogalmazható, hogy az utóbbiak esetén megengedjük a három csúcsból álló részgráfok között G_6 előfordulását is, az előbbiek esetén nem engedjük meg.

Az A mátrixhoz tartozó él-súlyozott G_A gráf erős tranzitivitása is kifejezhető a három csúcsból álló él-súlyozott részgráfok segítségével. Az 1. ábra esetein végigfutva, rövid diszkusszió elvégzésével azt látjuk, hogy a G_2 és G_5 gráf semmilyen súlyozással nem lehet erősen tranzitív, a G_6 és G_7 gráfok viszont eleve erősen tranzitívak, a súlyozástól függetlenül. A G_3, illetve G_4 gráf akkor és csak akkor erősen tranzitív, ha a bennük lévő 2 él súlya azonos. Végül, a G_1 gráf akkor és csak akkor erősen tranzitív, ha az élek súlyaira teljesül $a_{13} \geq \max\{a_{12}, a_{23}\}$. Az fentiekből következik az alábbi állítás helyessége:

Tétel. *Egy pozitív reciprok mátrixhoz tartozó, él-súlyokkal ellátott G_A irányított gráf akkor és csak akkor erősen tranzitív, ha a gráf három csúcsból álló részgráfjai között nem fordul elő sem G_2, sem G_5 típusú gráf, az összes G_3 és G_4 típusú részgráf 2-2 élének súlya azonos ($a_{12} = a_{13}$, illetve $a_{21} = a_{23}$), az összes G_1 típusú részgráf él-súlyaira pedig teljesül $a_{13} \geq \max\{a_{12}, a_{23}\}$.*

Megjegyezzük, hogy konzisztens mátrixok esetén a tranzitivitás nemcsak a "$>$", hanem a "$\geq$" relációra nézve is fennáll, azaz tetszőleges i, j, k indexhármasra teljesül, hogy $a_{ij} \geq 1$ és $a_{jk} \geq 1$ esetén mindig $a_{ik} \geq 1$. A konzisztens mátrix fogalmában tehát egy erősebb tranzitivitás testesül meg, mint a tranzitív mátrix fogalmában. Mivel azonban az előbbieket a konzisztens jelző már kategorizálja, a tranzitív szó szabad marad az előbbi definíció szerinti gyengébb tranzitivitás jellemzésére.

Az értékelési eljárás során a "$\geq$" relációra vonatkozó tranzitivitás teljesítését nem volna helyes elvárni az értékelőktől, amit a következő gondolatmenettel indokolhatunk:

Mivel az összehasonlítások számszerűsítése általában diszkrét (gyakran csak verbálisan kifejezett) skálán történik, nem zárhatjuk ki az olyan eseteket, amikor objektumok valamely $O_1, O_2, \ldots, O_k$ láncára ennek szomszédos elemei között az értékelő nem lát különbséget a vizsgált szempont alapján, ha történetesen a lánc minden tagja csak picivel előnyösebb az előzőnél, azonban könnyen lehetséges, hogy ugyanakkor a két szélső elem (O_1 és O_k) között már egyértelműen és markánsan észlelhető a különbség. Természetesen sok más (többnyire szubjektív) oka is lehet annak, hogy a tranzitivitás nem mindig teljesül az értékelési folyamat során.

6 Konklúzió

Páros összehasonlítás alapján készített pozitív reciprok mátrixok esetén ezek elemzésére használhatjuk a cikkben bevezetett három másik konzisztencia típust. Az elemzés során azt vizsgálhatjuk, hogy a páros összehasonlítás eredményeként kapott mátrix megfelel-e az enyhébb konzisztencia típusok valamelyikének, ha nem, akkor hol és milyen mértékben tér el azoktól? A bevezetett fogalmak haszna lehet az is, hogy – míg diszkrét pontozási skála esetén a klasszikus konzisztencia elvileg nem biztosítható – a három enyhébb konzisztencia típus esetén viszont általában biztosítható.

Felvetjük megfontolásra a páros összehasonlítás mellett triók (objektumhármasok) összehasonlításának a módszerét (amit lényegében már Gass [1] is felvetett): Bármely X, Y, Z objektum-hármas esetén az értékelő válasszon a következő lehetőségek között: 1. X, Y, Z mindegyike egyformán előnyös; 2. kettő közülük, pl. X és Y egyformán előnyös, Z azonban még előnyösebb mindkettőnél; 3. X és Y egyformán előnyös, Z azonban kevésbé előnyös mindkettőnél; 4. X, Y és Z sorrendbe állítható, pl. X előnyösebb, mint Y és Z, Y pedig előnyösebb, mint Z.

Irodalom

1. S. I. Gass, Tournaments, transitivity and pairwise comparison, *J. Op. Res. Soc.*, 49(1998) 616–624.
2. S. I. Gass and S.M. Standard, Characteristics of positive reciprocal matrices in the analytic hierarchy process, *J. Op. Res. Soc.*, 53(2002) 1385–89.
3. T. L. Saaty, *The analytic hierarchy process. Planning, priority setting, resource allocation.* McGraw-Hill, New York, 1980.

CRITERIA FOR PAIRWISE COMPARISON MATRICES

The aim of the paper is to give a small contribution to the methodology of analyzing pairwise comparison matrices. For this purpose, we define several suitable classes of positive reciprocal matrices based on mainly the qualitative feature of triads of a pairwise comparison matrix. Using a graph representation of a pairwise comparison matrix, graph theoretic approach is applied for the argumentation. This is especially useful for proving the theorem in Section 4.

LEXIKOGRAFIKUS DÖNTÉSEK EGY ÁLTALÁNOS DÖNTÉSI MODELLBEN[1]

DOMBI JÓZSEF – VINCZE NÁNDOR
Szegedi Tudományegyetem

A dolgozatban a lexikografikus döntési eljárást egy olyan általános döntési modell keretébe illesztjük bele, amelynek segítségével modellezhetők a PROMETHEE, ELECTRE, illetve a utility eljárások. Egy lexikografikus döntési függvényt konstruálunk meg, preferenciafüggvény és unáris operátor segítségével.

1 Bevezetés

A döntéselmélet története során nagyon sok eljárást fejlesztettek ki. A kezdeteket Condorcet valamint Cramer és Bernoulli modellje jelentette, ezt követte a döntéselmélet fejlődésének legnagyobb lökést adó Neumann-Morgenstern-féle axiomatizált modell, valamint hasznosság-függvény reprezentációk. Ide tartoznak a különböző additív illetve nem-tranzitív modellek, mint a Fishburn-féle SSB modell, Fishburn[4], valamint az outranking módszerek, az ELECTRE, a PROMEETHE vagy a Saaty által kidolgozott AHP módszer, Temesi [9], Olson [6]. A gyakorlati alkalmazás során felmerült a kérdés, hogy ezek a modellek megfogalmazhatók-e egy olyan egységes keretrendszerben, amelyben a paraméterek változtatásával megkaphatók az egyes speciális döntési eljárások? Ennek jelentősége abban áll, hogy számos döntési szituációban több módszer egymás melletti alkalmazásával határozható meg a kimenetek közötti sorrend. Ezt a keretrendszert adja meg Dombi [1, 2].

Dolgozatunk célja a lexikografikus döntés egy olyan numerikus reprezentációjának megadása, mellyel beleilleszthető ebbe a keretbe. Ennek tudható be, hogy a dolgozatban látszólag bonyolult reprezentációt adunk a lexikografikus döntésre. Legyen $a = (x_1, x_2, \ldots, x_n)$ és $b = (y_1, y_2, \ldots, y_n)$ alternatíva, az x_k, y_k hasznossági értékekkel megadva, és $w_1, w_2, \ldots, w_n$ súlyok. Adjuk meg a preferenciát a következő módon:

$$p(a,b) = \sum_{i=1}^{n} w_i \tau_i(p_i(x_i, y_i))$$

ahol a preferenciafüggvény

$$p(x,y) = \frac{y - x + 1}{2}$$

[1] Beérkezett: 2005. április 24.

és $\tau_i : [0,1] \to [0,1]$ egyváltozós monoton függvény. Dolgozatának első részében Dombi [1] megmutatta, hogy ha $\tau_i(x) = x$, akkor a preferenciák súlyozott átlagát kapjuk, másrészt a súlyozott átlag szerinti döntés felcserélhető a preferenciákon alapuló döntéssel, azaz a utilitás jellegű modellt ez a modell tartalmazza. Továbbá bizonyításra kerül az alábbi:

Tétel. Az ELECTRE és PROMETHEE módszerek p^{EL}, p^{PR} preferenciafüggvényeihez léteznek olyan τ_i^{EL} és τ_i^{PR} egyváltozós függvények, hogy

$$p^{EL}(a,b) = \sum_{i=1}^{n} w_i \tau_i(p_i(x_i, y_i))$$

$$p^{PR}(a,b) = \sum_{i=1}^{n} w_i \tau_i(p_i(x_i, y_i))$$

és lineáris esetben például:

$$\tau^{EL}(x) = \begin{cases} 0 & \text{ha } x \le p_i \ ; \\ \frac{x-p_i}{q_i-p_i} & \text{ha } p_i < x < q_i \ ; \\ 1 & \text{ha } q_i \le x \ , \end{cases}$$

ahol $0 \le p_i \le q_i \le \frac{1}{2}$, és

$$\tau^{PR}(x) = \begin{cases} 0 & \text{ha } x \le p_i \ ; \\ \frac{x-p_i}{q_i-p_i} & \text{ha } p_i < x < q_i \ ; \\ 1 & \text{ha } q_i \le x \ , \end{cases}$$

ahol $\frac{1}{2} \le p_i \le q_i \le 1$.

Felmerül a kérdés, hogy ebbe a családba beilleszthető-e a lexikografikus döntési eljárás? Jelen dolgozatban megmutatjuk, hogy ha w_i súlyokat illetve $\tau(x)$ függvényt speciálisan választjuk, akkor az általános modell a lexikografikus döntést is magában foglalja. Mivel a lexikografikus döntés nem kompenzatorikus, ezért egy nem folytonos $\tau(x)$ függggvényt kell alkalmazni:

$$\tau\left(\sum_{i=1}^{n} w_i \tau_i(p_i(x_i, y_i))\right)$$

Ez azonban nem csorbítja a modell értékét, mert a PROMETHEE és ELECTRE is beilleszthető ebbe az új modellbe, $\tau(x) = x$ alkalmazásával. Ennek azért van jelentősége, mert a konkrét x_i, y_i értékek ismerete nélkül, akármilyen kicsi is lehet a köztük levő eltérés. Ez azt jelenti, hogy eljárásunkat úgy konstruáljuk meg, hogy alkalmazható legyen on-line módon érkező adatok esetében is. Ekkor nem támaszkodhatunk arra, hogy a bejövő adatok között mekkora a legkisebb eltérés, ily módon előre adott súlyokkal kell dolgoznunk. Ha adataink között tetszőlegesen kis különbség lehet, felmerül a kérdés, hogy alkalmassá tehető-e a modell az indifferenciaküszöb figyelembevételére? Az említett általános modellben $\tau(x)$ módosításával ez elérhető, például:

$$\tau(x) = \begin{cases} 0 & \text{ha } 0 \le x < \frac{1}{2} - \delta; \\ 1 & \text{ha } \frac{1}{2} + \delta < x \le 1. \end{cases}$$

A véges lexikografikus rendezés általános koncepciójának alapja —követve Fishburn [3] terminológiáját— egy véges $I = \{1, 2, \dots, n\}$ halmaz, és egy $\prec_i$ rendezési reláció a nemüres X_i halmazon minden $i \in I$-re. Jelölje X_i elemeit x_{li}, ahol minden $i \in I$-re $l \in \{1, 2, \dots, m\}$. Jelölje $\sim_i$ a $\prec_i$ reláció szimmetrikus komplementerét, azaz bármely $x_{ji} \sim_i x_{ki}$ akkor és csak akkor, ha sem $x_{ji} \prec_i x_{ki}$, sem $x_{ki} \prec_i x_{ji}$ reláció nem teljesül. Ekkor $a_j = (x_{j1}, x_{j2}, \dots, x_{jn})$ és $a_k = (x_{k1}, x_{k2}, \dots, x_{kn})$-ra azt mondjuk, hogy a_k lexikografikusan megelőzi a_j-t az I-n értelmezett természetes $<$ rendezés mellett $\prec_i$ relációkra vonatkozóan, vagy jelölésben röviden: $a_j <^L a_k$ akkor és csak akkor, ha

$$\{i : i \in I \ \text{ és } \ (x_{ji} \prec_i x_{ki} \text{ vagy } x_{ki} \prec_i x_{ji})\}$$

halmaz nemüres, és $x_{ji} \prec_i x_{ki}$ az első (legkisebb) i-re ebben a halmazban. Pontosan ezért nevezhetjük a lexikografikus rendezést az első differencia alapján való rendezésnek is. A rendezések lexikografikus aggregációja során fontos a tulajdonságok öröklődésének vizsgálata, Solymosi [8]. Gyenge rendezések illetve lineáris rendezések aggregációja gyenge rendezés illetve lineáris rendezés. Parciális rendezések lexikografikus aggregációja viszont nem feltétlenül parciális rendezés, ciklus is kialakulhat, Fishburn [3].

A lexikografikus rendezésre egyik legismertebb példa az abc szerinti rendezés a szótárakban vagy lexikonokban. A lexikografikus rendezés általános koncepciójának megfelelően ekkor $I = \{1, 2, \dots, n\}$, ahol n a leghosszabb, a szótárban leírt szó hossza. Legyen $X_i = A = \{\emptyset, a, b, \dots, z\}$, $\emptyset \prec_i a \prec_i b \prec_i \dots \prec_i z$ mellett minden i-re. Egy $\alpha_1\alpha_2 \dots \alpha_r$ szó, ahol nyilván $r \le n$, az A^n halmaz egy $(\alpha_1, \alpha_2, \dots, \alpha_m, \emptyset, \dots, \emptyset)$ eleme lesz. Ekkor $<^L$ az A^n azon részhalmazain, amely a „valódi" szavakat jelenti, a szavak természetes abcbeli rendezését adja.

A többtényezős csoportos döntések elméletében a lexikografikus döntési eljárás az attribútumok vagy kritériumok egy hierarchiáján, vagy rendezett halmazán alapul, ahol az első különbség alapján való összehasonlítás elve azt mondja ki, hogy egy alternatíva „jobb" mint egy másik, akkor és csak akkor, ha az első „jobb" mint a második a legfontosabb kritériumon, amelyen különböznek. A bevezetőben említett koncepció szemantikus értelmezéseként legyen tehát a_j és a_k két alternatíva és $c_1, c_2, \dots, c_n$ különböző kritériumok, x_{ji} és x_{ki} pedig jelentsék az a_j és a_k alternatívák c_i kritérium szerinti hasznosságát (kiértékelését). Ekkor a_j és a_k alternatívákat azonosíthatjuk a kiértékelés vektoraikkal, azaz mondhatjuk, hogy $a_j = (x_{j1}, x_{j2}, \dots, x_{jn})$ és $a_k = (x_{k1}, x_{k2}, \dots, x_{kn})$.

Ekkor $\prec_i$a c_i kritérium által az alternatívákon megvalósított rendezési reláció. $x_{ji} \sim_i x_{ki}$ akkor és csak akkor, ha a_j és a_k indifferensek c_i kritérium szerint, és $x <^L y$ akkor és csak akkor, ha $x_{ji} \sim_i x_{ki}$ $i \in \{1, 2, \ldots, l-1\}$ és $x_{jl} \prec_i x_{kl}$.

A lexikografikus döntési szabály számos helyen hatékonyan alkalmazható, pályázatok kiértékelésében, Rapcsák [7], vagy szavazási rendszerekben a holtverseny elkerülésére. Ahogyan fentebb is említettük, dolgozatunkban a preferenciafüggvényt egy súlyozás segítségével valósítjuk meg. Itt a súlyokat úgy választottuk meg, hogy a nem kompenzatorikusság miatt egy kritérium által felállított sorrendet fontossági sorrendben utána következő kritériumok nem változtathatnak meg együttesen sem. Az alternatívahalmazról feltesszük, hogy nem létezik benne két lexikografikusan azonos elem. Vagyis, ha tetszőleges A alternatívahalmazon E jelöli azt az ekvivalenciarelációt, mely szerint $a_j E a_k$, ha a_j és a_k lexikografikusan azonos, akkor alternatívahalmaznak az A/E faktorhalmazt tekintjük.

