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 $i \in \{1, \dots, n\}$ és $= i - n$ ha $i \in \{n+1, \dots, 2n\}$
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
$< \dots >$	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 $\alpha = \{\alpha_1, \dots, \alpha_k\} \subseteq \{1, \dots, n\}$ 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*

$$\begin{cases} x_i\,(M\mathbf{x})_i \le 0 & \textit{minden } i \in \{1, \dots, n\} \textit{ indexre} \\ x_j\,(M\mathbf{x})_j < 0 & \textit{valamely } j \in \{1, \dots, n\} \textit{ indexre} \end{cases} \tag{2}$$

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

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

illetve a

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

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 $J_N = J \backslash J_B$ 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 $M \in \mathbb{R}^{n \times n}$ mátrix *átrendezésén* a $P^T M P$ mátrixot értjük, ahol P egy permutációmátrix. A következő eredmények bizonyítása megtalálható Cottle [5] dolgozatában.

2.3 Lemma. *Legyen $M \in \mathbb{R}^{n \times n}$ 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 $D \in \mathbb{R}^n$ 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 $M \in \mathbb{R}^{n \times n}$ 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.

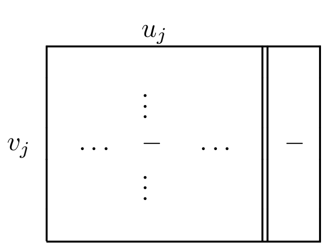
A lineáris komplementaritási feladat, (1), kiinduló, komplementáris megoldása az $\mathbf{u} = \mathbf{0}$, $\mathbf{v} = \mathbf{q}$. A pivot tábla mátrixát $\bar{M}$ jelöli.

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.

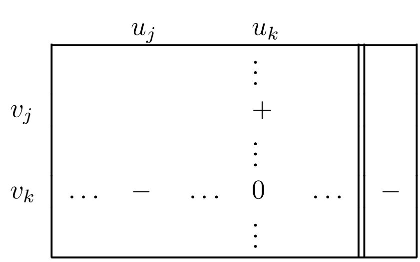
Kétfajta, diagonális és felcserélős pivot műveletet fogunk végezni, melyek mindegyike megőrzi az aktuális megoldás komplementáris voltát.

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 $\mathbf{s}_r : I \rightarrow \mathbb{N}_0^{2N}$

számláló vektor segítségével kezelik:

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

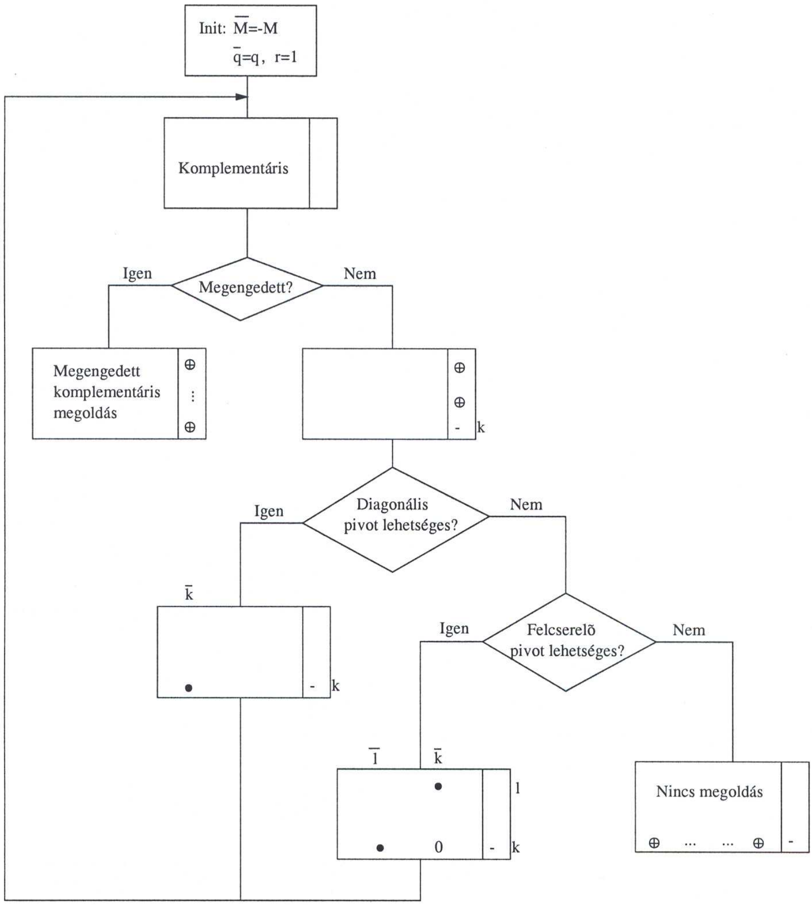
Legyen továbbá $\mathbf{s}_0(\alpha) = 0$ minden $\alpha \in I$ indexre. Könnyű belátni, hogy $\mathbf{s}_r \geq \mathbf{s}_{r-1}$ és $\mathbf{s}_r \neq \mathbf{s}_{r-1}$.

