Alkalmazott Matematikai Lapok **37** (2020), 317–338.
DOI: 10.37070/AML.2020.37.2.14

LINEÁRIS PROGRAMOK A MAXIMÁLIS ÉLSÚLYÚ KLIKK PROBLÉMÁRA

SZABÓ SÁNDOR, SZTOJKOVICS DÓRA

Egy adott gráfban a maximális élsúlyú klikk megtalálása egy ismert és fontos probléma, sok alkalmazással. A probléma megoldására léteznek a lineáris programozás eszközeit, valamint lineáris programozással nem kapcsolatos kombinatorikus alapú, keresési fát használó algoritmusok is. Egy olyan tanulmányhoz [6] fűzünk megjegyzéseket, amelyben a szerzők egy kombinatorikus alapú algoritmust hasonlítottak össze kettő lineáris programozás, és egy kvadratikus programozás alapú algoritmussal. Az [1] és [6] cikkben szereplő egyik programot módosítjuk. A módosításokhoz a gráf csúcsainak és éleinek színezésével jutunk. Az új programokat numerikus kísérletekben teszteltük.

1. Bevezetés

Legyen $G = (V, E)$, $V = \{v_1, v_2, \ldots, v_n\}$, egy véges, egyszerű, irányítatlan gráf. Tekintsük V-nek egy $C \subseteq V$ részhalmazát. C klikk G-ben, ha bármely két csúcsa között fut él, azaz minden $v_i, v_j \in C$ $(1 \leq i, j \leq n, i \neq j)$ esetén $\{v_i, v_j\} \in E$. Ha $|C| = k$, akkor azt mondjuk, hogy C egy k-klikk. C maximum klikk G-ben, ha minden $K \subseteq V$ klikk esetén $|K| \leq |C|$. Ekkor a klikk méretét (csúcsainak számát) az $\omega(G)$ szimbólummal jelöljük és a G klikkszámának nevezzük. Ha $\omega(G) = k$, akkor a klikk éleinek száma $\binom{k}{2} = \frac{k(k-1)}{2}$. C maximális klikk G-ben, ha tovább nem bővíthető, azaz bármely $v_i \in V, v_i \notin C$ csúcsot hozzávéve C-hez a kapott $C \cup \{v_i\}$ halmaz már nem alkot klikket. A definícióból következik, hogy minden maximum klikk maximális, de nem minden maximális klikk maximum klikk.

G gráf minden $\{u, v\} \in E$ éléhez rendeljünk hozzá egy $w_e(u, v)$ nemnegatív értéket, így egy élsúlyozott gráfot kapunk. Keressünk G-ben egy olyan klikket, amelyben az élek súlyának összege maximális. Ez egy sokat vizsgált NP-nehéz probléma fontos alkalmazásokkal ([4], [5], [7], [8]), amit maximális élsúlyú klikk problémának nevezünk. A [6] cikkben a szerzők mutatnak egy olyan kombinatorikus alapú algoritmust, amely gyorsabban számolt az általuk tesztelt gráfokon, mint más, lineáris programozás alapú algoritmusok. Célunk a cikkben mutatott vegyes-egészértékű program módosítása volt, amihez lokális csúcsszínezést és élszínezést, valamint globális élszínezést használtunk.

1.1. Csúcsszínezés

Színezzük ki G gráf csúcsait úgy, hogy a következő feltételek teljesüljenek:

1. Minden csúcsnak pontosan egy színe van.
2. Éllel összekötött csúcsok különböző színűek.

Ha a fenti feltételek teljesülnek, akkor a csúcsok jó színezéséről beszélünk. Tegyük fel, hogy G csúcsai k színnel jól színezhetők, de $k-1$ színnel már nem. Ekkor a k számot G kromatikus számának nevezzük, és $\chi(G)$ szimbólummal jelöljük. $\chi(G)$ meghatározása egy NP-nehéz optimalizálási probléma.

1.2. Élszínezés

Színezzük ki G gráf éleit úgy, hogy teljesüljenek az alábbi feltételek:

1. Minden élnek pontosan egy színe van.
2. G-ben bármely háromszög (3-klikk) mindhárom éle különböző színű.
3. G-ben bármely teljes négyszög (4-klikk) mind a hat éle különböző színű.

Ha ezek a kritériumok teljesülnek, akkor az élek élesített[1] jó színezéséről beszélünk. Az élek jó színezése egy ismert fogalom, amit akkor használunk, ha az egy közös csúcsra illeszkedő élek különböző színűek. A mi esetünkben egy csúcsra illeszkedhetnek azonos színű élek is.

A [6] cikk szerzői olyan vegyes-egészértékű programot használtak az összehasonlításban, melyben a változók egy része bináris volt. Mi olyan szempontból vizsgáltuk a programot, hogy a lineáris relaxációja, amelyben a bináris változóknak megengedjük, hogy tetszőleges 0 és 1 közötti valós értéket vegyenek fel, mennyire közelíti jól az optimumot. Az így kapott lineáris programot a vegyes-egészértékű program lineáris relaxációjának nevezzük. Egy egészértékű vagy vegyes-egészértékű program összes megengedett megoldása a lineáris relaxációjának is megengedett megoldása lesz. Egy maximum probléma esetén az egészértékű vagy vegyes-egészértékű program lineáris relaxációjának optimum értéke mindig felső becslést ad az eredeti program optimum értékére.

Az [1] és [6] cikkekben használt egészértékű és vegyes-egészértékű programok lineáris relaxációinál szeretnénk erősebb lineáris relaxációkat kapni. A lineáris relaxációk összehasonlításának két módszerét használjuk, egy elméleti jellegűt és egy numerikus számolásokon alapuló módszert:

[1] Az élesített itt nem azt jelenti, hogy javított, vagy erős, hanem azt, hogy az élszínek különbözőségét a 3- és 4-klikkek élei alapján követeljük meg.

1. Ellenőrizzük (ahol ez lehetséges), hogy az egyik lineáris program megengedett megoldásainak halmaza tartalmazza-e a másik lineáris program megengedett megoldásainak halmazát.
2. Gondosan választott tesztfeladatok esetén azt tapasztaljuk, hogy az új megfogalmazás tipikusan jobb felső becsléseket ad, mint a régi.

2. Egészértékű programok a maximum klikk problémára a szakirodalomból

A maximális élsúlyú klikk probléma megoldása előtt tekintsünk olyan egészértékű programokat, amelyek megoldják a maximum klikk problémát.

Legyen $G = (V, E)$, $V = \{v_1, v_2, \ldots, v_n\}$ továbbra is egy véges, egyszerű, irányítatlan gráf. Mutatunk három lineáris programozás alapú algoritmust, amelyek megadják G-ben a maximum klikket. Ezekben az egészértékű programokban nem vesszük figyelembe az élsúlyokat.