2 A lexikografikus döntési függvény konstrukciója

A lexikografikus döntési módszerrel kevés publikáció foglalkozik, numerikus reprezentációjára pedig alig találunk példát. Ezt indokolhatják az erre vonatkozó negatív eredmények, például a sík lexikografikus rendezésére vonatkozó (de tetszőleges n-dimenziós lineáris térre is igaz):

Tétel. Nem létezik olyan folytonos $f(x, y)$ valós függvény, amelyre $(x, y) <^L (v, z) \Longleftrightarrow f(x, y) < f(v, z)$.

Bizonyítás. Legyen x, x_1, x_2, y_1, y_2 olyan, hogy $x_1 < x < x_2$, és $y_1 < y_2$. Tegyük fel, hogy van olyan $f(x, y)$ függvény, amelyre

$$(x, y) <^L (v, z) \Longleftrightarrow f(x, y) < f(v, z) \; .$$

A fentiekre ekkor igaz a következő:

$$(x, y_2) <^L (x_2, y_1) <^L (x_2, y_2) \Longleftrightarrow f(x, y_2) <^L f(x_2, y_1) <^L f(x_2, y_2) \; .$$

Mivel $f(x, y)$ (x_2, y_2)-ben is folytonos, így legyen ε tetszőleges olyan rögzített pozitív érték, hogy $\varepsilon < f(x_2, y_2) - f(x_2, y_1)$. Ekkor van olyan δ, hogy ha

$$|(x_2, y_2) - (x, y_2)| < \delta \implies f(x_2, y_2) - f(x, y_2) < \varepsilon \; .$$

Azonban legyen x tetszőleges, a definíció szerinti, akkor

$$f(x_2, y_2) - f(x, y_2) > f(x_2, y_2) - f(x_2, y_1) > \varepsilon \; ,$$

ami ellentmondás. Így ilyen folytonos függvény nem létezik. ∎

2.1 Preferenciafüggvény és módosító függvény

A bevezetésben bemutattuk a lexikografikus döntési elvet. Ebben a fejezetben megkonstruálunk egy lexikografikus döntési függvényt. A konstrukcióhoz általános $p(x, y)$ preferenciafüggvényt és $\tau(x)$ módosító vagy más néven vágófüggvényt, vagy küszöbérték függvényt használunk, melyek a következők:

$$p(x, y) = \frac{y - x + 1}{2}$$

$$\tau(x) = \begin{cases} 0 & \text{ha } 0 \le x < \frac{1}{2}\ ; \\ \frac{1}{2} & \text{ha } x = \frac{1}{2}\ ; \\ 1 & \text{ha } \frac{1}{2} < x \le 1\ . \end{cases}$$

Legyen $A = \{a_1, a_2, \cdots, a_m\}$ az alternatívák halmaza, $C = \{c_1, c_2, \cdots, c_n\}$ a kritériumok halmaza, fontossági sorrend szerint. Jelölje x_{ij} az a_i alternatíva c_j kritérium szerinti kiértékelését (hasznosságát), és tegyük fel, hogy a kiértékelés értékei normáltak, azaz $0 \le x_{ij} \le 1$. Egy döntési helyzetet ezután mátrix alakban, a döntési mátrixszal írhatunk fel,

	c_1	c_2	$\dots$	c_n
a_1	x_{11}	x_{12}	$\dots$	x_{1n}
a_2	x_{21}	x_{22}	$\dots$	x_{2n}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
a_m	x_{m1}	x_{m2}	$\dots$	x_{mn}

2.1.1 A preferenciafüggvény tulajdonságai

A bevezetőben említettek alapján az alternatívákat azonosíthatjuk a kiértékelés n-eseikkel, azaz legyen $a_i = (x_{i1}, x_{i2}, \dots, x_{in})$ és $a_j = (x_{j1}, x_{j2}, \dots, x_{jn})$. Ha a $p(x, y)$ preferenciafüggvényben az $x = x_{ik}$, $y = x_{jk}$ helyettesítéseket elvégezzük, megkapjuk a c_k kritérium által az (a_i, a_j) alternatívák között felállított preferencia-sorrendet. Ekkor

$$\begin{array}{ll} 0 \le p(x_{ik}, x_{jk}) < \frac{1}{2} & \text{ha } x_{ik} > x_{jk} \\ p(x_{ik}, x_{jk}) = \frac{1}{2} & \text{ha } x_{ik} = x_{jk} \\ \frac{1}{2} < p(x_{ik}, x_{jk}) \le 1 & \text{ha } x_{ik} < x_{jk} \end{array}$$

2.1.2 A preferenciafüggvény és a módosító függvény kompozíciója

1. Definíció. Minden (a_i, a_j) párra definiálhatjuk a $p^*(a_i, a_j)$ kritériumonkénti preferencia n-est a következő módon: $p^*(a_i, a_j) = (\varepsilon_1^{ij}, \varepsilon_2^{ij}, \dots, \varepsilon_n^{ij})$, $\varepsilon_k^{ij} = \tau(p(x_{ik}, x_{jk})$, amelyekre teljesül:

$$\tau(p(x_{ik}, x_{jk})) = \begin{cases} 0 & \text{ha } x_{ik} > x_{jk}\ ; \\ \frac{1}{2} & \text{ha } x_{ik} = x_{jk}\ ; \\ 1 & \text{ha } x_{ik} < x_{jk}\ . \end{cases}$$

Mivel az alternatívahalmazban nem létezik két lexikografikusan azonos elem, így minden (a_i, a_j) párra, $a_i = (x_{i1}, x_{i2}, \ldots, x_{in})$, $a_j = (x_{j1}, x_{j2}, \ldots, x_{jn})$, van olyan k_1 és k_2, hogy $x_{ik_1} < x_{jk_1}$ és $x_{jk_2} < x_{ik_2}$.

2.2 A lexikografikus döntési eljárás

A dolgozat legfontosabb eredményét az alábbiakban foglalhatjuk össze:

1. Tétel. Legyen $A = \{a_1, a_2, \cdots, a_m\}$ az alternatívák halmaza, $C = \{c_1, c_2, \cdots, c_n\}$ a kritériumok halmaza, a döntéshozó által megadott fontossági sorrend szerint. Jelölje x_{ij} az a_i alternatíva c_j kritérium szerinti kiértékelését (hasznosságát), és feltesszük, hogy a kiértékelés értékei normáltak, azaz $0 \leq x_{ij} \leq 1$. Azaz a döntési mátrix:

	c_1	c_2	$\ldots$	c_n
a_1	x_{11}	x_{12}	$\ldots$	x_{1n}
a_2	x_{21}	x_{22}	$\ldots$	x_{2n}
$\vdots$	$\vdots$	$\vdots$	$\vdots$	$\vdots$
a_m	x_{m1}	x_{m2}	$\ldots$	x_{mn}

A $p(x, y)$ preferenciafüggvény és $\tau(x)$ módosító vagy küszöbérték függvény legyen olyan, ahogyan azt 2.1 alatt megadtuk. Ekkor léteznek olyan w_k $k = 1, 2, \ldots, n$ súlyok, hogy az m elemű

$$l_i = \frac{1}{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} \tau \left(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk})) \right)$$

pozitív valós számhalmazra teljesül: $l_i > l_j$ pontosan akkor, ha $a_i <^L a_j$.

A lexikografikus döntési függvény konstrukciójához tehát a bevezetőben említett —a nem kompenzatorikusság elvét megtartó— súlyozás megvalósításával jutunk el. Ezt a következőként valósítjuk meg: Legyen a c_i kritérium fontossági súlya:

$$w_i = \frac{1}{2^i} + \frac{1}{n2^n} .$$

Ellenőrizhető, hogy teljesül a

$$\sum_{k=1}^{n} w_k = 1$$

feltétel. A lexikografikus döntési függvényt ekkor a következő függvénykompozíció valósítja meg:

$$\tau \left(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk})) \right) = \begin{cases} 0 & \text{ha } a_i >^L a_j \ ; \\ 1 & \text{ha } a_i <^L a_j \ . \end{cases}$$

Ennek segítségével definiálható valós számok egy $[0,1]$-re normált sorozata:

$$l_i = \frac{1}{n}\sum_{\substack{j=1\\ j\neq i}}^{n} \tau\left(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk}))\right)$$

melyre $l_i > l_j$ pontosan akkor, ha $a_i >^L a_j$.

Ezt a lexikografikus döntési sorrendet reprezentáló sorozatot oly módon konstruáltuk meg, hogy egy adott a_i alternatíva esetén aggregáltuk az a_i alternatíva és minden $j = 1, 2, \ldots, i-1, i+1, \ldots, n$-re az a_j alternatíva közötti preferenciát. Ezen a gondolaton alapszik a PROMETHEE módszer globális preferenciakonstrukciója is.

A konstrukció helyességének bizonyításához először a súlyozás helyességét látjuk be.

1. Lemma. Legyen $\varepsilon_k^{ij} = \tau(p(x_{ik}, x_{jk}))$. Ekkor igaz a következő két állítás:

(1) $\min\limits_{i,j} \sum\limits_{k=1}^{n} w_k \varepsilon_k^{ij} = \frac{1}{2} + \frac{1}{n2^n}$ ha $a_i <^L a_j$, és minimumát $(\varepsilon_1^{ij}, \varepsilon_2^{ij}, \ldots, \varepsilon_n^{ij}) = (1, 0, \ldots, 0)$ helyen veszi fel.

(2) $\max\limits_{i,j} \sum\limits_{k=1}^{n} w_k \varepsilon_k^{ij} = \frac{1}{2} - \frac{1}{n2^n}$ ha $a_i <^L a_j$, és maximumát $(\varepsilon_1^{ij}, \varepsilon_2^{ij}, \ldots, \varepsilon_n^{ij}) = (0, 1, \ldots, 1)$ helyen veszi fel.

Az 1. lemma bizonyítása: (1) Ha $a_i < a_j$ és $\varepsilon_k^{ij} = \tau(p(x_{ik}, x_{jk}))$, akkor az $(\varepsilon_1^{ij}, \varepsilon_2^{ij}, \ldots, \varepsilon_n^{ij})$ preferencia n-esek közül a következő alakú tartalmaz minimális számú nem 0 elemet:

$$\Big(\underbrace{\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}}_{k}, 1, 0, \ldots, 0\Big) \quad \text{ahol} \quad 0 \leq k < n\,.$$

Ekkor a súlyozott összeg értéke:

$$\sum_{k=1}^{n} w_k \varepsilon_k^{ij} = \frac{1}{2} + \left(\frac{k}{2} + 1\right)\frac{1}{n2^n}\,,$$

ami nyilvánvalóan akkor lesz minimális, ha $k = 0$. Ekkor

$$(\varepsilon_1^{ij}, \varepsilon_2^{ij}, \ldots, \varepsilon_n^{ij}) = (1, 0, \ldots, 0) \quad \text{és} \quad \min_{i,j}\sum_{k=1}^{n} w_k \varepsilon_k^{ij} = \frac{1}{2} + \frac{1}{n2^n}\,.$$

(2) Ha $a_i > a_j$, akkor az $(\varepsilon_1^{ij}, \varepsilon_2^{ij}, \ldots, \varepsilon_n^{ij})$ preferencia n-esek közül a következő alakú tartalmaz maximális nem 0 elemet:

$$\Big(\underbrace{\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}}_{k}, 0, 1, \ldots, 1\Big) \quad \text{ahol} \quad 0 \leq k < n\,.$$

Ekkor

$$\sum_{k=1}^{n} w_k \varepsilon_k^{ij} = \frac{1}{2} - \left(\frac{k}{2} + 1\right) \frac{1}{n2^n} \, ,$$

Ez a kifejezés maximumát $k = 0$ esetben veszi fel. Ekkor

$$(\varepsilon_1^{ij}, \varepsilon_2^{ij}, \ldots, \varepsilon_n^{ij}) = (0, 1, \ldots, 1) \quad \text{és} \quad \max_{i,j} \sum_{k=1}^{n} w_k \varepsilon_k^{ij} = \frac{1}{2} - \frac{1}{n2^n} \, .$$

Ezzel beláttuk a lemma állítását, így a súlyozás helyességét bizonyítottuk. ∎

A tétel bizonyítása. Ekkor a súlyozott összeg értékére igaz:

$$\frac{1}{2} < \sum_{k=1}^{n} w_k \varepsilon_k^{ij} \leq 1 \quad \text{ha } a_i <^L a_j \, ;$$
$$0 \leq \sum_{k=1}^{n} w_k \varepsilon_k^{ij} < \frac{1}{2} \quad \text{ha } a_i >^L a_j \, .$$

Ekkor erre a súlyozott összegre alkalmazva a τ módosító függvényt, kapjuk, hogy

$$\tau \left(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk})) \right) = \begin{cases} 0 & \text{ha } a_i >^L a_j \, ; \\ 1 & \text{ha } a_i <^L a_j \, , \end{cases}$$

így $\tau(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk})))$ a lexikografikus preferenciarendezést adja két alternatíva között. Ily módon ezzel a konstrukcióval egy lexikografikus döntési függvényt adtunk meg. Az ebben szereplő $\tau(x)$ küszöbérték függvény nem folytonos, így a lexikografikus döntési függvény sem az. A módosító függvény általános alakja:

$$\tau_{p_i q_i}(x) = \begin{cases} 0 & \text{ha } 0 \leq x < p_i \, ; \\ \frac{x - p_i}{q_i - p_i} & \text{ha } p_i < x < q_i \, ; \\ 1 & \text{ha } q_i < x \leq 1 \, . \end{cases}$$

Innen adódik, hogy ha p_i és q_i paraméterek értéke $\frac{1}{2}$-hez tart, akkor a lexikografikus döntési eljárást döntési eljárások határértékeként kapjuk meg. Véve

$$\sum_{\substack{j=1 \\ j \neq i}}^{n} \tau \left(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk})) \right)$$

összeget, azon a_j alternatívák számát adja meg, melyek a_i-nél lexikografikusan nagyobbak. [0,1]-re transzformálva ezeket az értékeket kapjuk

$$l_i = \frac{1}{n} \sum_{\substack{j=1 \\ j \neq i}}^{n} \tau \left(\sum_{k=1}^{n} w_k \tau(p(x_{ik}, x_{jk})) \right)$$

normált értékeket, melyekre teljesül, hogy

$$l_i > l_j \Longleftrightarrow a_i < a_j \ ,$$

azaz a kapott értékeken a valósakon értelmezett rendezés szerinti sorrend megegyezik az alternatívákon vett lexikografikus sorrenddel. Így a tétel állítását beláttuk. ∎

3 A lexikografikus döntési függvény tulajdonságai

Ebben a fejezetben megvizsgáljuk a lexikografikus döntési függvényt a legfontosabb és a döntési függvénytől leggyakrabban megkívánt tulajdonságok, a döntőség, a neutralitás, a monotonitás, a gyenge és erős Pareto optimalitás tulajdonságok teljesülése szempontjából Hwang, Lin [5].