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

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

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.

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

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.

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

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)
$$\begin{array}{c|cc|c} & & v_p & \\ \hline & & & \oplus \\ & & & \vdots \\ & & & \oplus \\ & & & - \\ & & & \vdots \\ & & & - \\ u_p & & - & - \\ \hline \end{array}$$

(B)
$$\begin{array}{c|ccc|c} & & v_l & v_p & \\ \hline & & & & \\ u_l & & & + & \\ & & & & \\ u_p & & - & 0 & - \\ \hline \end{array}$$

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)
$$\begin{array}{c|ccc|c} & & v_p & v_l & \\ \hline & & & & \\ u_p & & & + & \\ & & & & \\ u_l & & - & 0 & - \\ \hline \end{array}$$

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 $i \notin J_1 := J''_l \cup \big\{q, p, \bar{p}, j, \bar{j}, l, \bar{l}\big\}$, 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 $M \in \mathbb{R}^{n\times n}$ elégséges mátrix és $\mathbf{q} \in \mathbb{R}^n$ 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:

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

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 $n_i \in I\!N$, $\|\mathbf{z}\|$ 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 $V \subseteq I\!R^n$ 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*

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

bármely $i = 1, \ldots, n$ indexre.

Tetszőleges lineáris altér esetén teljesül a következő:

4.1 Lemma. [16] *Legyen V lineáris altér az $I\!R^n$ 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 $M \in I\!R^{n\times n}$ 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:

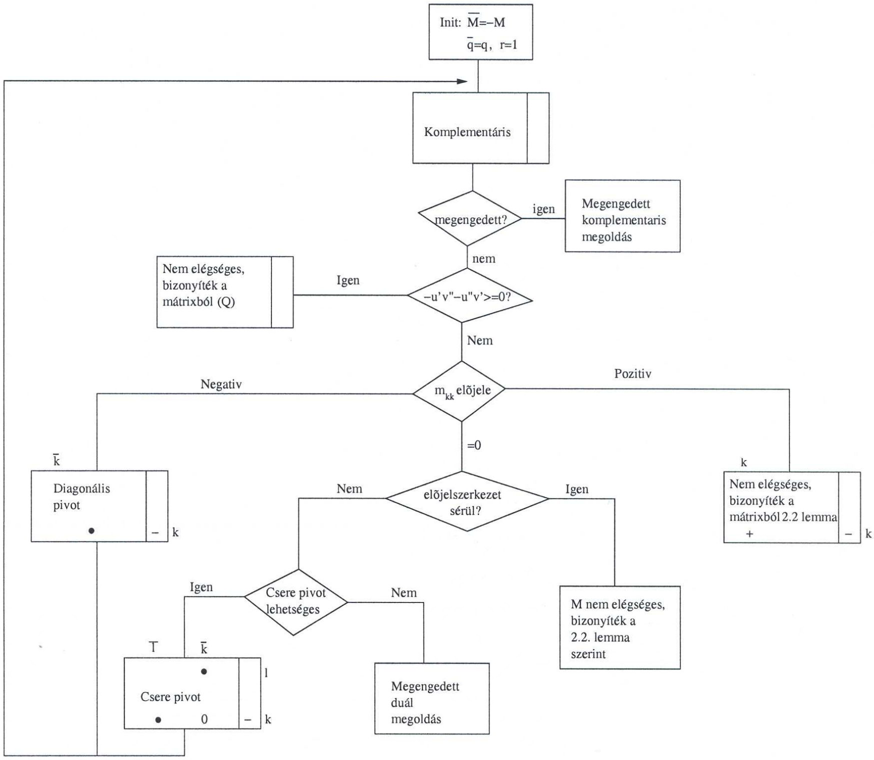
$$Q(p) := \begin{bmatrix} [1, \ldots, n] \\ [0, \ldots, 0] \end{bmatrix} \qquad p = 1, \ldots, n.$$

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:

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



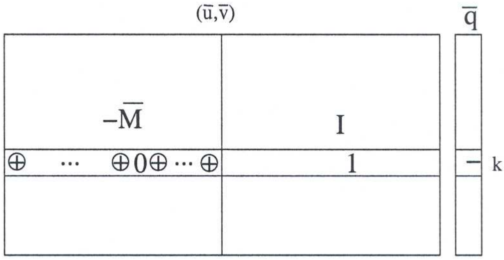
2. ábra. A módosított criss–cross típusú algoritmus folyamatábrája

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

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

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 $\mathbb{R}^n$, 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.