2.1. Él átfogalmazás

Legyen C egy klikk G-ben, és vezessük be az $x_1, x_2, \ldots, x_n$ döntési változókat. A változók száma megegyezik G csúcsainak számával. Az x_i változó értéke 1, ha v_i szerepel a klikkben ($1 \le i \le n$), 0 különben:

$$x_i = \begin{cases} 1, & \text{ha } v_i \in C, \\ 0, & \text{ha } v_i \notin C. \end{cases}$$

Ekkor a célfüggvény a következő alakban írható:

$$\sum_{i=1}^{n} x_i \to \max.$$

Ez maximalizálni fogja a csúcsok számát. Két változó összege akkor és csak akkor lehet 2-vel egyenlő, ha a megfelelő csúcsok elemei C-nek, ekkor össze vannak kötve éllel. Így, ha két csúcs között nem fut él, akkor mindkettő nem szerepelhet egy klikkben, és a megfelelő változók összege legfeljebb 1 lehet. A feltételek a következők:

$$x_i + x_j \le 1, \quad \text{ha } \{v_i, v_j\} \notin E.$$

Az él átfogalmazással kapott egészértékű program:

$$\left.\begin{array}{llr} \text{maximum:} & \displaystyle\sum_{i=1}^{n} x_i & (1.1) \\ \text{feltételek:} & x_i + x_j \le 1, \quad (v_i, v_j) \notin E & (1.2) \\ & x_i \in \{0, 1\}, \quad v_i \in V. & (1.3) \end{array}\right\} (1)$$

A következő két formálisabb állítás bizonyítása kiolvasható a fenti megfontolásokból.

Legyen $U \subseteq V$. Definiáljuk az $\underline{\alpha}^T = (\alpha_1, \alpha_2, \ldots, \alpha_n)$ vektort úgy, hogy

$$\alpha_i = \begin{cases} 1, & \text{ha } v_i \in U, \\ 0, & \text{ha } v_i \notin U. \end{cases}$$

Az $\underline{\alpha}^T$ vektort az U karakterisztikus vektorának nevezzük.

2.1. Állítás. *Ha $C \subseteq V$ klikk G-ben, akkor C karakterisztikus vektora, $\underline{\alpha}^T$, megengedett megoldása* (1)*-nek.*

2.2. Állítás. *Ha $\underline{\alpha}^T$ megengedett megoldása* (1)*-nek, és $\underline{\alpha}^T$ a $C \subseteq V$ karakterisztikus vektora, akkor C klikk G-ben.*

2.2. Független halmaz átfogalmazás

A független halmaz átfogalmazás célfüggvénye és a változói ugyanazok, mint az él átfogalmazás esetén. Legyen $I \subseteq V$ olyan halmaz, amelyben semelyik két csúcs között nem fut él, azaz minden $v_i, v_j \in I$ $(1 \leq i, j \leq n)$ esetén $(v_i, v_j) \notin E$. Ekkor azt mondjuk, hogy I független halmaz. Ha minden $J \subseteq V$ független halmazra $|J| \leq |I|$, akkor I maximum független halmaz. I maximális független halmaz, ha tovább nem bővíthető, azaz, ha bármely $v_i \in V, v_i \notin I$ csúcsot hozzávéve I-hez a kapott $I \cup \{v_i\}$ halmaz már nem független. Minden maximum független halmaz maximális, de egy maximális független halmaz nem feltétlenül maximum független halmaz.

Egy független halmazban semelyik két csúcs között sem fut él, ezért a halmaz elemei közül legfeljebb egy csúcs szerepelhet egy klikkben:

$$\sum_{v_i \in I} x_i \leq 1, \quad \text{ahol } I \text{ maximális független halmaz.}$$

A független halmaz átfogalmazással kapott egészértékű program:

$$\left.\begin{array}{llr} \text{maximum:} & \displaystyle\sum_{i=1}^{n} x_i & (2.1) \\ \text{feltételek:} & \displaystyle\sum_{v_i \in I} x_i \leq 1, \quad I \text{ maximális független halmaz} & (2.2) \\ & x_i \in \{0,1\}, \quad v_i \in V. & (2.3) \end{array}\right\} (2)$$

A következő két állítás indoklása hasonlít a 2.1. és 2.2. Állítások igazolásához.

2.3. Állítás. *Ha $C \subseteq V$ klikk G-ben, akkor C karakterisztikus vektora, $\underline{\alpha}^T$, megengedett megoldása* (2)*-nek.*

2.4. Állítás. *Ha* $\underline{\alpha}^T$ *megengedett megoldása* (2)*-nek, és* $\underline{\alpha}^T$ *a* $C \subseteq V$ *karakterisztikus vektora, akkor* C *klikk* G*-ben.*

A maximális független halmazok előállításához egy NP-nehéz feladatot kell megoldani, és így a független halmaz átfogalmazás nem tűnik gyakorlatilag kivitelezhetőnek. A helyzet ennél árnyaltabb. Amikor a G gráfnak sok éle van, akkor a független halmazok felsorolása megoldható a Bronn-Kerbosch algoritmussal. Amikor G-nek kevés éle van, akkor a maximum klikk probléma oldható meg elfogadható ráfordítással.

2.3. A Croce-Tadei átfogalmazás

Az átfogalmazás megtalálható az [1] cikkben. A célfüggvényt és változókat az előző esetekhez hasonlóan kapjuk.

Legyen $\overline{N}(v_i)$ azon csúcsok halmaza, amelyek nem szomszédai v_i-nek G-ben, azaz $\overline{N}(v_i) = \{u : \{v_i, u\} \notin E, u \in V\}$. Jelölje $\overline{N}_i$ a v_i csúcs nemszomszédainak számát, $\overline{N}_i = |\overline{N}(v_i)|$. Ha v_i eleme a C klikknek, akkor $\overline{N}(v_i)$ halmaz semelyik eleme nem szerepelhet a klikkben, amit a következő feltételekkel írhatunk le:

$$\overline{N}_i x_i + \sum_{v_j \in \overline{N}(v_i)} x_j \le \overline{N}_i.$$

Ez az egyenlőtlenség a 2.1. alfejezetbeli (1) egészértékű program egyenlőtlenségei közül a v_i-re illeszkedő nem-élekhez tartozóknak az összege. A Croce-Tadei átfogalmazással kapott egészértékű program:

$$\left.\begin{array}{rlr}
\text{maximum :} & \displaystyle\sum_{i=1}^{n} x_i & (3.1)\\
\text{feltételek:} & \displaystyle\overline{N}_i x_i + \sum_{v_j \in \overline{N}(v_i)} x_j \le \overline{N}_i, \quad v_i \in V & (3.2)\\
& x_i \in \{0,1\}, \quad v_i \in V. & (3.3)
\end{array}\right\} (3)$$

2.5. Állítás. *Ha* $C \subseteq V$ *klikk* G*-ben, akkor* C *karakterisztikus vektora,* $\underline{\alpha}^T$*, megengedett megoldása* (3)*-nak.*

Bizonyítás. Legyen C klikk G-ben, és tegyük fel indirekt módon, hogy $\underline{\alpha}^T$ nem megengedett megoldás, azaz létezik olyan $1 \le i \le n$, amire

$$\overline{N}_i \alpha_i + \sum_{v_j \in \overline{N}(v_i)} \alpha_j \ge \overline{N}_i + 1.$$

Két esetet különböztetünk meg aszerint, hogy $\alpha_i = 0$ vagy $\alpha_i = 1$.

1. eset: Ha $\alpha_i = 0$, akkor azt az egyenlőtlenséget kapjuk, hogy

$$\sum_{v_j \in \overline{N}(v_i)} \alpha_j \geq \overline{N}_i + 1.$$

α_j értéke legfeljebb 1 lehet, és $|\overline{N}(v_i)| = \overline{N}_i$, innen következik, hogy

$$\sum_{v_j \in \overline{N}(v_i)} \alpha_j \leq \overline{N}_i.$$

Ellentmondásra jutottunk.

