Domináló csúcsok szerepe hálózati folyamatok tervezésében

„Doktori értekezés tézisei”

Blázsik Zoltán

Szegedi Tudományegyetem
Szeged, 2008

1. Bevezetés

Hálózati problémák elemzése mind gyakorlati, mind matematikai szempontból figyelemreméltó eredményeket hozhat. Téziseim egy áttekintést adnak főbb eredményeimről három témában. Először a folyamatszintézis (PNS) problémában elért eredményeimet foglalom össze. A 8. fejezet de Bruijn gráfok perfekt domináló halmazairól szól. Két, korábban nyitott kérdésre adok választ. A harmadik téma az új HPPIT probléma. A tézicsoportokról az összefoglalás után részletesen írok, a kapcsolódó publikációk megadásával.

2. A PNS probléma matematikai modellje

Legyen M véges, nemüres halmaz, az anyagok halmaza. Legyen $O \subseteq \varphi'(M) \times \varphi'(M)$ nemüres halmaz a *műveleti egységek* halmaza, ahol $\varphi'(M)$ az M nemüres részhalmazainak halmazát jelenti. Egy $u = (\alpha, \beta) \in O$, műveleti egységre *(gép)* α és β rendre a *bemeneti* és *kimeneti halmaz*. Az (M, O) pár a *folyamat gráf* vagy P-*gráf*. Ennek az irányított kétrészes gráfnak a csúcshalmaza $M \cup O$, az élhalmaza $A = A_1 \cup A_2$ pedig két típusú $A_1 = \{(X, Y) : Y = (\alpha, \beta) \in O \text{ és } X \in \alpha\}$ és $A_2 = \{(Y, X) : Y = (\alpha, \beta) \in O \text{ és } X \in \beta\}$. Ha az $X_1, X_2, ..., X_n$, csúcsokra léteznek az $(X_1, X_2), (X_2, X_3), \ldots, (X_{n-1}, X_n)$ élek a P-gráfban, akkor $[X_1, X_n]$ jelölje ezt az *utat* az X_1-től az X_n-ig. Legyen az (m, o) és az (M, O) adott folyamat gráfok; (m, o) *részgráfja* az (M, O)-nak, ha $m \subseteq M$, $o \subseteq O$ és $o \subseteq \varphi'(m) \times \varphi'(m)$. Az O elemeire mint absztrakt átalakítókra gondolhatunk, amelyek előállítják a β halmazt az α-ból.

A PNS *strukturális modelljét* a következő módon definiáljuk. Jelölje M' a rendelkezésre álló anyagok halmazát, $P \subseteq M'$ az előállítani kívánt, *céltermék* anyagok halmazát és $R \subseteq M'$ a *nyersanyagok* $(P \cap R = \emptyset)$ halmazát. A PNS strukturális modellje alatt a $\mathbf{M} = (P, R, O)$ hármast értjük. Az (M, O), folyamat gráfban $M = \cup\{\alpha \cup \beta : (\alpha, \beta) \in O\} \cup P \cup R$ az anyagok halmaza. (M, O) ábrázolható mint a P-gráf, amely mutatja a műveleti egységek közötti kapcsolatokat. Minden olyan részfolyamat, amely segítségével a P anyaghalmazt elő lehet állítani az R nyersanyaghalmazból az O bizonyos elemeit felhasználva, az (M, O) egy részgráfja. A műveleti egységekről további tulajdonságokat nem ismerünk, pl. az előállítás folyamatáról, az anyagi egyensúlyról. Kombinatorikus jellemzőket állapíthatunk meg azokról a részgráfokról, amelyek rendelkeznek a fenti tulajdonsággal. Az alapvető közös ismérveket tekinti át pl. a [18] munka.

Az (M, O) folyamat gráfnak az (m, o) részgráfját az $\mathbf{M} = (P, R, O)$ *lehetséges megoldásának* nevezzük, ha teljesülnek a következő tulajdonságok:

($\mathcal{A}1$) $P \subseteq m$,

($\mathcal{A}2$) $\forall X \in m$, $X \in R \Leftrightarrow$ nincs (Y, X) típusú él (m, o)-ban.

($\mathcal{A}3$) $\forall Y_0 \in o$, $\exists$ $[Y_o, Y_n]$ út, melyre $Y_n \in P$,

($\mathcal{A}4$) $\forall X \in m$, $\exists(\alpha, \beta) \in o$ melyre $X \in \alpha \cup \beta$.