Ahogyan fentebb írtuk, minden a_i, a_j alternatívapárhoz hozzá tudunk rendelni egy kritériumonkénti preferenciákból előálló preferencia n-est, amelyet jelöljön

$$\varepsilon^{ij} = (\varepsilon_1^{ij}, \varepsilon_2^{ij}, \ldots, \varepsilon_n^{ij}), \quad \varepsilon_k^{ij} = \tau(p(x_{ik}, x_{jk})) \ ,$$

és amelyekre teljesül:

$$\varepsilon_k^{ij} = \begin{cases} 0 & \text{ha } x_{ik} > x_{jk} \ ; \\ \frac{1}{2} & \text{ha } x_{ik} = x_{jk} \ ; \\ 1 & \text{ha } x_{ik} < x_{jk} \ , \end{cases}$$

így $\varepsilon_k^{ij} \in \{0, \frac{1}{2}, 1\}$. Legyen most $E = \{0, \frac{1}{2}, 1\}^n$, ekkor $\varepsilon^{ij} \in E$. Jelölje ekkor F az $(\varepsilon_1^{ij}, \varepsilon_2^{ij}, \ldots, \varepsilon_n^{ij}) \longmapsto \tau\left(\sum_{k=1}^n w_k \tau(p(x_{ik}, x_{jk}))\right)$ hozzárendelést. $\tau\left(\sum_{k=1}^n w_k \tau(p(x_{ik}, x_{jk}))\right)$ fentebbi tulajdonságai miatt a lexikografikus döntési függvény felírható ezzel a jelöléssel:

$$F(\varepsilon^{ij}) = \begin{cases} 0 & \text{ha } a_i >^L a_j \\ 1 & \text{ha } a_i <^L a_j \ . \end{cases}$$

3.1 A lexikografikus döntési függvény fontosabb tulajdonságai

3.1.1 Döntőség

Egy döntési függvény $F : E \to \{0, \frac{1}{2}, 1\}$ döntő, ha

$$\varepsilon^{ij} \neq (\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}) \implies F(\varepsilon^{ij}) \neq \frac{1}{2} \ ,$$

gyengén döntő, ha

$$\left\{\varepsilon^{ij} : F(\varepsilon^{ij}) = \frac{1}{2}\right\} = (\frac{1}{2}, \frac{1}{2}, \ldots, \frac{1}{2}) \ ,$$

szigorúan döntő, ha

$$\left\{\varepsilon^{ij} : F(\varepsilon^{ij}) = \frac{1}{2}\right\} = \emptyset \, .$$

3.1.2 Neutralitás

Egy döntési függvény neutrális, ha

$$f(1-\varepsilon_1^{ij}, 1-\varepsilon_2^{ij}, \ldots, 1-\varepsilon_n^{ij}) = 1 - f(\varepsilon_1^{ij}, \varepsilon_2^{ij}, \ldots, \varepsilon_n^{ij}) \, ,$$

azaz, ha kritériumonként komplementerére változik két alternatíva között a preferencia, akkor a döntési függvény értéke (vagyis az aggregált preferencia) is komplementerére változik.

3.1.3 Monotonitás (pozitív válaszadás)

Egy döntési függvény monoton, ha $\varepsilon^{ij} \geq^L \varepsilon^{kl}$, akkor $F(\varepsilon^{ij}) \geq F(\varepsilon^{kl})$.

3.1.4 Gyenge Pareto optimalitás (az egybehangzó vélemények érvényesülése)

Egy döntési függvény teljesíti a gyenge Pareto optimalitás tulajdonságát, ha bármely (a_i, a_j) alternatívapárra teljesül, hogy ha

$$\forall k \; \varepsilon_k^{ij} = 1 \Longrightarrow F(\varepsilon_k^{ij}) = 1, \quad \text{vagy ha } \; \forall k \; \varepsilon_k^{ij} = 0 \Longrightarrow F(\varepsilon_k^{ij}) = 0 \, .$$

3.1.5 Szigorú Pareto optimalitás

Egy döntési függvény teljesíti a szigorú Pareto optimalitás tulajdonságát, ha

$$\forall k : \varepsilon_k^{ij} \in \left\{\frac{1}{2}, 1\right\} \text{ és } \exists l : \varepsilon_l^{ij} = 1 \Longrightarrow F(\varepsilon^{ij}) = 1 \, ,$$
$$\text{és ha } \; \forall k : \varepsilon_k^{ij} = \frac{1}{2} \Longrightarrow F(\varepsilon^{ij}) = \frac{1}{2} \, .$$

2. Lemma. A lexikografikus döntési függvény teljesíti a következő tulajdonságokat:

1. szigorú döntőség
2. neutralitás
3. monotonitás
4. gyenge Pareto optimalitás
5. szigorú Pareto optimalitás.

Bizonyítás. 1. Döntőség. Mivel feltettük, hogy alternatíváink Pareto optimális halmazt alkotnak, így minden a_i, a_j párra van olyan k egész, hogy

$$x_{ik} \neq x_{jk} \, , \quad \text{így} \quad \{\varepsilon^{ij} : F(\varepsilon^{ij}) = \frac{1}{2}\} = \emptyset \, ,$$

ezért kapjuk, hogy lexikografikus döntési függvény szigorúan döntő.

2. Neutralitás. A lexikografikus döntési függvénynél ez azt jelenti, hogy ha kritériumonként megfordul két alternatíva között a preferenciairány, akkor az két alternatíva közötti preferencia is ellentettjére fordul. Ez pedig a lexikografikus döntés szabályai miatt belátható, hogy igaz. Vagyis a lexikografikus döntési függvény teljesíti a neutralitás feltételeit, ha $\varepsilon_k^{ij} = \tau(p(x_{ik}, x_{jk}))$-ra:

$$\tau\left(\sum_{k=1}^{n} w_k \varepsilon_k^{ij}\right) = 1 - \tau\left(\sum_{k=1}^{n} w_k (1 - \varepsilon_k^{ij})\right) .$$

Mivel

$$\sum_{k=1}^{n} w_k (1 - \varepsilon_k^{ij}) = \sum_{k=1}^{n} w_k - \sum_{k=1}^{n} w_k \varepsilon_k^{ij} = 1 - \sum_{k=1}^{n} w_k \varepsilon_k^{ij} ,$$

és $\tau(x)$-re definíciója miatt teljesül a

$$\tau(\alpha) = 1 - \tau(1 - \alpha)$$

függvényegyenlet, így a lexikografikus döntési függvény neutrális.

3. Monotonitás (pozitív válaszadás). Legyen ε^{ij} az a_i és a_j, ε^{kl} pedig az a_k és a_l alternatívákhoz rendelt kritériumonkénti preferenciák n-ese. Ha $a_i >^L a_j$, akkor $F(\varepsilon^{ij}) = 1$, így $F(\varepsilon^{ij}) \geq F(\varepsilon^{kl})$ tetszőleges ε^{kl} esetén. Ha $a_i <^L a_j$, akkor $F(\varepsilon^{ij}) = 0$. Ekkor

$$\varepsilon^{ij} = (\underbrace{\frac{1}{2}, \ldots, \frac{1}{2}}_{k}, 0, \ldots) \quad \text{ahol} \quad 0 \leq k < n ,$$

így, ha $\varepsilon^{ij} \geq^L \varepsilon^{kl}$, akkor ε^{kl} olyan, hogy ha egy kritérium szerinti preferencia értéke ε^{ij}-ben $\frac{1}{2}$ akkor ugyanezen kritérium szerint ε^{kl}-ben 0 van, vagy pedig

$$\varepsilon^{kl} = (\underbrace{\frac{1}{2}, \ldots, \frac{1}{2}}_{k}, 0, \ldots) \quad \text{ahol} \quad 0 \leq k < n .$$

Ekkor mindkét esetben $F(\varepsilon^{kl}) = 0$, így a monotonitás teljesül.

4. Gyenge Pareto optimalitás. Mivel a módosító függvényre definíciója miatt teljesül, hogy

$$\tau\left(\sum_{k=1}^{n} w_k \cdot 1\right) = \tau\left(\sum_{k=1}^{n} w_k\right) = \tau(1) = 1 ,$$

így a gyenge Pareto optimalitás feltétel második fele teljesül a lexikografikus döntési függvényre, azaz

$$\tau\left(\sum_{k=1}^{n} w_k \cdot \frac{1}{2}\right) = \tau\left(\frac{1}{2} \cdot \sum_{k=1}^{n} w_k\right) = \tau\left(\frac{1}{2}\right) = \frac{1}{2} ,$$

azonban dolgozatunkban feltettük, hogy az alternatívák között nincs két lexikografikusan azonos, emiatt így a gyenge Pareto optimalitás teljesül.

5. Szigorú Pareto optimalitás. A szigorú Pareto otpimalitás feltétele teljesül, ha

$$\forall k:\ \varepsilon_k^{ij} \in \{\frac{1}{2}, 1\} \ \text{ és } \ \exists l : \varepsilon_l^{ij} = 1 \Longrightarrow F(\varepsilon^{ij}) = 1\,.$$

A feltétel második felére ugyanaz igaz, amit az előbbi pontban megfogalmaztunk, azaz, ha $\forall k : \varepsilon_k^{ij} = \frac{1}{2}$, akkor $F(\varepsilon^{ij}) = \frac{1}{2}$ valóban teljesül, ebben a dolgozatban viszont lexikografikus azonosságot az alternatívák között kizártuk, így $\forall k : \varepsilon_k^{ij} = \frac{1}{2}$ nem fordulhat elő. Így a lexikografikus döntési függvény szigorúan Pareto optimális. Így a lemma állítását bizonyítottuk. ∎

Köszönetnyilvánítás

Köszönettel tartozunk dolgozatunk lektorának, aki értékes észrevételeivel lehetővé tette a közölt döntési eljárás gondolatmenetének pontos kifejtését, rámutatva a modell lényeges és részletesebb kifejtést igénylő elemeire, mint amilyen például az adatok folytonossága, vagy az általános keretrendszerben a paraméterek választásának kérdése.

Irodalom

1. Dombi J. A general framework for the utility based outranking methods. *Fuzzy Logic and Soft Computing, Advances in fuzzy systems – applications and theory*, Ed. B. Bouchon-Meunier, R. R. Yager, R. A. Zadeh, 1995
2. Dombi J. A Common Preference Model for Various Decision Models. *Principles of Fuzzy Preference Modelling and Decision Making*, Ed. B. De Baets, J. Fodor. Academia Press, Gent, 2003
3. Fishburn P. C. Lexicographic orders, utilities and decision rules: a survey. *Management Science* 11 (1974) 1442–71
4. Fishburn P. C. *Nonlinear preference and utility theory*, Wheatsheaf Books Ltd., Brighton, 1988.
5. Hwang C. L., Lin M. J. *Group Decision Making under Multiple Criteria*, Springer Verlag, Berlin, Heidelberg, 1987.
6. Olson D. L. Decision Aids for Selection Problems, Springer Verlag, New York, 1996.
7. Rapcsák T. *Többszempontú döntési problémák.* MTA SZTAKI, 2003.
8. Solymosi T., Ordinális mérési skálák lexikografikus aggregációja. *Adat – modell – elemzés.* Kovács E. (szerk.), Aula, Budapest, 2001. 119–126.
9. Temesi J. *A döntéselmélet alapjai.* Aula, 2002.

LEXICOGRAPHIC DECISION MODEL IN A GENERAL FRAMEWORK

In this paper the lexicographic decision process is presented in a unified way. We construct a lexicographic decision function using a universal preference function and a unary function. This construction incorporates the different outranking approaches, the lexicographic decision process and the utility based decision making models. Finally we show some properties of the lexicographic decision function.

A CRISS-CROSS ALGORITMUS ÚJ VÁLTOZATAI LINEÁRIS KOMPLEMENTARITÁSI FELADATOKRA[1]

CSIZMADIA ZSOLT – ILLÉS TIBOR
ELTE Operációkutatási Tanszék

Új típusú criss-cross módszereket általánosítunk elégséges mátrixú lineáris komplementaritási feladatokra (LCP). A legtöbb LCP megoldó algoritmus előre feltételez bizonyos tulajdonságokat a feladat mátrixáról. Egy mátrix elégségessége nehezen ellenőrizhető tulajdonság (nem ismert rá polinomiális eljárás). Algoritmusunk Zhang lineáris programozási illetve Akkeleş-Balogh-Illés LCP-QP feladatra adott criss-cross típusú algoritmusával rokon. A mi algoritmusunk abban tér el a lineáris komplementaritási feladatokat megoldó korábbi módszerektől, hogy számunkra nem szükséges a priori információ a mátrix tulajdonságairól. Algoritmusunk leállási kritériumai: megoldja az LCP feladatot, megoldja az LCP feladat duálját illetve kijelzi azt, hogy a feladat mátrixa nem elégséges és ezért ciklizálásra kerül(het)ne sor. Annak ellenére, hogy algoritmusunk általánosabb feltételek mellett dolgozik, mint Akkeleşék módszere, mégis sikerült az algoritmus végességét egyszerűbben bizonyítani. Az algoritmus végessége egyben új, konstruktív bizonyítást jelent a Fukuda és Terlaky által LCP dualitás tételnek nevezett eredményre.

Kulcsszavak: Lineáris komplementaritási feladat, elégséges mátrixok, criss-cross típusú algoritmus, alternatíva és EP–tételek.

Mathematics Subject Classification 2000: 49M35, 90C20.

1 Bevezetés

A *lineáris komplementaritási feladat*ot (LCP) a következő módon fogalmazhatjuk meg: keresünk olyan $\mathbf{u}, \mathbf{v} \in I\!R^n$ vektorokat, amelyekre

$$-M\mathbf{u} + \mathbf{v} = \mathbf{q}, \quad \mathbf{u}\,\mathbf{v} = \mathbf{0}, \quad \mathbf{u},\ \mathbf{v} \geq \mathbf{0}\ , \tag{1}$$

ahol $M \in I\!R^{n\times n}$, $\mathbf{q} \in I\!R^n$ és $\mathbf{u}\,\mathbf{v} = (u_1v_1, \ldots, u_nv_n)$.

A lineáris komplementaritási feladat a matematikai programozás egyik legtöbbet kutatott területe. Kiterjedt a gyakorlati felhasználása, sokrétű elméleti problémái, algoritmusainak hatékonysága és egyszerűsége mind-mind olyan tényező, amely sok kutató számára vonzóvá teszi ezt a területet.

A lineáris komplementaritási feladat NP-teljes, sőt még akkor is NP-teljes marad, ha az M mátrixot megszorítjuk a negatív szemidefinit mátrixok

[1] Beérkezett: 2005. szeptember 8. E-mail: `csisza@math.elte.hu`, `illes@math.elte.hu`.

osztályára [4], vagy a P_0 mátrixok osztályára [14]. A P_0 mátrixok osztálynak része az ún. *elégséges mátrix*ok osztálya.

Az LCP feladat megoldására többféle pivot típusú algoritmus létezik. A criss–cross algoritmust egymástól függetlenül —de különböző optimalizálási feladatra közel egyidőben— fejlesztették ki Chang, Terlaky illetve Wang. Azóta a criss–cross módszer alatt algoritmusok egy családját értjük, amelyek az indexválasztási szabályban különböznek egymástól, és közös bennük az, hogy primál-duál típusú algoritmusok, a bázisok fizibilitását nem követelik meg, és az indexválasztás biztosítja az algoritmusok végességét.

Lineáris komplementaritási feladat adódik kvadratikus programozási feladat Karush–Kuhn–Tucker feltételeiből is. Konvex kvadratikus célfüggvény esetén az LCP feladat M mátrixa biszimmetrikus. Ilyen LCP feladatra fogalmazták meg új típusú criss–cross algoritmusaikat Akkeleş, Balogh és Illés [1]. Az általuk alkalmazott indexválasztási szabály a *LIFO* (utoljára bázisba bekerült változó elsőnek távozik onnan) illetve a *leggyakrabban mozgott változó* kiválasztási szabálya. Érdekes kérdés az, hogy melyik a legbővebb mátrixosztály, amelyre a criss–cross típusú algoritmus az előbbiekben említett pivotálási szabállyal általánosítható úgy, hogy ciklizálás ne lépjen fel.

Az elégséges mátrixok fogalmát először Cottle, Pang és Venkateswaran [7] vezették be. Az elégséges mátrixokat a P és PSD mátrixok általánosításainak foghatjuk fel. Väliaho [20] megmutatta, hogy az elégséges mátrixok valójában megegyeznek a P_* mátrixokkal. Hertog, Roos és Terlaky [10] bizonyították, hogy az elégséges mátrixok éppen azon mátrixok, amelyekre a minimálindexes criss–cross algoritmus tetszőleges jobb oldal esetén megoldja az LCP feladatot.

Első célunk, a LIFO és a leggyakrabban mozgott változó pivot szabállyal ellátott criss–cross algoritmust elégséges mátrixú LCP feladatokra általánosítani. Az általánosítás mellett az [1] cikkben közölt végesség bizonyítást is leegyszerűsítjük. Ezen választási szabályok egy előnye, hogy elsősorban az algoritmus elején jelentős szabadságot biztosítanak a változó kiválasztása terén, így gyakran lehetőséget kínálnak a numerikusan instabil báziscserék elkerülésére is.