2. eset: Ha $\alpha_i = 1$, akkor azt kapjuk, hogy

$$\overline{N}_i + \sum_{v_j \in \overline{N}(v_i)} \alpha_j \geq \overline{N}_i + 1.$$

Mindkét oldalból kivonva $\overline{N}_i$-t a

$$\sum_{v_j \in \overline{N}(v_i)} \alpha_j \geq 1$$

egyenlőtlenség adódik. Ekkor létezik $1 \leq k \leq n$, amelyre $v_k \in \overline{N}(v_i)$, azaz $\{v_i, v_k\} \notin E$ és $\alpha_k = 1$. Mivel $\alpha_i = 1$ és $\alpha_k = 1$, ezért $v_k \in C$ és $v_i \in C$. C klikk, ezért $\{v_i, v_k\} \in E$. Ellentmondásra jutottunk.

Mindkét esetben ellentmondásra jutottunk, ezzel az állítást beláttuk. □

2.6. Állítás. *Ha $\underline{\alpha}^T$ megengedett megoldása* (3)*-nak, és $\underline{\alpha}^T$ a $C \subseteq V$ karakterisztikus vektora, akkor C klikk G-ben.*

Bizonyítás. Legyen $\underline{\alpha}^T$ megengedett megoldása (3)-nak, és tegyük fel indirekt módon, hogy C nem klikk G-ben. Ekkor létezik $1 \leq i, k \leq n$, $i \neq k$, amire $v_i, v_k \in C$, de $\{v_i, v_k\} \notin E$. Következik, hogy $\alpha_i = 1$, $\alpha_k = 1$ és $v_k \in \overline{N}(v_i)$. $\underline{\alpha}^T$ megengedett megoldás, ezért az

$$\overline{N}_i \alpha_i + \sum_{v_j \in \overline{N}(v_i)} \alpha_j \leq \overline{N}_i$$

feltétel teljesül. Behelyettesítve α_i és α_k értékeit azt kapjuk, hogy

$$\overline{N}_i + 1 + \sum_{v_j \in \overline{N}(v_i), j \neq k} \alpha_j \leq \overline{N}_i.$$

Átrendezve az egyenlőtlenséget

$$\sum_{v_j \in \overline{N}(v_i), j \neq k} \alpha_j \leq -1$$

adódik. Ellentmondásra jutottunk, mivel az összeg minden tagja nemnegatív. Ezzel az állítást beláttuk. □

A három egészértékű program ekvivalens egymással abban az értelemben, hogy ugyanazok a megengedett megoldásaik és az optimum értékeik.

Az él átfogalmazásban $\mathcal{O}\left(\binom{n}{2} - |E|\right) = \mathcal{O}(n^2)$ feltétel jelenik meg. Vannak olyan n csúcsú gráfok, amelyek $\mathcal{O}(2^n)$ maximális klikket tartalmaznak, és olyanok, melyekben $\mathcal{O}(2^n)$ független halmaz található [3]. Ezeknél a gráfoknál, ha a független halmaz átfogalmazással szeretnénk megtalálni a maximum klikket, akkor nagyon sok feltételt kapunk az egészértékű programban. Ezekkel szemben a Croce-Tadei átfogalmazás csak n feltételt tartalmaz, melyeket könnyen meghatározhatunk a szomszédsági mátrixból.

A maximális élsúlyú klikk probléma megoldására egy lehetőség, hogy a fenti programokat további feltételekkel egészítjük ki. A [6] cikkben a szerzők az él átfogalmazást bővítették újabb egyenlőtlenségekkel. A Croce-Tadei átfogalmazásban általában kevesebb feltétel szerepel, ezért ezt fogjuk használni az él átfogalmazás helyett.

3. A maximális élsúlyú klikk probléma

A maximális élsúlyú klikk probléma megoldására a [6] cikkben kettő lineáris programozás alapú, egy kvadratikus programozás alapú és egy kombinatorikus alapú algoritmust mutatnak a szerzők. A tesztelt gráfok esetén a leggyorsabb algoritmus a kombinatorikus alapú volt, amit a vegyes-egészértékű program követett. Ennek a módosításával foglalkoztunk.

Legyen $G = (V, E)$, $V = \{v_1, v_2, \ldots, v_n\}$, egy véges, egyszerű, irányítatlan gráf, ahogy eddig is. Jelölje $N(v_i)$ azon csúcsok halmazát, amelyek szomszédai $v_i \in V$-nek. Legyen $N^+(v_i) = N(v_i) \cap \{v_j : j > i\}$, és legyen $U_i = \sum_{v_j \in N^+(v_i)} w_e(v_i, v_j)$.

3.1. Vegyes-egészértékű program [6]

A következő vegyes-egészértékű program optimális megoldása megadja a maximális élsúlyú klikket G-ben:

$$\left.\begin{array}{rll} \text{maximum:} & \displaystyle\sum_{i=1}^{n-1} z_i & (4.1) \\ \text{feltételek:} & x_i + x_j \leq 1, \quad \{v_i, v_j\} \notin E & (4.2) \\ & z_i \leq U_i x_i, \quad v_i \in V \setminus \{v_n\} & (4.3) \\ & z_i \leq \displaystyle\sum_{v_j \in N^+(v_i)} w_e(v_i, v_j) x_j, \quad v_i \in V \setminus \{v_n\} & (4.4) \\ & x_i \in \{0, 1\}, \quad v_i \in V. & (4.5) \end{array}\right\} (4)$$

A (4) vegyes-egészértékű program az él átfogalmazás kiegészítése a (4.2), (4.3) feltételekkel, új célfüggvénnyel és a $z_1, z_2, \ldots, z_{n-1}$ változókkal. A célfüggvény az $n-1$ darab új változó összegének a maximuma:

$$\sum_{i=1}^{n-1} z_i \to \max.$$

Legyen $C \subseteq V$ klikk G-ben és $\underline{\alpha}^T$ a C karakterisztikus vektora. Legyen $\underline{\beta}^T = (\beta_1, \beta_2, \ldots, \beta_{n-1})$, ahol β_i $(1 \leq i \leq n-1)$ értéke nem nagyobb, mint a v_i csúcsból az i-nél nagyobb indexű C-beli csúcsokba futó élek súlyának összege, ha $v_i \in C$, ellenkező esetben pedig legfeljebb 0:

$$\beta_i \leq \begin{cases} \displaystyle\sum_{v_j \in N^+(v_i) \cap C} w_e(v_i, v_j), & \text{ha } v_i \in C, \\ 0, & \text{ha } v_i \notin C. \end{cases}$$

Ekkor a $\displaystyle\sum_{i=1}^{n-1} \beta_i$ összeg maximuma a C klikk élsúlyainak összegével egyezik meg.

3.1. Állítás. *Az* $(\underline{\alpha}^T, \underline{\beta}^T)^T$ *vektor megengedett megoldása* (4)*-nek.*

Bizonyítás. A 2.1. Állítás kimondja, hogy ha C klikk G-ben akkor, a (4.2) feltételek teljesülnek.