Jelölje $S(\mathbf{M})$ az $\mathbf{M}$ lehetséges megoldásainak halmazát. Könnyű belátni, hogy $S(\mathbf{M})$ zárt a véges egyesítésre. Következésképpen,

$$\cup\{(m, o) : (m, o) \in S(\mathbf{M})\}$$

is egy lehetséges megoldás. Feltéve, hogy az $S(\mathbf{M}) \neq \emptyset$, ez a legbővebb lehetséges megoldás. Ezt a kitüntetett szerepű lehetséges megoldást és a neki megfelelő folyamat részgráfot az $\mathbf{M}$ *maximális struktúrájának* nevezzük. Érdemes megjegyezni, hogy a maximális struktúrának természetesen lehetnek olyan részgráfjai, amelyek nem lehetséges megoldások.

Az első kérdés a PNS modellel kapcsolatban éppen az, hogyan találhatunk egyáltalán lehetséges megoldást, hogyan találhatjuk meg egy strukturális modell maximális struktúráját? A [14] és [19] cikkekben mutattak be egyszerű, polinomidejű algoritmust a maximális struktúra generálására, ha az létezik.

Legyen adott egy $\mathbf{M} = (P, R, O)$ strukturális modell, valamint legyen z egy pozitív valós értékű súlyfüggvény az $\mathrm{S}(\mathbf{M})$ halmazon. Az első optimalizálási modellünk ekkor

$$\min\{z((m, o)) \ : \ (m, o) \in S(\mathbf{M})\}. \tag{1}$$

Nyilvánvaló a legolcsóbb előállítási folyamat megtalálásának fontossága. Az első megoldásokat globális optimalizálási eljárásokkal adták a fenti PNS probléma körébe tartozó konkrét problémákra (l. [13] és [22]). Kombinatorikus megfontolásokon is alapultak további módszerek, (l. pl. a [14], [18], [21]) cikkeket. Ezek a megoldási módszerek nagyon bonyolultak lehetnek általában.

Tegyük fel, hogy csak a műveleti egységeknek van költsége. Legyen $w : O \to R_+$ egy adott költségfüggvény az O halmazon. Legyen a célfüggvényünk egyszerűen a lehetséges megoldásban működő gépek összköltsége

$$\min\{\sum_{u \in o} w(u) \ : \ (m, o) \in S(\mathbf{M})\}. \tag{2}$$

3. A PNS probléma NP–teljes

A PNS elméletében nagyon fontos alapkérdés volt, vajon a (2) kombinatorikus optimalizálási probléma P-ben van-e vagy nem. Azt könnyű ellenőrizni, hogy $(m, o) \in S(\mathbf{M})$ teljesül-e, sőt azt is, hogy az értéke egy adott számnál kisebb-e. Ennek következtében (2) az NP bonyolultsági osztályban van. Bebizonyítottuk [2], hogy (2) NP–nehéz.

Megtartva az eredeti cikk jelölését (2)-re mint egy PNS$_w$-*problémára* hivatkozunk, a PNS probléma osztályt PNS$_w$-vel jelöljük. Definiáltunk PNS$_w$-nek egy részosztályát, amelyre (2) ekvivalens a klasszikus halmazlefedési problémával.

Jelölje PNS$_{w1}$ a PNS$_w$ azon részosztályát, melyre $O \subseteq \wp'(R) \times \wp'(P)$ teljesül. A részosztály szemléletesen olyan $\mathbf{M} = (P, R, O)$ strukturális modellel adható meg, amelyben minden műveleti egység csak nyersanyagot használ fel és csak célterméket gyárt. A gépek egymástól függetlenek, párhuzamosan működhetnek. A [2] cikkben megmutattuk, hogy

3.1. Tétel. *A PNS_{w1} PNS osztály ekvivalens a halmazlefedési problémával.*

Abban az esetben ha egy PNS$_{w1}$-problémára még az is teljesülne, hogy bármely kívánt anyagot legfeljebb egy gép állít elő, akkor az ekvivalens probléma egy halmazosztályozási probléma lenne. Mivel mind a halmazlefedési, mind a halmazosztályozási probléma jól ismert NP–teljes probléma, így a PNS$_{w1}$-probléma szintén NP–teljes. Ebből következik, hogy