Jelenleg nem ismeretes hatékony algoritmus annak eldöntésére, hogy egy adott mátrix elégséges-e. (Väliaho [19] kifejlesztett egy induktív módszert elégségesség ellenőrzésére, de a módszer nem polinomiális.) A korábban elégséges mátrixú LCP feladatokra adott algoritmusok gyakorlati szempontból hátrányos tulajdonsága volt, hogy előre kellett ismerniük az M mátrix elégségességét. Fukuda, Namiki és Tamura [8] adtak először olyan algoritmust, amely Fukuda és Terlaky [9] LCP dualitás tételének EP formában való megfogalmazására épült, és nem igényelt előzetes információt a mátrix elégségességéről. Ha az algoritmus ennek hiánya miatt nem tudott tovább lépni, vagy ciklizálni kezdett, akkor a mátrix nem elégséges voltának polinomiális méretű bizonyítékát szolgáltatta.

Második célunk az elégséges mátrixokra általánosított algoritmust úgy kiegészíteni, hogy tetszőleges M mátrix és $\mathbf{q}$ jobb oldal vektor esetén vagy megoldja a feladatot, vagy pedig polinomiális méretű bizonyítékát adja annak, hogy az M mátrix nem elégséges. Ez jelentősen megnöveli az algoritmus értékét, hiszen az LCP feladatok igen széles körére alkalmazhatóvá teszi az algoritmust anélkül, hogy a mátrix tulajdonságairól a priori információra lenne szükségünk. Sőt, a pivot pozíció megválasztásának az eredeti minimál indexes criss–cross algoritmushoz képest megnövekedett szabadsága számos esetben lehetővé teszi numerikusan instabil pivotálások elkerülését, növelve a criss–cross módszer gyakorlati alkalmazhatóságát.

Az így módosított algoritmus végességének a bizonyítása egyben új, konstruktív bizonyítást jelent a lineáris komplementaritási feladat dualitás tételének az erősebb, EP–tétel alakjában megfogalmazott változatára is.

Végezetül térjünk ki a cikkünkben használatos jelölésekre:

$\mathbf{x}, x_i$	vektorokat vastag betű, skalárokat normál betű jelzi
$\mathbf{v}\,\mathbf{u}$	a $\mathbf{v}$ és az $\mathbf{u}$ vektorok koordinátánkénti (Hadamard) szorzata
M	az LCP feladat együtthatómátrixa, $M \in I\!R^{n\times n}$
B	bázis, a $[-M, I] \in I\!R^{2n\times 2n}$ egy $n\times n$-es nemszinguláris részmátrixa
$\bar{M}$	egy adott bázishoz tartozó rövid pivot tábla mátrixa
M_B	egy adott bázishoz tartozó rövid pivot tábla mátrixa, ha célszerű kiemelni, hogy a B bázishoz tartozik
$\bar{\mathbf{m}}_j$	az $\bar{M}$ mátrix j-edik oszlopvektora
$\bar{\imath}$	$= n+i$ ha $i \in \{1,\ldots,n\}$ és $= i-n$ ha $i \in \{n+1,\ldots,2n\}$
J_B	a B bázishoz tartozó változók indexhalmaza
$J_N := \overline{J_B}$	a B bázison kívüli változók indexhalmaza
$\oplus, \ominus, +, -$	nemnegatív, nempozitív, pozitív ill. negatív elemet jelöl
$< \ldots >$	jelöli a generált alteret

2 Elégséges mátrixok

A lineáris komplementaritási feladatok vizsgálatát és megoldási módszereinek a hatékonyságát jelentősen befolyásolják az M mátrix tulajdonságai. A következőkben áttekintjük az LCP feladatok megfogalmazásakor használatos, fontosabb mátrixosztályokat.

Szükségünk lesz a következő, technikai jellegű definícióra.

2.1 Definíció. *Az $M \in I\!R^{n\times n}$ mátrix egy $\alpha = \{\alpha_1,\ldots,\alpha_k\} \subseteq \{1,\ldots,n\}$ indexhalmazhoz tartozó négyzetes részmátrixát, az $M_{\alpha\alpha}$ mátrixot,* diagonális menti négyzetes részmátrix*nak nevezzük.*

Következzen az *elégséges mátrix*ok [7] értelmezése.

2.2 Definíció. *Egy $M \in I\!R^{n\times n}$ mátrixot* oszlop elégséges*nek nevezünk, ha nem létezik $\mathbf{x} \in I\!R^n$ vektor, amelyre*

$$\left\{ \begin{array}{ll} x_i\,(M\mathbf{x})_i \le 0 & \textit{minden } i \in \{1,\ldots,n\} \textit{ indexre} \\ x_j\,(M\mathbf{x})_j < 0 & \textit{valamely } j \in \{1,\ldots,n\} \textit{ indexre} \end{array} \right. \quad (2)$$

és sor elégséges*nek nevezzük, ha a transzponáltja oszlop elégséges. Az M mátrix* elégséges mátrix*, ha egyszerre oszlop és sor elégséges is.*

Megmutatható, hogy az oszlop elégséges mátrixok pontosan azok a mátrixok, melyekre az LCP megoldáshalmaza konvex [7].

Az elégséges mátrixokat először Cottle és társszerzői [7] vezették be. Megmutatták, hogy ezek a P-mátrixok és a pozitív szemidefinit mátrixok általánosításai. Az elégséges mátrixokról azt is megmutatták, hogy speciális struktúrájú P_0-mátrixok.

Később Hertog és társszerzői [10] igazolták, hogy az elégséges mátrixok éppen azok a mátrixok, melyekre a szokásos minimálindex szabályú criss–cross módszer bármely jobb oldali $\mathbf{q}$ vektor esetén véges lépésben vagy talál egy megoldást, vagy kimutatja, hogy az LCP nem megoldható.

Az elégséges mátrixok néhány fontos tulajdonságának a bemutatásához szükségünk lesz a *szigorúan előjelfordító*, illetve *szigorúan előjeltartó* vektorok fogalmára, [8].

2.3 Definíció. *Egy* $\mathbf{x} \in I\!R^{2n}$ *vektort* szigorúan előjelfordító*nak nevezünk, ha*

$$\begin{array}{ll} x_i x_{\bar{\imath}} \leq 0 & \textit{minden } \; i = 1, \ldots, n \; \textit{indexre} \\ x_i x_{\bar{\imath}} < 0 & \textit{valamely } \; i \in \{1, \ldots, n\} \; \textit{indexre.} \end{array}$$

Egy $\mathbf{x} \in I\!R^{2n}$ *vektort* szigorúan előjeltartó*nak nevezünk, ha*

$$\begin{array}{ll} x_i x_{\bar{\imath}} \geq 0 & \textit{minden } \; i = 1, \ldots, n \; \textit{indexre} \\ x_i x_{\bar{\imath}} > 0 & \textit{valamely } \; i \in \{1, \ldots, n\} \; \textit{indexre.} \end{array}$$

Vezessük be a

$$V := \big\{ (\mathbf{u}, \mathbf{v}) \in I\!R^{2n} \mid [-M, I](\mathbf{u}, \mathbf{v}) = \mathbf{0} \big\}$$

illetve a

$$V^{\perp} := \big\{ (\mathbf{x}, \mathbf{y}) \in I\!R^{2n} \mid [I, M^T](\mathbf{x}, \mathbf{y}) = \mathbf{0} \big\}$$

altereket. Nyilvánvaló, hogy a V illetve $V^{\perp}$ $I\!R^{2n}$ alterek egymás ortogonális kiegészítő alterei.

2.1 Lemma. *Egy $M \in I\!R^{n \times n}$ mátrixot pontosan akkor nevezünk elégséges mátrixnak, ha a V altérben nincs szigorúan előjelfordító vektor, illetve a $V^{\perp}$ altérben nincs szigorúan előjeltartó vektor.*

Vezessük be a *rövid pivot tábla* fogalmát, amely lehetővé teszi a későbbiek során bizonyításaink egyszerű szemléltetését.

2.4 Definíció. *Legyen adott egy véges S vektorhalmaz, melynek tagjait a J halmazzal indexeltük. Legyen továbbá a $J_B \subseteq J$ olyan, hogy az elemeivel indexelt vektorok az S egy bázisát alkotják. Jelölje $J_N = J \setminus J_B$ a nem bázisbeli vektorok indexhalmazát. Ekkor a J_B bázishoz tartozó* rövid pivot tábla *az $M \in I\!R^{|J_B| \times |J_N|}$ mátrix, amelyre m_{ij} a $j \in J_N$ index által jelölt vektor J_B szerinti előállításának $i \in J_B$ index által jelölt bázisvektor együtthatóját adja meg.*

A következő lemmára a mátrixok előjelszerkezetének vizsgálatakor lesz szükségünk, mely előjelszerkezet valójában az elégséges mátrixok általunk felhasznált minden tulajdonságát tartalmazza.

2.2 Lemma. (Cottle, Pang és Venkateswaran [7].) *Legyen M elégséges mátrix, B bázis,*

$$\bar{M} = [\bar{m}_{ij} : i \in J_B, j \in J_N]$$

a hozzá tartozó rövid pivot tábla. Ekkor a következő állítások teljesülnek:

(a) $\bar{m}_{i\bar{\imath}} \geq 0$ *minden* $i \in J_B$ *indexre, továbbá*

(b) minden $i \in J_B$ *indexre, ha* $\bar{m}_{i\bar{\imath}} = 0$ *akkor* $\bar{m}_{i\bar{\jmath}} = \bar{m}_{j\bar{\imath}} = 0$ *vagy* $\bar{m}_{i\bar{\jmath}} \cdot \bar{m}_{j\bar{\imath}} < 0$ *minden* $j \in J_B$, $j \neq i$ *esetén.*

Meg kell említeni, hogy a lemma bizonyítása konstruktív, tehát ha valahol sérül a kívánt előjelszerkezet, akkor az $\bar{M}$ táblájából könnyen kiolvasható a bizonyíték, hogy M nem elégséges.

Egy $M \in I\!R^{n\times n}$ mátrix *átrendezésén* a $P^T MP$ mátrixot értjük, ahol P egy permutációmátrix. A következő eredmények bizonyítása megtalálható Cottle [5] dolgozatában.

2.3 Lemma. *Legyen $M \in I\!R^{n\times n}$ egy tetszőleges sor (oszlop) elégséges mátrix. Ekkor az M mátrix*

1. *tetszőleges átrendezése is sor (oszlop) elégséges,*

2. *DMD alakú szorzata is sor (oszlop) elégséges, ahol $D \in I\!R^n$ diagonális mátrix,*

3. *tetszőleges diagonális menti négyzetes részmátrixa is sor (oszlop) elégséges.*

Könnyen igazolható, hogy ha az M elégséges, akkor belőle tetszőleges pivot sorozat után kapott $\bar{M}$ mátrix is az, vagyis az elégséges mátrixok osztálya pivot műveletre zárt, így a criss–cross típusú algoritmusok során a tábla elégségessége megőrződik.

Egy $M \in I\!R^{n\times n}$ mátrix és $\alpha \subseteq \{1, \ldots, n\}$ esetén ha $M_{\alpha\alpha}$ nemszinguláris, akkor az α indexhalmazhoz tartozó blokkpivotálási műveletet (mely során az α indexű változókon egyszerre végzünk báziscserét [15]) jelölje η_α.

2.4 Lemma. *Legyen $M_{\alpha\alpha}$ az M oszlop (sor) elégséges mátrix egy nemszinguláris, diagonális menti négyzetes részmátrixa. Ekkor az $M' = \eta_\alpha(M)$ is oszlop (sor) elégséges [5].*

Tehát az elégséges mátrixok osztálya a blokkpivotálásra nézve is zárt.

3 Lineáris komplementaritási feladatok alternatíva tétele

Annak eldöntése, hogy egy tetszőleges LCP feladatnak van-e megoldása, NP-beli feladat, és nem feltétlen *co*-NP-beli, de bizonyos mátrixosztályokra az.

Ilyen mátrixosztály az elégséges mátrixok osztálya is. Fogalmazzuk át az (1) feladatot egy kissé, ehhez definiáljuk a

$$\begin{aligned} V(M,\mathbf{q}) &= \left\{(\mathbf{u},\mathbf{v}) \in I\!R^{2n} : -M\mathbf{u}+\mathbf{v}=\mathbf{q}\right\} \\ V(M,\mathbf{q})^{\perp} &= \left\{(\mathbf{x},\mathbf{y}) \in I\!R^{2n} : \mathbf{x}+M^T\mathbf{y}=\mathbf{0},\, \mathbf{q}^T\mathbf{y}=-1\right\} \end{aligned}$$

affin altereket. Az (1) lineáris komplementaritási feladatot ekkor a következő alakban fogalmazhatjuk meg,

$(P-LCP)$: keresendő az $(\mathbf{u},\mathbf{v}) \in V(M,\mathbf{q}) \cap I\!R^{2n}_{\oplus}$, melyre $\mathbf{u}\cdot\mathbf{v}=\mathbf{0}$ teljesül.

Az optimalizálás elméletben felmerülő igen természetes kérdés, hogy ha a $(P-LCP)$ feladatnak nincsen megoldása, akkor az adataival megadható-e egy olyan feladat, amelynek van megoldása.

Fukuda és Terlaky [9] válaszolta meg ezt a kérdést egy igen általános formában, irányított matroidokon definiált lineáris komplementaritási problémák esetén. Fukuda és Terlaky eljárását követve definiáljuk a $(D-LCP)$ feladatot.

$(D-LCP)$: keresendő az $(\mathbf{x},\mathbf{y}) \in V(M,\mathbf{q})^{\perp} \cap I\!R^{2n}_{\oplus}$, melyre $\mathbf{x}\cdot\mathbf{y}=\mathbf{0}$ teljesül.

Az alábbiakban megmutatjuk, hogy az így definiált rendszerek közül legfeljebb az egyik oldható meg.

3.1 Lemma. *Bármely $M \in I\!R^{n\times n}$ elégséges mátrix és $\mathbf{q} \in I\!R^n$ vektor esetén legfeljebb az egyik teljesül:*

(1) az LCP feladatnak van $(\mathbf{u},\mathbf{v})$ megengedett komplementáris megoldása,

(2) a D-LCP feladatnak van $(\mathbf{x},\mathbf{y})$ megengedett komplementáris megoldása.

Bizonyítás. Tegyük fel, hogy mindkettő megoldható, és legyen $(\mathbf{u},\mathbf{v})$ a $(P-LCP)$ és az $(\mathbf{x},\mathbf{y})$ a $(D-LCP)$ feladat egy-egy megoldása. Ekkor a

$$-M\mathbf{u}+\mathbf{v}=\mathbf{q}$$

feltételből az $\mathbf{y}^T$ vektorral balról való szorzás után

$$-\mathbf{y}^T M\mathbf{u}+\mathbf{y}^T\mathbf{v}=\mathbf{y}^T\mathbf{q}=-1$$

adódik. A $(D-LCP)$ első feltételét használva, az egyenlet bal oldalát átalakítva, és a változók nemnegativitását is figyelembe véve

$$0 \le \mathbf{x}^T\mathbf{u}+\mathbf{y}^T\mathbf{v}=-1$$

összefüggést kapjuk, ami ellentmondás. ∎

3.1 Megjegyzés. *3.1 fejezetben az ott bevezetésre kerülő új típusú criss–cross algoritmus segítségével megmutatjuk, hogy a $(P-LCP)$ és $(D-LCP)$*

feladatok közül valamelyik mindig megoldható (3.2 Tétel), ha az M mátrix elégséges.

Ez egyben azt is jelenti, hogy ha az M mátrixunk elégséges és racionális, akkor a $(P-LCP)$ feladat nem megoldhatósága jól jellemzett, és polinomiális méretű bizonyíték adható rá, éspedig a $(D-LCP)$ feladat egy megoldása.

A 3.1 lemma általánosításával a 4. fejezetben foglalkozunk.