Tekintsük a (4.3) feltételeket. Ha $v_i \in C$, akkor

$$\beta_i \leq \sum_{v_j \in N^+(v_i) \cap C} w_e(v_i, v_j) \quad \text{és} \quad U_i \alpha_i = \sum_{v_j \in N^+(v_i)} w_e(v_i, v_j).$$

$(N^+(v_i) \cap C) \subseteq N^+(v_i)$ és a súlyok nemnegatívak, ezért $\beta_i \leq U_i \alpha_i$. Ha $v_i \notin C$, akkor $\beta_i \leq 0$ és $U_i \alpha_i = 0$, tehát az egyenlőtlenség ebben az esetben is teljesül.

Tekintsük a (4.4) feltételeket. Ha $v_i \in C$, akkor

$$\beta_i \leq \sum_{v_j \in N^+(v_i) \cap C} w_e(v_i, v_j) = \sum_{v_j \in N^+(v_i)} w_e(v_i, v_j) \alpha_j.$$

Ha $v_i \notin C$, akkor $\beta_i \leq 0$, a jobb oldalon pedig egy nemnegatív érték szerepel, tehát az egyenlőtlenség mindkét esetben teljesül. □

3.2. ÁLLÍTÁS. *Ha* $(\underline{\alpha}^T, \underline{\beta}^T)^T$ *megengedett megoldása* (4)*-nek, akkor az* $\underline{\alpha}^T$ *karakterisztikus vektorhoz tartozó* $C \subseteq V$ *halmaz klikk* G*-ben, valamint ha* $v_i \in C$*, akkor*

$$\beta_i \leq \sum_{v_j \in N^+(v_i) \cap C} w_e(v_i, v_j),$$

és $\beta_i \leq 0$*, ha* $v_i \notin C$*.*

Bizonyítás. A 2.2. Tételből következik, hogy a (4.2) feltételek miatt C klikk G-ben.

Ha $v_i \notin C$, akkor a (4.3) feltételekből adódik, hogy $\beta_i \leq 0$, a (4.4) egyenlőtlenségekből pedig következik, hogy

$$\beta_i \leq \sum_{v_j \in N^+(v_i)} w_e(v_i, v_j)\alpha_j.$$

A két korlát körül az első erősebb, ezért $\beta_i \leq 0$ teljesül.

Ha $v_i \in C$, akkor a (4.3) egyenlőtlenségekből azt kapjuk, hogy $\beta_i \leq U_i$, és a (4.4) feltételekből következik, hogy

$$\beta_i \leq \sum_{v_j \in N^+(v_i)} w_e(v_i, v_j)\alpha_j = \sum_{v_j \in N^+(v_i) \cap C} w_e(v_i, v_j).$$

Mivel

$$\sum_{v_j \in N^+(v_i) \cap C} w_e(v_i, v_j) \leq \sum_{v_j \in N^+(v_i)} w_e(v_i, v_j) = U_i,$$

ezért

$$\beta_i \leq \sum_{v_j \in N^+(v_i) \cap C} w_e(v_i, v_j).$$

□

A tételekből következik, hogy az optimális megoldás megadja a G-beli maximális élsúlyú klikket.

4. Új matematikai programok a maximális élsúlyú klikk problémára

Ebben a fejezetben a (4) vegyes-egészértékű program (4.2) és (4.3) feltételein módosítunk. A Croce-Tadei egyenlőtlenségek esetén, ha $x_i = 0$, akkor az i-edik feltétel megengedi, hogy v_i bármelyik nemszomszédját bevegyük a klikkbe. Ennek az egyenlőtlenségnek a módosítására több lehetőségünk is van.

A [6]-ban szereplő (4) program helyett az alábbi programot fogjuk használni.

$$\left.\begin{array}{lll}
\text{maximum:} & \displaystyle\sum_{i=1}^{n-1} z_i & (5.1)\\
\text{feltételek:} & \displaystyle C_i x_i + \sum_{v_j \in \overline{N}(v_i)} x_j \le C_i, \quad v_i \in V & (5.2)\\
 & z_i \le M_i x_i, \quad v_i \in V \setminus \{v_n\} & (5.3)\\
 & \displaystyle z_i \le \sum_{v_j \in N^+(v_i)} w_e(v_i, v_j) x_j, \quad v_i \in V \setminus \{v_n\} & (5.4)\\
 & \displaystyle\sum_{i=1}^{n} x_i \le S & (5.5)\\
 & x_i \in \{0,1\}, \quad v_i \in V. & (5.6)
\end{array}\right\} (5)$$

C_i megválasztását a 4.1. és a 4.2. alfejezetekben, M_i választását a 4.3. alfejezetben, S választását pedig a 4.4. alfejezetben fogjuk részletezni.

4.1. Az (5.2) korlátban a C_i megválasztása lokális csúcsszínezéssel

Tekintsük a Croce-Tadei egyenlőtlenségek i-edik csúcsra vonatkozó (3.2) feltételét, és színezzük ki v_i nemszomszédait jól. Alkalmazzuk a mohó színezőt: az első csúcs színe legyen 1. A második csúcs színe is legyen 1, ha nem szomszédja az elsőnek és 2, ha szomszédja. Tegyük fel, hogy kiszíneztünk k csúcsot s színnel jól. Tekintsük azt a legkisebb színszámú színosztályt, amelyben egyik csúcs sem szomszédja v_{k+1}-nek. Ha van ilyen, akkor v_{k+1} is bekerül ebbe a színosztályba, ha nincs, akkor v_{k+1} színe legyen $s+1$. Mivel egy klikkben bármely két csúcs között fut él, ezért minden csúcs különböző színű lesz. Így a színosztályok száma felső becslést ad a maximum klikk méretére.

A következő állításokban azt fogjuk megmutatni, hogy az eredeti (4) vegyes-egészértékű program

– (4.2) feltételét lecserélhetjük az (5.2) feltételre. Az így kapott programot jelöljük (6)-tal.

– (4.3) feltételét lecserélhetjük az (5.3) feltételre. Az így kapott programot jelöljük (7)-tel.

Legyen $G = (V, E)$, $V = \{v_1, v_2, \ldots, v_n\}$, egy véges, egyszerű, irányítatlan gráf. Színezzük ki minden csúcs esetén a csúcs nemszomszédaiból álló részgráfot a fenti színezéssel. Jelölje C_i a v_i nemszomszédai által kifeszített részgráf esetén kapott felső becslést a részgráfban található maximum klikk méretére. A Croce-Tadei egyenlőtlenségek i-edik feltételét úgy élesíthetjük, ha $\overline{N}_i$ helyére C_i-t írunk.

Jelölje $\overline{\omega}_i$ az i-edik csúcs nemszomszédai által kifeszített G-beli részgráfban található maximum klikk méretét. C_i felső becslés a klikkméretre, ezért $\overline{\omega}_i \leq C_i$.

4.1. Állítás. *Legyen $C \subseteq V$ karakterisztikus vektora $\underline{\alpha}^T$, és legyen $\underline{\beta}^T = (\beta_1, \beta_2, \ldots, \beta_{n-1})$, ahol minden $1 \leq i \leq n-1$ esetén*

$$\beta_i \leq \sum_{v_j \in N^+(v_i) \cap C} w_e(v_i, v_j),$$

ha $v_i \in C$, valamint $\beta_i \leq 0$, ha $v_i \notin C$. Ha C klikk G-ben, akkor $(\underline{\alpha}^T, \underline{\beta}^T)^T$ megengedett megoldása (6)*-nak.*

Bizonyítás. Legyen C klikk G-ben, és tegyük fel indirekt módon, hogy $(\underline{\alpha}^T, \underline{\beta}^T)^T$ nem megengedett megoldás. A 3.1. Állításból következik, hogy a (4.3) és (4.4) egyenlőtlenségek teljesülnek, tehát az (5.2) feltételek között létezik olyan $1 \leq i \leq n$, amire

$$C_i \alpha_i + \sum_{v_j \in \overline{N}(v_i)} \alpha_j \geq C_i + 1.$$

Két esetet különböztetünk meg aszerint, hogy $\alpha_i = 0$ vagy $\alpha_i = 1$.