3.1 A criss-cross típusú algoritmus

Elsőként Akkeleş, Balogh és Illés [1] algoritmusának általánosításával foglalkozunk elégséges mátrixokra. Jelölje $\mathcal{I} := \{u_1, u_2, ..., u_n, v_1, v_2, ..., v_n\} \cup \{q\}$ a változók halmazát, míg $I := \{1, 2, ..., n, \bar{1}, \bar{2}, ..., \bar{n}\} \cup \{q\}$ a megfelelő indexhalmazt, ahol $|\mathcal{I}| = |I| = 2n+1$. A jelölés egyszerűsítésének érdekében legyen $\bar{\bar{\alpha}} = \alpha$ minden $\alpha \in I \backslash \{q\}$ indexre, vagyis az $\bar{\alpha}$ komplementáris párja az α. Felhívjuk a figyelmet azonban, hogy a rövid pivot táblát minden esetben az $\{1, \ldots, n\}$ koordinátákkal indexeljük értelemszerűen.

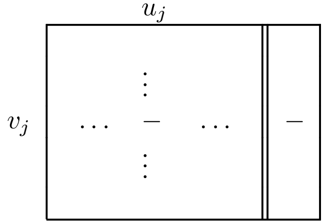
A lineáris komplementaritási feladat, (1), kiinduló, komplementáris megoldása az $\mathbf{u} = \mathbf{0}$, $\mathbf{v} = \mathbf{q}$. A pivot tábla mátrixát $\bar{M}$ jelöli.

A kezdeti komplementáris megoldásból, pivotálással, megengedett komplementáris megoldás előállítása a célunk. A 2.3 és a 2.4 lemmák biztosítják, hogy a feladat mátrixának az elégségessége a pivotálások során megőrződik.

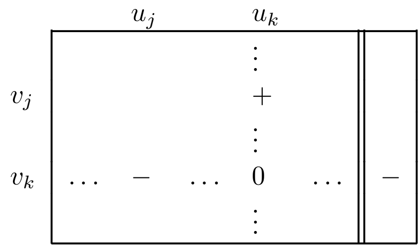
Kétfajta, diagonális és felcserélős pivot műveletet fogunk végezni, melyek mindegyike megőrzi az aktuális megoldás komplementáris voltát.

Legyen az algoritmus egy lépésében a v_j változó értéke nem megengedett. Ha $\bar{m}_{jj} < 0$, akkor *diagonális pivot*ot végzünk, mely során u_j belép a bázisba, míg v_j elhagyja azt.

Ha azonban $\bar{m}_{jj} = 0$, akkor olyan k indexen kell pivotálni, melyre $\bar{m}_{jk} < 0$. Az így kapott megoldás azonban nem lesz komplementáris, így ennek visszaállítására pivotálni kell a (k, j) pozíción. A 2.2 lemma alapján $\bar{m}_{kj} > 0$. A két pivotot együtt *felcserélős pivot*nak nevezzük.



Diagonális pivot



Felcserélős pivot

A felcserélős pivot ábrája szerinti helyzetben u_j és u_k belép a bázisba, míg v_j és v_k elhagyja azt. Azt mondjuk, hogy ekkor az u_k, v_k változókat *aktívan*, míg az u_j, v_j változókat *passzívan* választottuk ki.

A LIFO pivotálási szabályt az [1] cikkben a szerzők egy $\mathbf{s}_r : I \to \mathbb{N}_0^{2N}$

számláló vektor segítségével kezelik:

$$\mathbf{s}_r(\alpha) = \begin{cases} r, & \text{ha az } \alpha \in I \text{ indexű változó mozog az } r\text{-edik iterációban} \\ \mathbf{s}_{r-1}(\alpha), & \text{egyébként.} \end{cases}$$

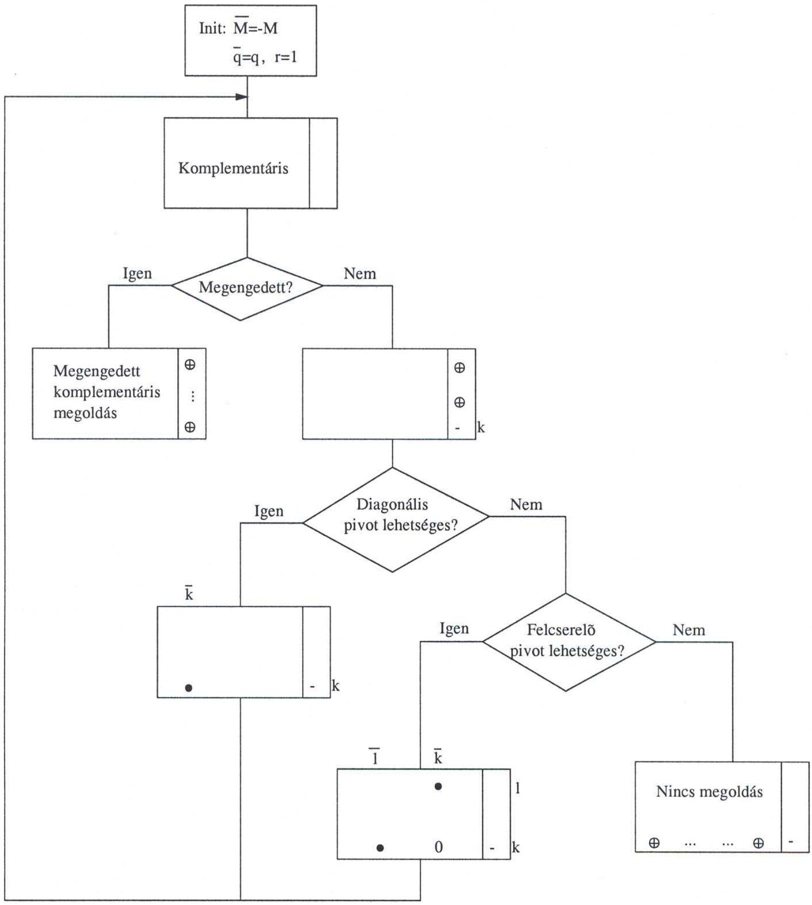
Legyen továbbá $\mathbf{s}_0(\alpha) = 0$ minden $\alpha \in I$ indexre. Könnyű belátni, hogy $\mathbf{s}_r \geq \mathbf{s}_{r-1}$ és $\mathbf{s}_r \neq \mathbf{s}_{r-1}$.

Az algoritmust a következőképpen fogalmazhatjuk meg.

Input:
  Adott az (1) feladat az M elégséges mátrixszal és legyen
  $\bar{M} := -M$, $\bar{q} := q$, $r := 1$.
Begin
  $J := \{\alpha \in I : \bar{q}_\alpha < 0\}$
  While $(J \neq \emptyset)$ **do**
    $J_{\max} := \{\beta \in J : \mathbf{s}_{r-1}(\beta) \geq \mathbf{s}_{r-1}(\alpha),\ \text{bármely } \alpha \in J\}$
    Legyen $k \in J_{\max}$ tetszőleges
    If $(\bar{m}_{kk} < 0)$ **then**
      diagonális pivot $\bar{m}_{kk}$ elemen
      $\mathbf{s}$ modosítása
      $r := r + 1$
    Else
      $K := \{\alpha \in I : \bar{m}_{k\alpha} < 0\}$
      If $(K = \emptyset)$ **then**
        Stop: Az LCP-nek nincs megengedett megoldása
      Else
        $K_{\max} = \{\beta \in K : \mathbf{s}_{r-1}(\beta) \geq \mathbf{s}_{r-1}(\alpha),\ \text{bármely } \alpha \in K\}$
        Legyen $l \in K_{\max}$ tetszőleges
        Felcserélős pivot az $\bar{m}_{kl}$ és $\bar{m}_{lk}$ elemeken
        $\mathbf{s}$ módosítása
        $r := r + 2$
      Endif
    Endif
  EndWhile
Stop: Megengedett és komplementáris megoldást állítottunk elő
End

Criss–cross típusú algoritmus

Az algoritmus a triviális komplementáris megoldásból indul, és mivel csak diagonális illetve felcserélő pivotokat csinál, ezért a komoplementaritást meg is őrzi. Lévén a mátrix elégséges, a 2.2 lemma biztosítja, hogy ha felcserélő pivotra kerül sor, akkor a mátrix kiválasztott elemeinek az előjele megfelelő lesz. Az algoritmus csak olyan esetben áll le, ha vagy nincs megoldás, vagy megtalálta azt. Elég tehát azt igazolni, hogy az algoritmus *véges*. Figyelembe véve azt, hogy a lehetséges bázisok száma véges, ezért azt kell megmutatnunk, hogy a criss-cross típusú algoritmus nem *ciklizál*.



1. ábra. A criss–cross típusú algoritmus folyamatábrája

3.2 Az ortogonalitási tulajdonság

Akkeleş, Balogh és Illés [1] bizonyítását általánosítjuk elégséges mátrixokra, miközben egyben le is egyszerűsítjük azt, lehetővé téve az algoritmus módosítását az EP–tételek szellemében. Bizonyításunk jelentős része a jól ismert ortogonalitási tételen alapszik.

Definiáljuk a $\mathbf{t}^{(i)}$, $i \in J_B$ illetve a $\mathbf{t}_j$, $j \in J_N \cup \{q\}$ vektorokat [11] a következőképpen:

$$\left(\mathbf{t}^{(i)}\right)_k = \begin{cases} \bar{m}_{ik}, & \text{ha } k \in J_N \cup \{q\} \\ 1, & \text{ha } k = i \\ 0, & \text{ha } k \in J_B \setminus \{i\} \end{cases}$$

illetve

$$(\mathbf{t}_j)_k = \begin{cases} \bar{m}_{kj}, & \text{ha } k \in J_B \\ -1, & \text{ha } k = j \\ 0, & \text{ha } k \in (J_N \cup \{q\}) \setminus \{j\} \end{cases}$$

Ezek után az ortogonalitási tételt [12,11] az alábbi formában mondhatjuk ki:

3.1 Tétel. *Bármely $M \in I\!R^{n\times n}$ mátrix és tetszőleges B' illetve B'' bázisok esetén az $M_{B'}$ mátrixhoz tartozó $\mathbf{t}'^{(i)}$ vektorok merőlegesek az $M_{B''}$ mátrixhoz tartozó $\mathbf{t}''_j$ vektorokra.*

3.3 Majdnem leállási táblák

Rátérhetünk az algoritmus végességének bizonyítására. Tegyük fel, hogy van olyan példa, amelyre az algoritmus nem véges. Mivel a lehetséges komplementáris bázisok száma véges, legfeljebb $\binom{2n}{n}$, így ez csak úgy lehetséges, ha ciklizálás lép fel. A ciklizálást mutató példák közül vegyünk egy minimális méretűt. Egy ilyen probléma esetén a minimalitás miatt egy ciklus során minden változó mozog.

Tekintsünk azt a pillanatot, amikor már minden változó legalább egyszer mozgott. Ekkor már $|J_{\max}| = |K_{\max}| = 1$ minden iterációban, mivel minden pivotálásnál a mozgó változókhoz olyan s számláló értéket rendelünk, mely még nem szerepelt, és azonos értékű változóknak mindig az egyike és csak az egyike van bázisban.

Tekintsük az r-edik iterációt. Az aktuális bázisbeli változók közül az $\mathbf{s}$ szerinti rendezés alapján legkisebb sorszámú v_p változó indexére igaz, hogy

$$p = \operatorname{argmin}\{i \in J_B \,:\, s_r(i) \le s_r(j),\ \forall\, j \in J_B\}\,.$$

Figyelembe véve, hogy a v_p változónak az $\mathbf{s}$ szerinti értéke mindaddig nem módosul, amíg a bázisban van, ezért a pivotálási szabály szerint, ha a sorában kezdeményezünk pivotálást valamely $r' > r$ iterációban, akkor (i) nem megengedettnek kell lennie, és (ii) a nem megengedett változók között az $\mathbf{s}$ szerinti rendezésben maximális kell, hogy legyen az $s_{r'}(p)$ értéke. Az (ii) feltétel kizárólag úgy teljesülhet, hogy a v_p az egyetlen nem megengedett változó az r' iterációban. Ehhez az állapothoz tartozó rövid pivot táblák a következők lehetnek:

1. Az algoritmus kiválasztotta a v_p változót a bázisból való távozásra.

Az $\bar{m}_{pp} < 0$, vagyis diagonális pivot [(a) tábla] lehetséges: u_p bekerül a bázisba, míg v_p elhagyja azt.

$$
(a)\qquad
\begin{array}{c|ccc|c|}
 & & & u_p & \\
\hline
 & & & & \oplus \\
 & & & & \oplus \\
 & & & & \vdots \\
 & & & & \vdots \\
 & & & & \oplus \\
 & & & & \oplus \\
v_p & & & - & - \\
\hline
\end{array}
\qquad\qquad
(b)\qquad
\begin{array}{c|ccc|c|}
 & u_j & & u_p & \\
\hline
 & & & & \oplus \\
v_j & & & + & \oplus \\
 & & & & \vdots \\
 & & & & \vdots \\
 & & & & \oplus \\
 & & & & \oplus \\
v_p & - & & 0 & - \\
\hline
\end{array}
$$

A diagonális pivot után az $\mathbf{s}$ értékei az alábbi szabály szerint módosulnak

$$
s_{r'}(\alpha) = \begin{cases} r', & \text{ha } \alpha \in \{p, \bar{p}\}\,; \\ s_{r'-1}(\alpha), & \text{egyébként}\,. \end{cases}
$$

2. Az algoritmus kiválasztja v_p változót a bázisból való távozásra, de $\bar{m}_{pp} = 0$, vagyis felcserélős pivot [(b) tábla] szükséges. Az u_p és u_j változók belépnek a bázisba, míg a v_p és v_j változók elhagyják azt.

A $\mathbf{q}$ oszlopa azonos az előző esetével. Ebben az esetben nem lényeges, hogy u_j vagy v_j változó van-e a bázisban. Mi azt az esetet vizsgáljuk, mikor v_j van a bázisban. A másik esetben csupán az $s_{r'+1}(\alpha)$ definíciójában kell a j és $\bar{j}$ szerepét felcserélni.

$$
\mathbf{s}_{r'+1}(\alpha) = \begin{cases} r', & \text{ha } \alpha \in \{\bar{p}, j\}\,; \\ r'+1, & \text{ha } \alpha \in \{p, \bar{j}\}\,; \\ s_{r'-1}(\alpha), & \text{egyébként}\,. \end{cases}
$$

3. Az algoritmus egy u_j változót választ a bázisba való belépésre, de $\bar{m}_{jj} = 0$, így felcserélős pivotra van szükség, és az algoritmus kiválasztja az u_p változót is.

$$
(c)\qquad
\begin{array}{c|cccccccc|c|}
 & & & & u_p & & & & u_j & \\
\hline
 & & & & & & & & \ominus & \\
 & & & & & & & & \vdots & \\
 & & & & & & & & \ominus & \\
v_p & & & & & & & & + & \\
 & & & & & & & & \ominus & \\
 & & & & & & & & \vdots & \\
 & & & & & & & & \ominus & \\
v_j & \oplus & \cdots & \oplus & - & \oplus & \cdots & \oplus & 0 & - \\
\hline
\end{array}
$$

A választási szabály értelmében v_j sorában csak az u_p és a $\mathbf{q}$ oszlopában lehet negatív elem, és lévén $\bar{m}_{jj} = 0$, így a 2.2 lemma alapján az u_j oszlopa a v_j sorának segítségével kitölthető. Ebben az esetben sem lényeges, hogy u_j vagy v_j van-e a bázisban. Mi azt az esetet vizsgáljuk, mikor v_j van a bázisban, a másik esetben csupán az $s_{r'+1}(\alpha)$ definíciójában kell a j és $\bar{j}$ szerepét felcserélni.

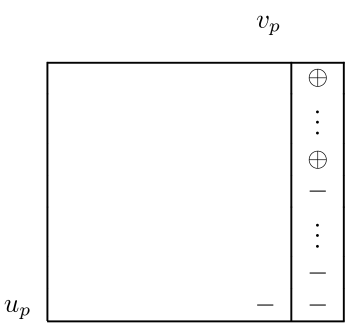
$$s_{r'+1}(\alpha) = \begin{cases} r', & \text{ha } \alpha \in \{p, \bar{j}\}\,; \\ r'+1, & \text{ha } \alpha \in \{\bar{p}, j\}\,; \\ s_{r'-1}(\alpha), & \text{egyébként}\,. \end{cases}$$

A későbbiek miatt érdemes megfigyelni, hogy az 1. és 2. esetekben csak az algoritmus választási szabálya befolyásolta a döntésünket, míg a 3. esetben kihasználtuk a mátrixunk elégségességét az u_j oszlopának kitöltésekor.