1. eset: Ha $\alpha_i = 0$, akkor a következő egyenlőtlenséget kapjuk:

$$\sum_{v_j \in \overline{N}(v_i)} \alpha_j \geq C_i + 1.$$

$\overline{\omega}_i \leq C_i$ miatt

$$\sum_{v_j \in \overline{N}(v_i)} \alpha_j \geq \overline{\omega}_i + 1.$$

Tekintsük a $C \cap \overline{N}(v_i)$ halmazt. Mivel C klikk, a halmaz részhalmazai is klikket alkotnak, és az elemek száma nem nagyobb, mint az $\overline{N}(v_i)$ által kifeszített G-beli részgráfban található maximum klikk mérete: $|C \cap \overline{N}(v_i)| \leq \overline{\omega}_i$. $|C \cap \overline{N}(v_i)| = \sum_{v_j \in \overline{N}(v_i)} \alpha_j$, ezért azt kapjuk, hogy

$$\sum_{v_j \in \overline{N}(v_i)} \alpha_j \leq \overline{\omega}_i.$$

Ellentmondásra jutottunk.

2. eset: Ha $\alpha_i = 1$, akkor a következő egyenlőtlenség adódik:

$$C_i + \sum_{v_j \in \overline{N}(v_i)} \alpha_j \geq C_i + 1.$$

Mindkét oldalból kivonva C_i-t

$$\sum_{v_j \in \overline{N}(v_i)} \alpha_j \geq 1$$

egyenlőtlenséget kapjuk. Ekkor létezik $1 \leq k \leq n$, amire $\alpha_k = 1$ és $v_k \in \overline{N}(v_i)$, azaz $\{v_i, v_k\} \notin E$. Mivel $\alpha_i = 1$ és $\alpha_k = 1$, ezért $v_i \in C$ és $v_k \in C$. Mindkét csúcs eleme C-nek, ezért $\{v_i, v_k\} \in E$. Ellentmondásra jutottunk.

Mindkét esetben ellentmondásra jutottunk, ezzel az állítást beláttuk. □

4.2. Állítás. *Ha* $(\underline{\alpha}^T, \underline{\beta}^T)^T$ *megengedett megoldása* (6)*-nak, és* $\underline{\alpha}^T$ *a* $C \subseteq V$ *karakterisztikus vektora, akkor* C *klikk* G*-ben.*

Bizonyítás. Legyen $(\underline{\alpha}^T, \underline{\beta}^T)^T$ megengedett megoldása (6)-nak, és tegyük fel indirekt módon, hogy C nem klikk G-ben. Ekkor létezik $1 \leq i, k \leq n$, $i \neq k$, amire $v_i, v_k \in C$, de $\{v_i, v_k\} \notin E$. Következik, hogy $\alpha_i = 1$, $\alpha_k = 1$ és $v_k \in \overline{N}(v_i)$. $\underline{\alpha}^T$ megengedett megoldás, ezért

$$C_i \alpha_i + \sum_{v_j \in \overline{N}(v_i)} \alpha_j \leq C_i$$

feltétel teljesül. Behelyettesítve α_i és α_k értékeit azt kapjuk, hogy

$$C_i + 1 + \sum_{v_j \in \overline{N}(v_i), j \neq k} \alpha_j \leq C_i.$$

Átrendezve az egyenlőtlenséget

$$\sum_{v_j \in \overline{N}(v_i), j \neq k} \alpha_j \leq -1$$

adódik. Ellentmondásra jutottunk, mivel az összeg minden tagja nemnegatív. Ezzel az állítást beláttuk. □

További lehetőség a javításra, hogy a csúcsok színezésére más eljárásokat alkalmazunk, amelyek jobb korlátokat adnak. Mi a programokban csak a mohó színezőt használtuk.

4.2. Az (5.2) korlátban a C_i megválasztása lokális élszínezéssel

Ebben az esetben is a (3.2) feltételeket módosítjuk, most azonban nem a csúcsokat színezzük, hanem az éleket. Ehhez tekintsük minden v_i csúcs esetén a v_i nemszomszédaiból álló részgráfokat G-ben. Vegyük egy adott részgráf összes élét, és színezzük őket a következő módon: az első él az 1 színt kapja. A második él esetén vizsgáljuk meg, hogy van-e közös csúcsa az első éllel. Közös csúcsból legfeljebb egy lehet, így két eset fordulhat elő: egy közös csúcs van, vagy nincs közös csúcs. Legyen az első él két csúcsa a_1 és a_2, a második él csúcsai pedig b_1 és b_2. Lehetséges esetek:

1. Egy közös csúcs van. Tegyük fel, hogy $a_1 = b_1$ (az általánosság megsértése nélkül feltehetjük). Ha a_2 és b_2 között fut él, akkor a második él színe 2, és azt mondjuk, hogy a két él klikket alkot, egyébként 1.

2. Nincs közös csúcs. Ha $\{a_1, a_2, b_1, b_2\}$ egy 4-klikk G-ben, akkor a második él színe 2, és azt mondjuk, hogy a két él klikket alkot, egyébként 1.

Tegyük fel, hogy az első k élt kiszíneztük s színnel. A $(k+1)$-edik él színének meghatározásához végezzük el a fenti vizsgálatot a korábbi élekkel, és tekintsük azt a legkisebb színszámú színosztályt, amelyben a $(k+1)$-edik él semelyik éllel nem alkot klikket. Ha van ilyen, akkor a $(k+1)$-edik él is bekerül ebbe a színosztályba, ha nincs, akkor a színe legyen $s+1$. Ezzel az éleknek egy élesített jó színezését kapjuk. Egy klikk összes éle különböző színű lesz, így az eljárás végén az előforduló legnagyobb színszám felső becslést ad a gráfban található C maximum klikk éleinek számára. Ebből a C csúcsainak számára vonatkozó becslést könnyen megkaphatjuk. Ha az élekre vonatkozó becslés M, akkor a $\binom{|C|}{2} \le M$ egyenlőtlenségből a $|C|$ értékére vonatkozó felső becslés másodfokú egyenlet megoldásából adódik: $|C| \le \left[\frac{1+\sqrt{1+8M}}{2}\right]$.

Színezzük ki minden csúcs esetén a csúcsok nemszomszédaiból alkotott részgráfok éleit a fenti színezéssel. Jelölje K_i a v_i nemszomszédai által kifeszített részgráf esetén kapott felső becslést a részgráfban található maximum klikk méretére.

A Croce-Tadei feltételek i-edik egyenlőtlenségét ebben az esetben is úgy javíthatjuk, ha $\overline{N}_i$ helyére K_i-t írunk:

$$K_i x_i + \sum_{v_j \in \overline{N}(v_i)} x_j \le K_i, \quad v_i \in V \tag{8}$$

A (4) vegyes-egészértékű program (4.2) feltételét helyettesítsük a (8) feltétellel. Az így kapott programot jelöljük (9)-cel. Ezt megtehetjük, és az erre vonatkozó bizonyításokat megkapjuk, ha a 4.1. és a 4.2. Állításokban és bizonyításaikban a C_i jelöléseket lecseréljük K_i-re.