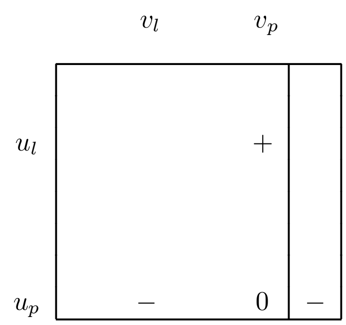
Most pedig tekintsük azt a pillanatot, mikor az u_p újra elhagyja a bázist. Ez legyen az $r'' > r'$ iterációban, és a hozzátartozó bázist jelölje B''. Az r'' iterációhoz tartozó rövid pivot tábla a következő háromféle formát öltheti fel:

A. A pivotálási szabály alapján az u_p változót választjuk ki a bázis elhagyására, $\bar{m}_{pp} < 0$, vagyis diagonális pivotálásra kerül sor.

(A)



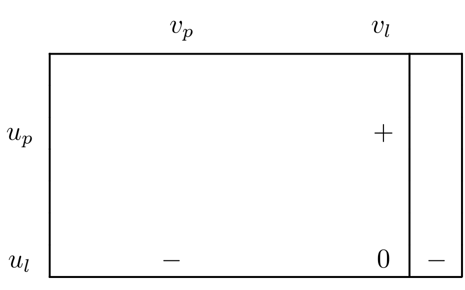
(B)



B. A pivotálási szabály alapján u_p változót választjuk ki a bázis elhagyására, de $\bar{m}_{pp} = 0$, vagyis felcserélős pivotra van szükség: v_l (vagy u_l) belép a bázisba, míg u_l (vagy v_l) elhagyja azt.

C. A algoritmus az u_l(vagy v_l) változót választja, de $\bar{m}_{ll} = 0$, így felcserélős pivotra van szükség, és v_p belép a bázisba, míg u_l elhagyja azt. A második pivotálásnál u_p távozik a bázisból és v_l belép.

(C)



A következőkben megmutatjuk, hogy ha az M elégséges mátrix, akkor az $(a) - (c)$ táblák egyikét sem követheti az $(A) - (C)$ táblák valamelyike.

A következő fejezetben definiált algoritmus elemzésének érdekében külön figyelmet fordítunk arra, hogy az esetek vizsgálatában milyen szerepet játszik a mátrix elégségessége.

3.4 Segédtételek

A következőkben megmutatjuk, hogy az $(a)-(c)$ táblák egyikét sem követheti az $(A)-(C)$ táblák egyike sem. Előbb azokat az eseteket vizsgáljuk meg, amelyek nem használják a lineáris komplementaritási feladat mátrixának elégségességét.

Először azt mutatjuk meg, hogy a (c) táblát nem követheti az (A) illetve (B) táblák egyike sem.

3.2 Lemma. *Jelölje $M_{B'}$ a (c) esethez, míg $M_{B''}$ az (A) (illetve a (B)) esethez tartozó bázis táblákat. Tekintsük a $\mathbf{t}'^{(\bar{j})}$ és $\mathbf{t}''_q$ vektorokat, amelyek rendre a v_j bázis változó $M_{B'}$ táblából kiolvasható sorához, illetve az $M_{B''}$ táblából kiolvasható $\mathbf{q}$ vektor oszlopához tartoznak. Ekkor*

$$(\mathbf{t}'^{(\bar{j})})^T\,\mathbf{t}''_q > 0.$$

Bizonyítás. Legyen $J'' := \{\alpha \in J_{B''} \;:\; \bar{q}''_i < 0\}$. Az (A) (illetve a (B)) tábla értelmezése miatt $p \in J''$, és mivel az u_p változót választottuk a bázisból való távozásra, ezért az indexválasztási szabály alapján a J'' elemei közül ez érkezett legkésőbb a bázisba, azaz egyik $\alpha \in J''\backslash\{p\}$ indexű változó sem mozgott az $M_{B'}$ bázis óta, és így $J''\backslash\{p\} \subset J_{B'}$ teljesül, és így $\mathbf{t}'_{\bar{j}i} = 0$ minden $i \in J''\backslash\,\{\bar{j}, p\}$ indexre, vagyis

$$\sum_{i \in J''\backslash\{\bar{j},p\}} t_{\bar{j}i} t''_{iq} = 0\;, \tag{3}$$

Figyelembe véve, hogy $s_{r'}(\bar{j}) = s_{r'}(p)$ és $s_{r''-1}(j) > s_{r''-1}(p)$ tudjuk, hogy t''_{jq} és $t''_{\bar{j}q}$ nemnegatívak. A (c) táblából kiolvashatjuk, hogy $t'_{\bar{j}j} = 0$, $t'_{\bar{j}\bar{j}} = 1$, $t'_{\bar{j}\bar{p}} = 0$, $t'_{\bar{j}p} < 0$ és $t'_{\bar{j}q} < 0$, így

$$t'_{\bar{j}\bar{j}}\,t''_{\bar{j}q} + t'_{\bar{j}j}t''_{jq} + t'_{\bar{j}\bar{p}}t''_{\bar{p}q} + t'_{\bar{j}p}\,t''_{pq} + t'_{\bar{j}q}\,t''_{qq} \geq t'_{\bar{j}p}t''_{pq} - t'_{\bar{j}q} > 0\;, \tag{4}$$

lévén $t''_{qq} = -1$ definíció szerint, és $t''_{pq} < 0$ az algoritmus választási szabályának értelmében ((A) és (B) táblák).

Ha továbbá $l \notin J'' \cup \{j, \bar{j}, p, \bar{p}, q\}$, akkor ismét a (c) ábrából kiolvasható, hogy $t'_{\bar{j}l} \geq 0$ és J'' definíciója szerint $t''_{lq} \geq 0$, vagyis

$$\sum_{l \notin J \cup \{j,\bar{j},p,\bar{p},q\}} t'_{\bar{j}l}\,t''_{lq} \geq 0\;. \tag{5}$$

Összeadva a (3)-(5) egyenlőtlenségeket éppen az állításunkat kapjuk. ∎

Figyeljük meg, hogy a (c) táblából a v_j változó sorát tekintettük, míg az (A) és (B) tábláknál a $\mathbf{q}$ oszlopát, tehát nem használtuk ki a mátrix elégségességét, és így a lemma csak a pivotálási szabálytól függ: a (c) és (A) (illetve (B)) esetek, az ortogonalitási tétel miatt, az előző lemmát figyelembe véve kizáróak, függetlenül a mátrix elégségességétől.

A következő lemmánk értelmében az (a) és a (b) táblákat nem követheti a (C) tábla.

3.3 Lemma. *Jelölje $M_{B'}$ az (a) (illetve (b)) esethez, míg $M_{B''}$ a (C) esethez tartozó bázis táblákat. Tekintsük a $\mathbf{t}'_q$ és $\mathbf{t}''^{(l)}$ vektorokat, melyek rendre az $M_{B'}$ tábla $\mathbf{q}$ oszlopához, illetve az $M_{B''}$ tábla u_l bázisváltozójának a sorához tartoznak. Ekkor*

$$(\mathbf{t}''^{(l)})^T\, \mathbf{t}'_q > 0\ .$$

Bizonyítás. Az előző lemmához hasonlóan $J''_l := \{i \in I_{N''} : t''_{li} < 0\} \subset I_{N'}$, így $t'_{iq} = 0$ teljesül bármely $i \in J''_l$ indexre, és emiatt

$$\sum_{i \in J''_l} t''_{li}\, t'_{iq} = 0\ . \tag{6}$$

Továbbá, minden $j \notin J_2 := J''_l \cup \{\bar{l}, l, \bar{p}, p, q\}$ indexre $t''_{lj} \geq 0$ és $t'_{jq} \geq 0$, így

$$\sum_{j \notin J_2} t''_{lj} t'_{jq} \geq 0\ . \tag{7}$$

Az $M_{B'}$ és $M_{B''}$ táblákból kiolvasható, hogy $t'_{qq} = -1$, $t''_{ll} = 1$, $t''_{l\bar{l}} = t''_{lp} = t'_{pq} = 0$, továbbá $t''_{l\bar{p}} < 0$, $t''_{lq} < 0$, $t'_{\bar{p}q} < 0$, $t'_{\bar{l}q} \geq 0$ *és* $t'_{lq} \geq 0$ így

$$t''_{l\bar{l}} t'_{\bar{l}q} + t''_{ll}\, t'_{lq} + t''_{l\bar{p}}\, t'_{\bar{p}q} + t''_{lp} t'_{pq} + t''_{lq}\, t'_{qq} = t'_{lq} + t''_{l\bar{p}}\, t'_{\bar{p}q} - t''_{lq} \geq t''_{l\bar{p}} t'_{\bar{p}q} - t''_{lq} > 0\ .$$

Összeadva a (6) – (7) összefüggéseket, a kívánt állítást kapjuk. ∎

Az előző lemmában sem használtuk ki az M mátrix elégségességét. Rátérhetünk azoknak az eseteknek a tárgyalására, amikor a feladat mátrixának az elégségessége lényeges lesz.

A következő lemmában igazoljuk, hogy az (a) (illetve (b)) táblákat nem követheti sem az (A), sem pedig a (B) tábla.

3.4 Lemma. *Legyen adott az $(\mathbf{u}', \mathbf{v}')$ és $(\mathbf{u}'', \mathbf{v}'')$, komplementáris megoldások, amelyek rendre az (a) és (b), illetve az (A) és (B) táblákhoz tartoznak. Ekkor*

$$(\mathbf{u}' - \mathbf{u}'')\, M\, (\mathbf{u}' - \mathbf{u}'') \leq \mathbf{0}\ .$$

Bizonyítás. Mind a négy esetet egyszerre bizonyítjuk.

$$\begin{aligned}
(\mathbf{u}' - \mathbf{u}'')\, M\, (\mathbf{u}' - \mathbf{u}'') &= (\mathbf{u}' - \mathbf{u}'')\, (\mathbf{q} + M\,\mathbf{u}' - \mathbf{q} - M\,\mathbf{u}'') \\
&= (\mathbf{u}' - \mathbf{u}'')\, (\mathbf{v}' - \mathbf{v}'') \\
&= \mathbf{u}'\,\mathbf{v}' - \mathbf{u}'\,\mathbf{v}'' - \mathbf{u}''\,\mathbf{v}' + \mathbf{u}''\,\mathbf{v}'' \\
&= -\mathbf{u}'\,\mathbf{v}'' - \mathbf{u}''\,\mathbf{v}',
\end{aligned}$$

ahol az utolsó egyenlőség a megoldások komplementaritásának a következménye. Jelölje $J'' := \{\, \alpha \in J_{B''} \,:\, \bar{q}''_i < 0 \,\}$. A pivotálási szabály miatt $s_{r''}(p) > s_{r''}(\alpha)$, bármely $\alpha \in J''\setminus\{p\}$ indexre, így ezen indexek nem mozogtak B' bázis óta, vagyis $\alpha \in J_{B'}$, és így bármely $i \in J''\setminus\{p\}$ indexre

$$u'_i\, v''_i + u''_i\, v'_i = 0 \tag{8}$$

teljesül, hiszen az u'_i vagy v''_i, illetve u''_i vagy v'_i értéke nulla. Az (a) és (b) illetve az (A) és (B) táblákból kiolvasható, hogy $u'_p = 0$, $v'_p < 0$ és $u''_p < 0$, $v''_p = 0$, vagyis

$$u'_p\, v''_p + u''_p\, v'_p > 0\,.$$

Továbbá bármely $j \notin J''$ indexre $u'_j, v'_j, u''_j, v''_j \geq 0$, és emiatt

$$u'_j\, v''_j + u''_j\, v'_j \geq 0\,. \tag{9}$$

Összefoglalva, az $(\mathbf{u}' - \mathbf{u}'')$ vektor olyan, hogy $(\mathbf{u}' - \mathbf{u}'')\, M\, (\mathbf{u}' - \mathbf{u}'') \leq \mathbf{0}$. ∎

Fontosnak tartjuk megjegyezni, hogy a bizonyításunk konstruktív, azaz a B' illetve a B'' bázisokból az M mátrix nem elégségességét bizonyító $\mathbf{u}' - \mathbf{u}''$ vektor könnyedén meghatározható.

A utolsó lemmánkban azt az esetet vizsgáljuk meg, amikor a (c) táblát a (C) tábla követné.

3.5 Lemma. *Jelölje $M_{B'}$ a (c), míg $M_{B''}$ a (C) esethez tartozó bázis táblákat. Tekintsük a $\mathbf{t}'_j$ és $\mathbf{t}''^{(l)}$ vektorokat, amelyek az $M_{B'}$ bázis tábla u_j nem bázis változójának az oszlopához, illetve az $M_{B''}$ bázis tábla u_l bázis változójának a sorához tartoznak. Ekkor*

$$(\mathbf{t}''^{(l)})^T\, \mathbf{t}'_j < 0\,.$$

Bizonyítás. Legyen $J''_l = \{i \in I_{N''} \,:\, t''_{li} < 0\}\setminus\{j\}$. Mivel tábláink komplementárisak, és a pivotálási szabály szerint mindig azt a változót választjuk a lehetségesek közül, amelyik a legutoljára mozgott, így a J''_l indexeihez tartozó változók az $M_{B'}$ óta nem mozogtak, ezért $\bar{J}''_l \subset I_{B'}$ és $J''_l \subset I_{N'}$, és így $t'_{ij} = 0$ teljesül, bármely $i \in J''_l$ esetén. Összefoglalva

$$\sum_{i \in J''_l \cup \{q\}} t''_{li}\, t'_{ij} = 0\,. \tag{11}$$

Másfelől, ha $i \notin J_1 := J''_l \cup \{q, p, \bar{p}, j, \bar{j}, l, \bar{l}\}$, akkor $t'_{ij} \leq 0$ a (c) ábra alapján. A J''_l definíciója miatt $t''_{li} \geq 0$, és így

$$\sum_{i \notin J_1} t''_{li}\, t'_{ij} \leq 0\,, \tag{12}$$

Az $M_{B'}$ és $M_{B''}$ táblákból, a $\mathbf{t}$ vektorok definícióját is figyelembe véve

$$t'_{pj} = t''_{l\bar{l}} = t'_{\bar{j}j} = t'_{qj} = t''_{lp} = 0,\ t''_{ll} = 1,\ t'_{jj} = -1 \ \text{ és } t'_{lj} \leq 0,\ t''_{l\bar{p}} < 0,\ t'_{\bar{p}j} > 0$$

így

$$t''_{lq}\,t'_{qj} + t''_{lp}\,t'_{pj} + t''_{l\bar{p}}\,t'_{\bar{p}j} + t''_{lj}\,t'_{jj} + t''_{l\bar{j}}\,t'_{\bar{j}j} + t''_{ll}t'_{lj} + t''_{l\bar{l}}\,t'_{\bar{l}j} < -t''_{lj}\,. \tag{13}$$

Hátra van még, hogy megmutassuk, $t''_{lj} \geq 0$. A B' bázis táblán a (c) esetén felcserélős pivotot hajtunk végre. Ekkor $s_{r'}(\bar{j}) = s_{r'}(p) = r'$ és $s_{r'+1}(j) = s_{r'+1}(\bar{p}) = r' + 1$. Az $M_{B''}$ tábla esetén $J''_{max} = \{\bar{p}\}$. Mivel a tábla komplementáris, így két eset lehetséges:

1. Ha $j \in I_{N''}$, akkor az u_j az $(r^{'} + 1)$ és a $r^{''}$ iterációk között mozog, így $s_{r''}(j) > s_{r''}(\bar{p})$, és ez a pivotálási szabály szerint csak úgy lehetséges, ha $t''_{lj} \geq 0$.
2. Ha $j \in I_{B''}$, akkor $t''_{lj} = 0$.

Összeadva a (11) – (13) egyenlőtlenségeket, a kívánt állítást kapjuk. ∎

Az előző lemma bizonyításakor a (c) tábla szerkezeténél kihasználtuk a tábla elégségességét.

3.5 A criss–cross típusú algoritmus végessége

Ebben a részben igazoljuk a criss–cross algoritmus végességét, majd ennek segítségével új, konstruktív bizonyítást adunk a lineáris komplementaritási feladatok dualitástételére is.