Élszínezéssel erősebb korlátokat kaptunk, mint csúcsszínezéssel, viszont ebben az esetben az összes élt össze kellett hasonlítanunk, ezért hosszabb volt a futásidő. Ha egy G gráf csúcsai k színnel jól színezhetők, akkor G élei élesítetten jól színezhetők $\binom{k}{2}$ színnel. Ebből az adódik, hogy van olyan élesített élszínezés, amely nem ad rosszabb felső korlátot a klikkméretre, mint a csúcsok jó színezése. Természetesen egy éleket színező mohó algoritmus adhat rosszabb felső korlátot a klikkméretre, mint egy szerencsésebb mohó csúcsszínező. Az, hogy az élszínezés tipikusan jobb korlátokat ad, mint a csúcsszínezés, egy empirikus tény, amit a példafuttatások során tapasztaltunk.

4.3. Az (5.3) korlátban az M_i választása

A lokális csúcs- és élszínezést nemcsak a Croce-Tadei egyenlőtlenségeknél alkalmazhatjuk, hanem a vegyes-egészértékű program

$$z_i \leq U_i x_i, \quad v_i \in V \setminus \{v_n\}$$

alakú feltételei esetén is.

$N^+(v_i)$ jelöli azokat a szomszédos csúcsokat, amelyek indexe i-nél nagyobb, U_i pedig a v_i-ből $N^+(v_i)$-be érkező élek súlyának összegét. Színezzük ki az $N^+(v_i)$ csúcsok által kifeszített G-beli részgráf csúcsait jól vagy éleit élesített jó színezéssel. Ha v_i szerepel egy C klikkben, akkor az i-nél nagyobb indexű szomszédai közül azok, amelyek szintén szerepelnek C-ben, klikket alkotnak egymással. Tegyük fel, hogy a színezéssel kapott becslés a részgráfban található maximum klikk méretére L_i, ami azt jelenti, hogy az $N^+(v_i)$ halmaz elemeiből legfeljebb L_i szerepelhet C-ben. Ezért tekintsük a v_i-ből induló $N^+(v_i)$ csúcsokba érkező élek súlyai közül az első L_i darab legnagyobb súly összegét, ezt jelölje M_i. Mivel $M_i \leq U_i$, ezért U_i helyére M_i-t írva szorosabb korlátokat kapunk. A módosított feltétel:

$$z_i \leq M_i x_i, \quad v_i \in V \setminus \{v_n\}.$$

Legyen C_i^+ az $N^+(v_i)$ csúcsok által kifeszített G-beli részgráfban található maximum klikk, és jelölje s_i a v_i-ből az C_i^+ csúcsokba futó élek súlyának összegét. $|C_i^+| \leq L_i$ és $s_i \leq M_i$.

4.3. Állítás. *A (7) vegyes-egészértékű program optimális megoldása megadja a maximális élsúlyú klikket G-ben.*

Bizonyítás. Azt kell megmutatnunk, hogy a (4.3) egyenlőtlenséget lecserélhetjük (5.3)-ra.

Legyen C maximális élsúlyú klikk G-ben. A (4) vegyes-egészértékű program C-hez tartozó optimális megoldásában

$$z_i = \begin{cases} \displaystyle\sum_{v_j \in N^+(v_i) \cap C} w_e(v_i, v_j), & \text{ha } v_i \in C, \\ 0, & \text{ha } v_i \notin C. \end{cases}$$

$|N^+(v_i) \cap C| \leq |C_i^+| \leq L_i \leq |N^+(v_i)|$, ezért

$$\sum_{v_j \in N^+(v_i) \cap C} w_e(v_i, v_j) \leq s_i \leq M_i \leq U_i.$$

Ebből következik, hogy a $z_i \leq U_i x_i$ feltételt valóban lecserélhetjük a $z_i \leq M_i x_i$ egyenlőtlenségre. Ezzel az állítást beláttuk. □

4.4. Az (5.4) korlátban az S választása

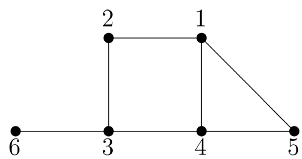
A programokat tovább javíthatjuk, ha az eredeti gráfot is kiszínezzük. Ezt globális színezésnek nevezzük. A kapott értéket jelöljük S-sel. Ez felső becslést ad a gráfban található maximum klikk méretére, ezért a korábbi programokhoz hozzávehetjük a

$$\sum_{i=1}^{n} x_i \leq S$$

feltételt. A tesztekben globális élszínezést használtunk a plusz feltétel megfogalmazásához.

4.5. Példa

Tekintsük a következő G gráfot:



1. ábra. G gráf

G-nek 6 csúcsa és 7 éle van. Az él átfogalmazásból adódó egészértékű program a maximum klikk problémára 6 változót és 8 feltételt tartalmaz. A változók 0-1 változók.

	x_1	x_2	x_3	x_4	x_5	x_6		
	1	1	1	1	1	1	$\rightarrow$	max
(1)	1		1				$\leq$	1
(2)	1					1	$\leq$	1
(3)		1		1			$\leq$	1
(4)		1			1		$\leq$	1
(5)		1				1	$\leq$	1
(6)			1		1		$\leq$	1
(7)				1		1	$\leq$	1
(8)					1	1	$\leq$	1

1. táblázat. Él átfogalmazás

A Croce-Tadei átfogalmazásból kapott egészértékű programot a 2. táblázat, a módosított Croce-Tadei átfogalmazásból (a (3) program a (3.2) korlát helyett az

(5.2) feltétellel) kapott egészértékű programot pedig a 3. táblázat tartalmazza. A programokban 6 változó és 6 feltétel szerepel. A változók 0-1 változók.

	x_1	x_2	x_3	x_4	x_5	x_6		
	1	1	1	1	1	1	$\to$	max
(1)	2		1			1	$\leq$	2
(2)		3		1	1	1	$\leq$	3
(3)	1		2		1		$\leq$	2
(4)		1		2		1	$\leq$	2
(5)		1	1		3	1	$\leq$	3
(6)	1	1		1	1	4	$\leq$	4

2. táblázat. Croce-Tadei átfogalmazás

	x_1	x_2	x_3	x_4	x_5	x_6		
	1	1	1	1	1	1	$\to$	max
(1)	2		1			1	$\leq$	2
(2)		2		1	1	1	$\leq$	2
(3)	1		2		1		$\leq$	2
(4)		1		1		1	$\leq$	1
(5)		1	1		2	1	$\leq$	2
(6)	1	1		1	1	3	$\leq$	3

3. táblázat. Módosított Croce-Tadei átfogalmazás

A megengedett megoldások halmaza a három egészértékű program esetén azonos. Az $x_i \in \{0, 1\}$ változókat a $0 \leq x_i \leq 1$ valós változókkal helyettesítve az egészértékű programok lineáris relaxációit kapjuk. A kérdés az, hogy a lineáris relaxációk megengedett megoldásainak A_1, A_2, A_3 halmazai hogyan viszonyulnak egymáshoz.