3.2 Tétel. *A criss–cross típusú algoritmus véges az elégséges mátrixszal adott lineáris komplementaritási feladatra.*

Bizonyítás. Bizonyításunk indirekt, azaz tegyük fel, hogy az algoritmus nem véges. Tekintettel arra, hogy a lineáris komplementaritási feladatnak véges sok különböző bázisa van, és az algoritmus egyértelműen meghatározza pivotáláskor a következő bázist, ezért abból, hogy az algoritmus nem véges, következik az algoritmus ciklizálása.

Tekintsük a fejezet elején említett minimális ellenpéldát. Ebben minden változó mozog egy ciklus során. A lemmák tanulsága szerint az utolsóként a bázisba belépő u_p változó már nem léphet ki a bázisból:

- Ha az (a) vagy a (b) esetben lép be, majd pedig az (A) vagy a (B) esetek valamelyikénél távozik a bázisból, akkor a 3.4 Lemma ellentmond az M mátrix elégségességének.
- Ha a (c) esetben lép be, majd az (A) vagy (B) esetek valamelyikében távozik a bázisból, akkor a 3.2 Lemma ellentmond az ortogonalitási tulajdonságnak.
- Ha a (c) esetben lép be, majd a (C) esetben távozik a bázisból, akkor a 3.5 Lemma ellentmond az ortogonalitási tulajdonságnak.
- Ha a (b) vagy a (c) esetben lép be, majd a (C) esetben távozik a bázisból, akkor a 3.2 Lemma ellentmond az ortogonalitási tulajdonságnak.

Minden lehetséges eset ellentmondásra vezet, így állításunkat beláttuk. ∎

A következő táblázat mutatja, hogy mely esetekben használtuk ki az M mátrix elégségességét:

	(a)	(b)	(c)
(A)	$*$	$*$	
(B)	$*$	$*$	
(C)			$*$

Az elégséges mátrixú LCP feladatok megoldhatóságának jellemzésére Fukuda és Terlaky [9] tételét az alábbi alternatíva tételként fogalmazhatjuk meg.[2]

3.3 Tétel. *Bármely $M \in I\!R^{n\times n}$ elégséges mátrix és $\mathbf{q} \in I\!R^n$ vektor esetén pontosan az egyik teljesül:*

(1) az LCP feladatnak van $(\mathbf{u}, \mathbf{v})$ megengedett komplementáris megoldása,

(2) a D-LCP feladatnak van $(\mathbf{x}, \mathbf{y})$ megengedett komplementáris megoldása.

Bizonyítás. Következik a 3.1 Lemmából és a 3.2 Tételből. ∎

Mivel a 3.1 Lemma csupán azt biztosítja, hogy a tétel két esete egyszerre nem állhat fenn, míg az egyik eset mindig fennállását a 3.2 Lemma algoritmikus bizonyításával igazoltuk, így a 3.3 Tétel konstruktív.

4 EP-tételek

Ebben a fejezetben szeretnénk a criss–cross típusú algoritmust alkalmassá tenni tetszőleges lineáris komplementaritási feladat megoldására. Természetesen nem várhatjuk el, hogy bármilyen típusú mátrixszal adott lineáris komplementaritási feladatot megoldhassunk a criss–cross típusú algoritmussal. Azt azonban fontosnak tartjuk, hogy ha egy feladatot nem tudunk megoldani ezzel az algoritmussal, akkor annak az okát meg tudjuk mutatni, és a bizonyítékunkat arra, hogy a mátrixunk nem elégséges, polinom időben leellenőrizhessük.

Gondolatmenetünknek megfelelő elmélet alapjait Cameron és Edmonds alapozta meg [2] dolgozatukban. Ők bevezették az ún. *EP–tétel*eket.

Egy EP (Existentially Polynomial time) tétel formája a következő:

$$[\forall \mathbf{x} : F_1(\mathbf{x}) \text{ vagy } F_2(\mathbf{x}) \text{ vagy } \dots \text{ vagy } F_k(\mathbf{x})] \ ,$$

ahol $F_i(\mathbf{x})$ olyan állítás, melynek formája

$$F_i(\mathbf{x}) = [\exists \mathbf{y}_i \text{ amelyre } \|\mathbf{y}_i\| \le \|\mathbf{x}\|^{n_i} \text{ és } f_i(\mathbf{x}, \mathbf{y}_i)] \, .$$

[2] Fukuda és Terlaky [9] cikkükben eredményüket dualitás tételnek hívják, habár szerintünk az alternatíva tétel pontosabban tükrözi a tétel mondanivalóját.

Itt $n_i \in \mathbb{N}$, $\|\mathbf{z}\|$ jelöli a $\mathbf{z}$ kódolási hosszát, $f(\mathbf{x}, \mathbf{y})$ pedig olyan állítás, melynek teljesülésére van polinomiális bizonyíték.

Mielőtt rátérnénk a lineáris komplementaritási feladat dualitás tételének az EP-tétel formájában történő megfogalmazására, szükségünk lesz néhány fogalom és állítás kimondására.

Egy $\mathbf{x}$ vektor *tartó*jának az $\{\, i \mid \mathbf{x}_i \neq 0 \,\}$ halmazt nevezzük. Egy adott vektorhalmazból egy minimális tartójú vektort *elemi vektor*nak nevezünk. Szükségünk lesz a vektorok *konform* [8] előállítására.

4.1 Definíció *Legyen $V \subseteq \mathbb{R}^n$ tetszőleges lineáris altér, $\mathbf{x}, \mathbf{x}^1, \ldots, \mathbf{x}^k$ vektorok a V altérből. Azt mondjuk, hogy az $\mathbf{x}$ vektor* konform *módon felbomlik $\mathbf{x}^1, \ldots, \mathbf{x}^k$ vektorokra, ha $\mathbf{x} = \mathbf{x}^1 + \ldots + \mathbf{x}^k$, és*

$$\begin{array}{lcl} x_i = 0 & \Longrightarrow & x_i^1 = \ldots = x_i^k = 0, \\ x_i > 0 & \Longrightarrow & x_i^1, \ldots, x_i^k \geq 0, \\ x_i < 0 & \Longrightarrow & x_i^1, \ldots, x_i^k \leq 0 \end{array}$$

bármely $i = 1, \ldots, n$ indexre.

Tetszőleges lineáris altér esetén teljesül a következő:

4.1 Lemma. [16] *Legyen V lineáris altér az $\mathbb{R}^n$ térben. Ekkor bármely $\mathbf{x} \in V$ felbontható konform módon a V lineáris altér $\mathbf{c}^1, \ldots, \mathbf{c}^k$ elemi vektoraira.*

A fenti lemma segítségével megmutatható, hogy ha M nem elégséges, akkor ennek bizonyítéka megadható egy elemi, vagy két elemi vektor összegeként [8].

4.2 Lemma. *Ha M nem oszlop elégséges (sor elégséges), akkor létezik egy szigorúan előjelváltó (szigorúan előjeltartó) elemi vektor a V altérben ($V^\perp$ altérben), vagy létezik egy szigorúan előjelváltó (szigorúan előjeltartó) $\mathbf{x}$ vektor a V altérben, ($\mathbf{y}$ a $V^\perp$ altérben) mely felírható két komplementáris, elemi vektor összegeként.*

Az eddigi eredményekből kiindulva igazolható a következő tétel.

4.1 Tétel. [8] *Legyen $M \in \mathbb{R}^{n \times n}$ nem elégséges mátrix. Ekkor létezik M nem elégséges voltára olyan bizonyíték, amelynek a mérete polinomiálisan korlátozott az M input méretének a függvényében.*

A lineáris komplementaritási feladat dualitástételét EP–formában szeretnénk felírni. Ehhez előbb olyan alakra kell hoznunk a tételt, amelyben a mátrix elégségessége már nem feltétele az állításnak.

4.2 Tétel. *Bármely $M \in \mathbb{Q}^{n \times n}$ racionális mátrix és $\mathbf{q} \in \mathbb{Q}^n$ esetén legalább az egyik teljesül:*

(1) a (P-LCP) feladatnak van $(\mathbf{u}, \mathbf{v})$ megengedett komplementáris megoldása,

(2) a (D-LCP) feladatnak van $(\mathbf{x}, \mathbf{y})$ megengedett komplementáris megoldása,

(3) az M mátrix nem elégséges.

A tétel még nincs EP–formában. Az (1) és (2) rész teljesülése esetén maga a megoldás mérete polinomiális méretű. A (3) részhez azonban meg kell még mutatni, hogy ha az M mátrix nem elégséges, akkor ennek van egy polinomiális méretű bizonyítéka.

Most már megfogalmazhatjuk az LCP dualitástételt EP-formában [8], felhasználva a 4.1 Tételt.

4.3 Tétel. *Bármely $M \in \mathbb{Q}^{n\times n}$ mátrix, és $\mathbf{q} \in \mathbb{Q}^n$ esetén legalább az egyik teljesül:*

(1) a (P-LCP) feladatnak van $(\mathbf{u}, \mathbf{v})$ megengedett komplementáris megoldása, amelynek a kódolási mérete az M és $\mathbf{q}$ kódolási méretével polinomiálisan korlátozott;

(2) a (D-LCP) feladatnak van $(\mathbf{x}, \mathbf{y})$ megengedett komplementáris megoldása, amelynek a kódolási mérete az M és $\mathbf{q}$ kódolási méretével polinomiálisan korlátozott;

(3) az M mátrix nem elégséges, és ennek van olyan bizonyítéka, amelynek a kódolási mérete az M kódolási méretével polinomiálisan korlátozott.

Fontos kiemelni, hogy az (1) és (2) esetek kizáróak, míg a (3) egyszerre teljesülhet akár az (1), akár a (2) esettel együtt is. Másfelől, természetes feltételként jelentkezik az, hogy a mátrix elemei racionális számok legyenek.

A tétel bizonyításához módosítani kell az algoritmusunkat és igazolni a módosított algoritmus végességét.

Az algoritmust úgy módosítjuk, hogy vagy megoldja a $(P-LCP)$ feladatot vagy a duálisát, vagy kimutatja, hogy a bemeneti mátrix nem elégséges[3], és ennek polinomiális méretű bizonyítékát szolgáltatja.

A 2.2 Lemma biztosítja, hogy ha a mátrix elégséges, akkor a pivotálási műveletek mindig elvégezhetők, ha pedig nem, akkor megadja a kívánt bizonyítékot arra, hogy az M mátrix nem elégséges.

Hátra van annak vizsgálata, hogyan követjük nyomon a ciklizálás elkerülését. A *ciklizálás elkerülésének* a vizsgálatakor vegyük észre, hogy az eredeti algoritmus végességének bizonyításában valójában nem igazán lényeges, hogy a ciklizáló példa minimális. Tekintsünk ugyanis egy tetszőleges ciklizáló példát. Legyen a ciklusban részt vevő változók indexeinek halmaza R. Figyeljünk meg egy olyan pillanatot, mikor már elkezdődött a ciklizálás, minden a ciklizálásban résztvevő változó már mozgott, és a ciklus során az algoritmus olyan változót választ a bázisba való belépésre, melynek az R bázison kívüli változói közül a legkisebb az $\mathbf{s}$ értéke. Ehhez a pillanathoz tartozó bázist jelölje B', és legyen a legkisebb $\mathbf{s}$ értékű, az R halmazbeli belépő változó az u_p. Jelölje B'' azt a bázist, mikor az u_p legközelebb mozog.

Ekkor az $M_{B'}$, illetve az $M_{B''}$ pivot táblák szerkezete, az R indexű változókra illetve a $\mathbf{q}$ vektorra megszorítva, pontosan az $(a)-(c)$ illetve $(A)-(B)$ tábláké lehet. A két állapot között olyan változó, amelynek az indexe nem az

[3]Egy mátrix elégségességének az eldöntésére jelenleg nem ismert hatékony, polinomiális algoritmus.

R halmazba tartozik, nem mozgott. Így a 3.2, 3.3, 3.5 lemmák esetében a fundamentális elemi vektorok szorzatában az R illetve q indexein kívül pontosan az egyik tag van a bázisban, tehát ezen indexekhez a szorzatban mindig nulla tartozik. Ugyanezen okból a 3.4 lemma $-\mathbf{u}^{'}\cdot\mathbf{v}^{''}-\mathbf{u}^{''}\cdot\mathbf{v}^{'}$ szorzatában az R halmazon kívüli indexeken nullák lesznek. Vagyis a bizonyítások átmenthetők általános ciklizáló példára is.

Input
  Adott az (1) feladat. $\bar{M} = -M$, $\bar{\mathbf{q}} = \mathbf{q}$, $r = 1$, Q inicializálása
Begin
  While $((J := \{\alpha \in I : \bar{\mathbf{q}}_\alpha < 0\}) \neq \emptyset)$ **do**
    $J_{\max} := \{\beta \in J : \mathbf{s}_{r-1}(\beta) \geq \mathbf{s}_{r-1}(\alpha)$, bármely $\alpha \in J$- re$\}$
    Legyen $k \in J_{\max}$ tetszőleges.
    Ellenőrizendő a $-\mathbf{u}^{'}\cdot\mathbf{v}^{''}-\mathbf{u}^{''}\cdot\mathbf{v}^{'}$ az elmentett $Q(k)$ segítségével:
    If $(-\mathbf{u}^{'}\cdot\mathbf{v}^{''}-\mathbf{u}^{''}\cdot\mathbf{v}^{'} \leq\neq \mathbf{0})$ **then**
      Stop: M nem elégséges, bizonyíték az $\mathbf{u}^{'} - \mathbf{u}^{''}$.
    Endif
    If $(\bar{m}_{kk} < 0)$ **then**
      diagonális pivot az $\bar{m}_{kk}$ értéken
      $Q(k) = [J_B, \bar{\mathbf{m}}_q]$, $r := r+1$
    ElseIf $(\bar{m}_{kk} > 0)$
      Stop: M nem elégséges, bizonyíték mint a 2.2 lemmában.
    Else /* $\bar{m}_{kk} = 0$ */
      $K := \{\alpha \in I : \bar{m}_{k\alpha} < 0\}$
      If $(K = \emptyset)$ **then**
        Stop: D-LCP megoldás
      Else
        $K_{\max} = \{\beta \in K : \mathbf{s}_{r-1}(\beta) \geq \mathbf{s}_{r-1}(\alpha)$, bármely $\alpha \in K$- ra$\}$
        Legyen $l \in K_{\max}$ tetszőleges.
        If ($\mathbf{m}_k$ és $\mathbf{m}^k$ vagy $\mathbf{m}_l$ és $\mathbf{m}^l$ előjelszerkezete sérűl) **then**
          Stop: M nem elégséges, bizonyíték mint a 2.2 lemmában.
        Endif
        Felcserélős pivot az $\bar{m}_{kl}$ és az $\bar{m}_{lk}$ számokon, $\mathbf{s}$ módosítása
        $Q(k) = [J_B, \bar{\mathbf{m}}_q]$, $Q(l) = [\emptyset, \mathbf{0}]$, $r := r+2$
      Endif
    Endif
  EndWhile
Stop: Megengedett és komplementáris megoldást állítottunk elő.
End

Módosított criss–cross típusú algoritmus általános (LCP) feladatra

*Az elégségesség hiányának a kezelés*ekor idézzük fel, hogy az elégségességet a 3.4 lemma, illetve a 3.5 lemma esetében használtuk ki. Ez utóbbi az elégségességgel járó, a 2.2 lemmára alapuló előjelszerkezetet használta. Tehát ha az algoritmus minden felcserélős pivot esetén ((c) és (C) ilyen esetekhez tartoznak) ellenőrzi az előjelszerkezet meglétét, akkor az ortogonalitási tétel miatt a (c) táblát nem követheti a (C) tábla. Ha bárhol sérül az előjelszerkezet, az M nem elégséges voltára a kívánt bizonyítékot ugyanez a lemma biztosítja.

Marad az (a) és a (b), illetve az (A) és a (B) táblák esete. A 3.4 lemma a

$$-\mathbf{u}^{'} \cdot \mathbf{v}^{''} - \mathbf{u}^{''} \cdot \mathbf{v}^{'} \tag{14}$$

alakú vektorok nem szigorúan előjelfordító voltán alapul, olyan $M_{B'}$ és $M_{B''}$ táblák esetén, ahol ugyanazon változó mozog mindkét pivot során, és mindkét esetben a vizsgált változó az aktívan választott (vagyis nem a felcserélős pivot második változója). Érdemes megfigyelni, hogy nincs szükség a teljes táblára, elegendő annak a $\mathbf{q}$ vektorhoz tartozó oszlopa (vagyis az aktuális megoldás), illetve a bázisban levő indexek halmaza. Amennyiben a (14) vektor szigorúan előjelfordító, akkor a 3.4 lemma utáni megjegyzés értelmében a lineáris komplementaritási feladat mátrixának nem elégséges voltára a bizonyíték az $\mathbf{u}^{'} - \mathbf{u}^{''}$ vektor.

Vezessünk be egy $Q(p)$ $(p = 1, \ldots, n)$ listát. A lista minden indexéhez két n dimenziós vektor tartozik. Egy $Q(j) = [\{I\}, \{\mathbf{h}\}]$ alakú értékadás alatt azt értjük, hogy a j indexhez tartozó Q értékben a bázisváltozók indexeinek helyére az I indexeket, a $\mathbf{q}$ vektorhoz tartozó értékek helyére pedig a $\mathbf{h}$ vektort írjuk. Induláskor legyen minden p indexre:

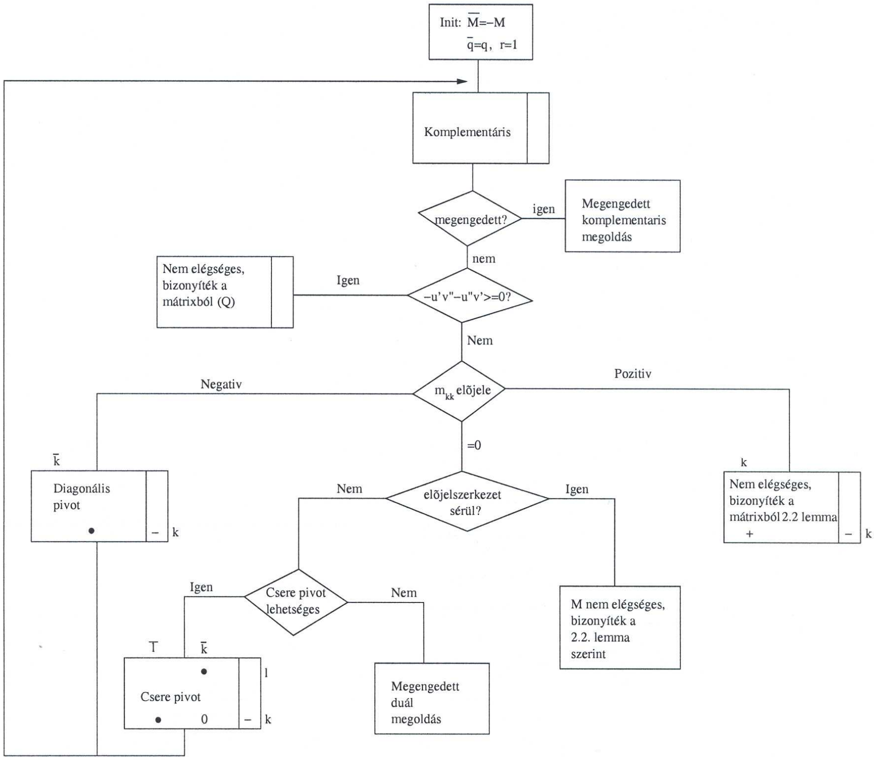
$$Q(p) := \begin{bmatrix} [1, \ldots, n] \\ [0, \ldots, 0] \end{bmatrix} \qquad p = 1, \ldots, n.$$

Mikor egy u_l vagy v_l változó egy olyan pivot során távozik a bázisból, mely során vagy diagonális pivothoz tartozik, vagy olyan felcserélőshöz, melyben ő aktív (az ő $\mathbf{s}$ értékét vizsgálta az algoritmus), a $Q(l)$ értékét úgy módosítjuk, hogy az első vektorba az u_l vagy v_l változó bázisból való kilépése előtti bázis változók *index*eit, míg a második vektorba az u_l vagy v_l változó bázisból való kilépése előtti bázis változók *érték*ét írjuk, vagyis

$$Q(l) := \begin{bmatrix} [\text{bázisváltozók indexei}] \\ [\text{bázisváltozók aktuális értéke}] \end{bmatrix}.$$

Ha az u_l vagy v_l változó passzív módon (felcserélős pivot esetén, a diagonális érték nulla volta miatt) lép be a bázisba, akkor a $Q(l)$ értéket módosítsuk a következőképpen:

$$Q(l) := \begin{bmatrix} [1, \ldots, n] \\ [0, \ldots, 0] \end{bmatrix}.$$



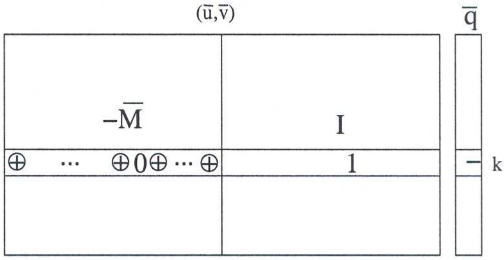
2. ábra. A módosított criss–cross típusú algoritmus folyamatábrája

Az algoritmus, mikor a báziscseréhez ér, megnézi, hogy az aktívan belépő változót előző kilépésekor is aktívan választotta, vagy sem. Ha igen, a Q lista segítségével ellenőrzi a (14) vektort, majd csak ezután módosítja a Q listát. Mivel a komplementáris változók pivot műveletek során egyszerre mozognak, így nem szükséges külön helyet fenntartani számukra a Q listában.

Érdemes megfigyelni, hogy a Q kezdeti, illetve a felcserélős pivot esetén a passzív változókra beállított értéke miatt elegendő bármely pivot esetén ellenőrizni a (14) szorzatot. Ha nem az (a) (vagy a (b)) táblát követi az (A) (vagy a (B)) tábla, akkor a szorzat mindig nulla lesz.

A Q listát nem lenne szükséges minden esetben kitölteni. Így az algoritmus minimális módosításával tárhelyet lehet megtakarítani. Megfigyelhető továbbá, hogy ekkor a legrosszabb esetben a Q lista tárigénye n^2 egész és n^2 lebegőpontos szám tárigényével azonos.

Meg kell még vizsgálnunk azt az esetet, mikor a $(P - LCP)$ feladatnak nincs megoldása. Ez akkor fordulhat elő, ha az algoritmus azt találja, hogy $K = \emptyset$. Ehhez az esethez tartozó pivot tábla a következő:



3. ábra.

Definiáljuk a következő vektort:

$$(\mathbf{x}', \mathbf{y}') = \mathbf{t}^{(k)} \mid_{J_N \cup J_B} .$$

Az ortogonalitási tételből adódik, hogy a fenti vektor az $[-M^T \mid -I]$ mátrix összes sorára merőleges, vagyis $M^T\mathbf{x}' + \mathbf{y}' = \mathbf{0}$. Ugyancsak az ortogonalitást használva, de a jobb oldali $\mathbf{q}$ vektor oszlopára (a kiindulási bázisban) adódik, hogy

$$(\mathbf{x}', \mathbf{y}')^T (\mathbf{t}_q \mid_{J_N \cup J_B}) = (\mathbf{x}', \mathbf{y}')^T (\mathbf{q}, \mathbf{0}) = \mathbf{x}'^T \mathbf{q} = q_k .$$

Vagyis a $(\mathbf{x}, \mathbf{y}) = (\mathbf{x}', \mathbf{y}')/(-q_k)$ vektor a $(D - LCP)$ feladat egy megoldása, mivel a nemnegativitás és a komplementaritás a tábla szerkezetéből adódóan teljesül.

5 Összefoglalás

Akkeleş, Balogh és Illés [1] új típusú, lineáris feltételes, konvex kvadratikus célfüggvényes feladatra megfogalmazott, criss–cross algoritmusait általánosítottuk elégséges mátrixok esetére. A végesség bizonyítását, az általánosítás mellett, leegyszerűsítettük.

Az elégséges mátrixokra általánosított algoritmust a jobb gyakorlati alkalmazhatóság érdekében kiegészítettük úgy, hogy ne kelljen előre feltételezni a bemeneti mátrix elégségességét. Ha az elégségesség hiánya miatt az algoritmus nem tudja biztosítani a végességet, akkor leáll és polinomiálisan korlátozott méretű bizonyítékot ad az adott mátrix nem elégséges voltára.

Célunkat a lineáris komplementaritási feladatok dualitás tételének [9], az EP-tétel [8] formájának a felhasználásával értük el. Az algoritmus Akkeleşék új típusú pivotálási szabályainak köszönhetően, főleg az első báziscserék esetén jelentős választási szabadságot biztosít, lehetővé téve esetlegesen felmerülő numerikusan instabil báziscserék elkerülését. Ez nyilvánvalóan javítaná a criss–cross típusú algoritmusok gyakorlati alkalmazhatóságát.

Hasonlóan a leírt LIFO (last in – first out: utoljára bekerülő – először kikerülő) pivotálási szabályhoz, ugyanígy bizonyítható, illetve általánosítható a *leggyakrabban választott változó* (most–often–selected–variable) pivotálási szabály is. Ehhez mindössze az $\mathbf{s}$ vektort kell másként definiálni, például

$$\mathbf{s}_r(\alpha) = \begin{cases} \mathbf{s}_{r-1}(\alpha) + 1, & \text{ha } \alpha \in I \text{ mozog az } r\text{-edik iterációban;} \\ \mathbf{s}_{r-1}(\alpha), & \text{egyébként.} \end{cases}$$

Az algoritmus elején jelentős szabadságunk van a nem megengedett változók kiválasztásakor. Ez az algoritmus numerikus viselkedése szempontjából mindenféleképpen előnyösnek tekinthető, azonban magában rejti a ciklizálás lehetőségét is. Később azonban rögzül a változók egymáshoz viszonyított sorrendje. Ezt a sorrendet az adatokon túl az is befolyásolja, hogy előzőleg milyen pivot pozíciókat választottunk.

A változók sorrendjének egyértelművé válása biztosítja az algoritmus végességét, a pivot pozíció kiválasztási szabadságának a rovására.

Érdekességképpen megemlítjük, hogy az $\mathbf{s}$ vektor segítségével a minimálindexes választási szabály, mint speciális eset leírható, és a bizonyítások változatlanul működnek. Rögzítsük ehhez az $\mathbf{s}$ vektort a következő módon: $s_r(i) = i$ minden iterációban, bármely $i \in I$ indexre. Ekkor a bizonyítások jelentősen leegyszerűsödnek.

Köszönetnyilvánítás

Illés Tibor köszönetet mond a Magyar Tudományos Akadémiának a *Bolyai János Kutatási Ösztöndíj*ért (BO/00334/00), amellyel a kutatásait a cikk előkészítő szakaszában támogatták. A szerzők folytonos optimalizálási kutatásait az OTKA T049789 pályázata támogatja. Csizmadia Zsolt és Illés Tibor köszönetet mond a MOL Rt.-nek a számukra jelenleg is folyósított operációkutatási ösztöndíjért.

Irodalom

1. A. A. Akkeleş, Balogh L. és Illés T., A véges criss-cross módszer új variánsai biszimmetrikus lineáris komplementaritási feladatra, *Alkalmazott Matematikai Lapok* 21 pp. 1–25 (2003)
2. K. Cameron, J. Edmonds, Existentially polytime theorems, in: W. Cook, P. D. Seymour (Editors), *Polyhedral Combinatorics*, DIMACS Series in Discrete Mathematics and Theoretical Computer Science AMS pp. 83–100 (1990).
3. Y. Y. Chang, Least index resolution of degeneracy in linear complementarity problems, *Technical Report* 79-14, Department of Operations Research, Stanford University, Stanford, CA, (1979).
4. S. J. Chung, NP-completeness of the linear complementarity problem, *Journal of Optimization Theory and Applications*, Vol. 60, No. 3, pp. 393–399 (1989).
5. R. W. Cottle, The principal pivoting method revisited, *Mathematical Programming*, Vol. 48, No. 3, pp. 369–386 (1990).
6. R. W. Cottle, J.-S. Pang, R. E. Stone, The linear complementarity problem, *Computer Science and Scientific Computing* (1992).

7. R. W. Cottle, J.-S. Pang, V. Venkateswaran, Sufficient matrices and the linear complementarity problem, *Linear Algebra and Its Applications*, Vol. 114/115, pp. 230–249 (1989).
8. K. Fukuda, M. Namiki, A. Tamura, EP theorems and linear complementarity problems, *Discrete Applied Mathematics*, Vol. 84, pp. 107–119 (1998).
9. K. Fukuda, T. Terlaky, Linear complementarity and oriented matroids, *Journal of the Operations Research Society of Japan*, Vol. 35, pp. 45–61 (1992).
10. D. den Hertog, C. Roos, T. Terlaky, The linear complementarity problem, sufficient matrices, and the criss-cross method, *Linear Algebra and Its Applications*, Vol. 187, pp. 1–14 (1993).
11. Klaszky E. és Terlaky T., A pivot technika szerepe a lineáris algebra néhány alapvető tételének a bizonyításában, *Alkalmazott Matematikai Lapok* 14, 425–448 (1989).
12. E. Klaszky, T. Terlaky, Some generalizations of the criss-cross method for quadratic programming. *Optimization*, Vol. 24, pp. 127-139 (1991).
13. E. Klaszky, T. Terlaky, The role of pivoting in proving some fundamental theorems of linear algebra, *Linear Algebra and Its Applications*, Vol. 151, pp. 97–118 (1991).
14. M. Kojima, N. Megiddo, T. Noma, A. Yoshise, *A unified approach to interior point algorithms for linear complementarity problems*, Lecture Notes in Computer Science, 538 (1991).
15. K. G. Murty, *Linear complementarity, Linear and Nonlinear Programming*, Sigma Series in Applied Mathematics, Vol. 3, (1988).
16. R.T. Rockafellar, The elementary vectors of a subspace of $I\!R^n$, in: R. C. Bose, T. A. Dowling (Eds.), *Combinatorial Mathematics and Its Applications*, Proceedings Chapel Hill Conference, pp. 104–127 (1969).
17. T. Terlaky, A convergent criss-cross method, *Math. Oper. und Stat. ser. Optimization*, Vol. 16, No. 5, pp. 683–690 (1985).
18. H. Väliaho , A new proof for the criss-cross method for quadratic programming, *Optimization*, Vol. 25, pp. 391–400 (1992).
19. H. Väliaho, Criteria for Sufficient Matrices, *Linear Algebra and its Applications*, Vol. 233, pp. 109–129 (1996).
20. H. Väliaho, P_* matrices are just sufficient, *Linear Algebra and Its Applications* Vol. 239, pp. 103–108 (1996).
21. Zh. Wang, A conformal elimination free algorithm for oriented matroid programming, *Chinese Annals of Mathematics*, 8, B1, (1987).

NEW VARIANTS OF THE CRISS-CROSS METHOD FOR LINEAR COMPLEMENTARITY PROBLEMS

The sufficient matrix class is the widest class, for which the usual criss-Cross algorithm is proved to be finite (Hertog, Roos and Terlaky, 1993). There is no polynomial algorithm known to check wether a matrix is sufficient or not. Following the results of Akkeleş, Balogh and Illés (2003), the criss-cross algorithm with LIFO and most-often-selected pivot rules are extended for general linear complementarity problems (LCP). While most algorithms require a priori information on some properties of the input matrix, our new, criss-cross type algorithms work for

every matrix, and either solves the problem, or provides a polynomial sized certificate that the input matrix does not belong to the class of sufficient matrices. The finiteness of our algorithm provides a new, constructive proof for the duality theorem of LCP, developed by Fukuda and Terlaky (1992) for oriented matroids.

CONTENTS

TARTALOM

SZIGMA
Matematikai-közgazdasági folyóirat

A Gazdaságmodellezési Társaság lapja

Terjeszti a Gazdaságmodellezési Társaság

ISSN 0039-8128

`www.szigma.ktk.pte.hu`