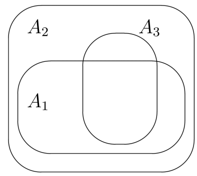
A Croce-Tadei átfogalmazás minden feltétele az él átfogalmazás bizonyos feltételeinek összege, amiből következik, hogy $A_1 \subseteq A_2$. Egy egészértékű program feltételei megfeleltethetők hipersíkoknak. Kiszámítva a koordinátatengelyek és a hipersíkok metszetét, arra a következtetésre juthatunk, hogy $A_3 \subseteq A_2$.

Legyen $P = (p, p, p, p, p, p)$ egy tesztpont. Ha $p = 0.5$, akkor láthatjuk, hogy $P \in A_1, P \in A_2, P \notin A_3$.

Legyen $Q = (0, q, 0, 0, q, 0)$ szintén egy tesztpont. Ha $q = 0.75$, akkor láthatjuk, hogy $Q \notin A_1$, $Q \in A_2$, $Q \notin A_3$. Ha $q = 0.66$, akkor $Q \notin A_1$, $Q \in A_2$, $Q \in A_3$. Végül, ha $q = 0.5$, akkor $Q \in A_1$, $Q \in A_2$, $Q \in A_3$.

A $q = 0.66$ és a $q = 0.75$ választások mutatják, hogy az él átfogalmazás $x_2 + x_5 \leq 1$ feltétele nem következhet sem az eredeti, sem a módosított Croce-Tadei átfogalmazásból.

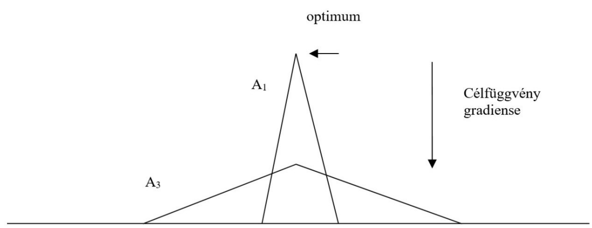
A lineáris relaxációk megengedett megoldásainak halmazai közötti kapcsolatot a következő ábra szemlélteti:



2. ábra. A megengedett megoldások halmazai

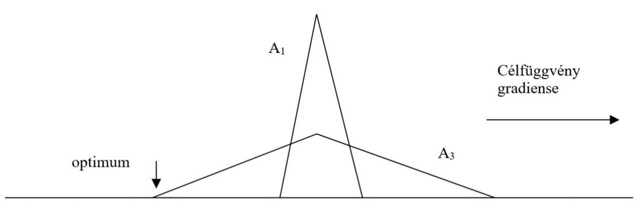
A 2. ábra nemcsak a példára, hanem általánosan is igaz. A következő ábrák azt mutatják, hogy bizonyos célfüggvényekre az él átfogalmazás, míg más célfüggvényre a módosított Croce-Tadei átfogalmazás ad jobb felső korlátot.

Az alábbi helyzetben az él átfogalmazás optimum értéke nagyobb, mint a módosított Croce-Tadei átfogalmazásban.



3. ábra. A célfüggvény gradiense vertikális

Ha a célfüggvény gradiense horizontális, akkor fordul a helyzet. A Croce-Tadei átfogalmazás optimum értéke nagyobb, mint az él átfogalmazás optimuma.



4. ábra. A célfüggvény gradiense horizontális

A maximum klikk probléma él és Croce-Tadei átfogalmazásának megengedett megoldásai általában is úgy viszonyulnak egymáshoz, mint a példa esetében. Ha egy csúcssúlyozott maximum klikk problémát tekintünk, vagyis egy $c_1x_1 + \ldots + c_nx_n$ alakú célfüggvényt alkalmazunk, akkor a $c_1, \ldots, c_n$ együtthatók válaszhatók úgy, hogy az él átfogalmazás adja a kisebb optimumot, de válaszhatók úgy is, hogy a Croce-Tadei átfogalmazáshoz tartozik a kisebb optimum.

A maximális élsúlyú klikk probléma esetében a $z_1, \ldots, z_{n-1}$ új változók miatt a geometriai kép megváltozik. A két átfogalmazás összehasonlítására nem marad más eszközünk, mint numerikus kísérletek eredményeinek összehasonlítása. Ezen az alapon a Croce-Tadei átfogalmazás jobb becsléseket ad, mint az él átfogalmazás. Ez egy megfigyeléseken alapuló észrevétel, amelynek van a gyakorlatban is használható jelentősége, de nem szeretnénk matematikai értelemben vett bizonyított eredmény szintjére emelni.

5. Számítási eredmények

A módosított programokat a [6] cikkben található DIMACS gráfokon teszteltük. Ezek a gráfok nem élsúlyozottak, ezért a cikk szerzőihez hasonlóan az éleket a $w_e(v_i, v_j) = (i + j) \bmod 200 + 1$ módszerrel súlyoztunk, bár a [2] munka kritizálja ezt a gyakorlatot. Az eredményeket a 6. táblázatban foglaltuk össze. A táblázatban az eredeti és a módosított programok lineáris relaxációinak optimum értékei szerepelnek. Az egyes oszlopok jelentését a 4. táblázat tartalmazza.

$\lvert V\rvert$	csúcsok száma
$\lvert E\rvert$	élek száma
d	élsűrűség
optimális súly	a maximális élsúlyú klikk élsúlyainak összege
LR	az eredeti (4) vegyes-egészértékű program lineáris relaxációjának optimuma
LP1	a (6) vegyes-egészértékű program lineáris relaxációjának optimuma
LP2	a (9) vegyes-egészértékű program lineáris relaxációjának optimuma
LP3	a (9) vegyes-egészértékű program lineáris relaxációjának optimuma kiegészítve az (5.5) feltétellel
LP4	az (5) vegyes-egészértékű program lineáris relaxációjának optimuma az (5.2) feltétel helyett a (8) feltétellel

4. táblázat. Az oszlopok jelentése a 6. táblázatban

Az LR oszlopot összehasonlítva az LP1, LP2, LP3 és LP4 oszlopokkal láthatjuk, hogy jelentősen csökkentek az optimum értékek, minél több helyen módosítottuk a programot, annál jobb becsléseket kaptunk. Ez azonban azzal járt, hogy tovább tartott az egyes feltételek meghatározása.

Az optimum értékek csökkenésétől azt reméljük, hogy ez futásidő megtakarításhoz vezet. Az 5. táblázatban összefoglalt eredmények megerősítik ezt a várakozásunkat. Egészértékű programokkal kerestünk maximum klikket tesztgráfokban. A futásidőket az utolsó kettő oszlop tartalmazza. EP1 jelöli az (1) programmal (él átfogalmazás), kapott értékeket, és EP2 a módosított Croce-Tadei átfogalmazással (a (3) program a (3.2) korlát helyett az (5.2) feltétellel) kapott eredményeket.

gráf	$\lvert V \rvert$	$\lvert E \rvert$	$\omega(G)$	EP1 (s)	EP2 (s)
brock200_4	200	13 089	17	1 264.57	250.96
c-fat200-2	200	3 235	24	108.58	1.20
c-fat500-2	500	9 139	26	3 920.36	88.95
hamming6-4	64	704	4	4.71	1.28
hamming8-2	256	31 616	128	0.70	0.56
johnson8-2-4	28	210	4	0.57	0.64
johnson8-4-4	70	1 855	14	0.87	0.72
johnson16-2-4	120	5 460	8	176.47	0.57
keller4	171	9 435	11	198.16	63.67
MANN_a9	45	918	16	0.77	0.68

5. táblázat. Futásidők

6. Összefoglalás

Jelen munkában megmutattuk, hogy a maximális élsúlyú klikk problémát megoldó (4) vegyes-egészértékű program módosítására milyen lehetőségeink vannak csúcs- és élszínezés alkalmazásával, továbbá hogy az általunk tesztelt gráfok esetén az új programok lineáris relaxációinak optimumai hogyan közelítették az eredeti optimumokat. A kutatás folytatására további lehetőség a programok módosítása más gráfszínező eljárásokkal, kernelizálással és kombinatorikus alapú metszősíkokkal.

Köszönetnyilvánítás

Jelen munka az Innovációs és Technológiai Minisztérium ÚNKP-19-2 kódszámú Új Nemzeti Kiválóság Programjának szakmai támogatásával készült.

gráf	$\lvert V \rvert$	$\lvert E \rvert$	d	optimális súly	LR	LP1	LP2	LP3	LP4
brock200_1	200	14 834	0.7454	21 230	745 274	404 970	325 396	291 002	157 715
brock200_2	200	9 876	0.4963	6 542	494 967	165 306	98 245	89 333	33 107
brock200_3	200	12 048	0.6054	10 303	604 606	251 884	174 825	157 365	69 594
brock200_4	200	13 089	0.6577	13 967	657 976	299 278	219 071	197 618	92 811
c-fat200-1	200	1 534	0.0771	7 734	77 034	11 260	11 260	11 260	10 506
c-fat200-2	200	3 235	0.1626	26 389	162 735	40 564	40 564	40 259	37 558
c-fat200-5	200	8 473	0.4258	168 200	425 773	284 558	284 558	248 360	230 283
c-fat500-1	500	4 459	0.0357	10 738	224 609	15 787	15 787	15 787	14 707
c-fat500-2	500	9 139	0.0733	38 350	460 288	55 986	55 986	55 986	51 941
c-fat500-5	500	23 191	0.1859	205 864	1 167 941	316 807	316 807	303 693	284 812
c-fat500-10	500	46 627	0.3738	804 000	2 348 177	1 345 450	1 345 450	1 188 530	1 112 891
DSJC500_5	500	62 624	0.5020	9 626	3 154 650	891 040	481 241	443 642	120 271
hamming6-2	64	1 824	0.9048	32 736	60 192	60 192	60 192	60 192	37 926
hamming6-4	64	704	0.3492	396	23 232	6 776	4 130	2 905	811
hamming8-2	256	31 616	0.9686	800 624	1 558 464	1 558 464	1 558 464	1 558 464	1 148 298
hamming8-4	256	20 864	0.6392	12 360	1 021 056	302 535	218 089	184 080	47 902
johnson8-2-4	28	210	0.5556	192	3 150	900	900	900	286
johnson8-4-4	70	1 855	0.7681	6 552	66 780	26 843	26 712	26 712	8 700
johnson16-2-4	120	5 460	0.7647	3 808	308 860	41 377	41 377	41 377	6 819
keller4	171	9 435	0.6491	6 745	483 595	168 637	122 228	104 061	32 049
MANN_a9	45	918	0.9273	5 460	21 654	17 323	17 323	17 323	7 872

6. táblázat. A lineáris programok optimum értékei

Hivatkozások

[1] Croce, F. D. and Tadei, R.: *A multi-KP modeling for the maximum-clique problem*, European Journal of Operational Research, Vol. **73** No. **3**, pp. 555-561 (1994). DOI: 10.1016/0377-2217(94)90252-6

[2] McCreesh, C., Prosser, P., Simpson, K. and Trimble, J.: *On Maximum Weight Clique Algorithms, and How They Are Evaluated*, In: Beck J. (eds) Principles and Practice of Constraint Programming. CP 2017. Lecture Notes in Computer Science, Springer, Cham, Vol. **10416** (2017). DOI: 10.1007/978-3-319-66158-2_14

[3] Moon, J. W. and Moser, L.: *On cliques in graphs*, Israel J. Math., Vol. **3**, pp. 23-28 (1965). DOI: 10.1007/BF02760024

[4] Östergård, P. R.: *A new algorithm for the maximum-weight clique problem*, Electronic Notes in Discrete Mathematics, Vol. **3**, pp. 153-156 (1999). DOI: 10.1016/S1571-0653(05)80045-9

[5] Pullan, W.: *Approximating the maximum vertex/edge weighted clique using local search*, Journal of Heuristics, Vol. **14**, pp. 117-134 (2008). DOI: 10.1007/s10732-007-9026-2

[6] Shimizu, S., Yamaguchi, K. and Masuda, S.: *A Branch-and-Bound Based Exact Algorithm for the Maximum Edge-Weight Clique Problem*, In: Lee R. (eds) Computational Science/Intelligence & Applied Informatics. CSII 2018. Studies in Computational Intelligence, Springer, Cham, Vol. **787** (2019). DOI: 10.1007/978-3-319-96806-3_3

[7] Shimizu, S., Yamaguchi, K. and Masuda, S.: *A maximum edge-weight clique extraction algorithm based on branch-and-bound*, Discrete Optimization, Vol. **37**, paper: 100583 (2020). DOI: 10.1016/j.disopt.2020.100583

[8] Shimizu, S., Yamaguchi, K. and Masuda, S.: *Mathematical programming formulation for the maximum edge-weight clique problem*, IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences (in Japanese), Vol. **J100-A** No. **8**, pp. 313-315 (2017).

Szabó Sándor 1954-ben született Karcagon. Az Eötvös Loránd Tudományegyetemen tanult matematikát és fizikát. Jelenleg a Pécsi Tudományegyetem Matematikai és Informatikai Intézetében dolgozik. Fő érdeklődési területe a diszkrét matematika és alkalmazásai.

Szabó Sándor
Pécsi Tudományegyetem
Alkalmazott Matematika Tanszék
7624 Pécs, Ifjúság útja 6.
szabos@ttk.pte.hu

Sztojkovics Dóra 1996-ban született Pécsett. 2015-ben kezdte meg tanulmányait matematika alapszakon a Pécsi Tudományegyetemen. A 34. Országos Tudományos Diákköri Konferencián első helyezést ért el Tanulás- és Tanításmódszertani - Tudástechnológiai Szekcióban. Következő tanévben elnyerte az ÚNKP ösztöndíjat, és új kutatási területe a diszkrét matematika lett. 2020-ban alkalmazott matematikus diplomát szerzett.

Sztojkovics Dóra
Pécsi Tudományegyetem
7624 Pécs, Ifjúság útja 6.
sztojkovics@gmail.com

LINEAR PROGRAMS FOR THE EDGE WEIGHTED MAXIMUM CLIQUE PROBLEM

Sándor Szabó, Dóra Sztojkovics

Finding an edge weighted maximum clique in a given graph is an interesting problem and has many important applications. An earlier work [6] claims that a solver based on combinatorial considerations and exhaustive search outperforms solvers based on linear and quadratic programming. In this paper we propose modifications in the linear program reformulation of the edge weighted maximum clique problem. The modifications are based on coloring the nodes and edges of the graph. In order to assess the performance of the programs we carried out numerical experiments.