3.1. Következmény. *A (2) PNS probléma NP–nehéz.*

Figyeljük meg, hogy a PNS$_{w1}$-problémában az anyagok halmazát két diszjunkt részre oszthatjuk, a P-re és az R-re, és mindegyik műveleti egység bemenete az R egy nemüres részhalmaza, míg a P egy nemüres részhalmaza a kimenete. Általánosítva ezt az észrevételt a [2] cikkben további részosztályait definiáltuk a PNS$_w$-problémáknak. Nevezetesen, legyen $k \geq 1$ egy tetszőleges rögzített pozitív egész, melyre tekintsük azokat a feladatokat, melyekre $M = M_1 \bigcup \cdots \bigcup M_{k+1}$, ahol az $M_1, \ldots, M_{k+1}$ anyaghalmazok páronként diszjunkt, nemüres halmazok. Továbbá, legyen $O = O_1 \bigcup \cdots \bigcup O_k$, ahol $O_i \subseteq \wp'(M_1 \bigcup \cdots \bigcup M_i) \times \wp'(M_{i+1})$, $i = 1, \ldots, k$. Az ilyen PNS$_w$-problémákat hívjuk PNS$_{wk}$-*problémáknak*.

A fenti PNS osztályok motiválhatták a [20] és [27]) cikkeket. Ebben a két munkában bizonyítás található a másik irányra, nevezetesen a szerzők megmutatták, hogy (2) szintén egy speciális esete a halmazlefedési (halmazosztályozási) problémának. Így tehát a minsum változata a súlyozott PNS problémának egyszerre általánosítása és speciális esete a halmazlefedési problémának. Mivel (2) egy NP–teljes probléma, így jogosan kutathatjuk *jól-megoldható* speciális osztályait [28], [29], és *heurisztikus algoritmusokat* fejleszthetünk rá [8].

4. Korlátok a konzisztens döntési leképezések számára

Először különböző Korlátozás és Szétválasztás alapú technikákat alkalmaztak PNS problémák egzakt optimális megoldásának megtalálására. Ilyeneket javasoltak és elemeztek az [24] és [25] cikkekben, és bevezették a *döntési leképezés* fogalmát(l. [15]). Az (M, O) P-gráfja $\mathbf{M}$-nek meghatároz egy Δ függvényt, amely az $M \setminus R$ halmazból $\varphi'(O)$-ba képez. Tetszőleges $X \in M \setminus R$, anyagra legyen

$$\Delta(X) = \{(\alpha, \beta) : (\alpha, \beta) \in O \ \& \ X \in \beta\}.$$

Legyen továbbá m az $M \setminus R$ egy részhalmaza, valamint legyen $\delta(X)$ a $\Delta(X)$ egy részhalmaza minden $X \in m$ esetén. A δ függvény az m elemeihez az O bizonyos részhalmazait rendeli. Legyen $\delta[m] = \{(X, \delta(X)) : X \in m\}$, egy döntési leképezés, amely az $\mathbf{M}$-hez van hozzárendelve. A $\delta[m]$-et *konzisztensnek* mondjuk, ha $\delta(X) \cap \Delta(Y) \subseteq \delta(Y)$ teljesül minden $X, Y \in m$ esetén. Az $\mathbf{M}$ összes konzisztens döntési leképezéseinek halmazát jelölje $\Omega_{\mathbf{M}}$. Amennyiben $\delta[m] \in \Omega_{\mathbf{M}}$ és $m = M \setminus R$, akkor használhatjuk a rövidebb δ jelölést a $\delta[M \setminus R]$ helyett. Egy döntési leképezést mint döntések sorozatát képzelhetjük el. Mindegyik anyaghoz hozzárendeljük azokat a gépeket, amelyek elő tudnák állítani. A konzisztencia azt az egyszerű tényt fejezi ki, hogy ha az X miatt a $\delta(X)$ egyik elemeként valamely műveleti egység egy Y anyagot szintén előállít, vagyis $\delta(X) \cap \Delta(Y)$ eleme, akkor bele kell venni a $\delta(Y)$ halmazba is, vagyis, $\delta(Y) \supseteq \delta(X) \cap \Delta(Y)$.

Az *op* függvény legyen az $\Omega_{\mathbf{M}}$ halmazon értelmezve, tetszőleges $\delta[m] \in \Omega_{\mathbf{M}}$ esetén,

$$op(\delta[m]) = \cup\{\delta(X) : X \in m\}.$$

Vezessük be a következő függvényeket is, tetszőleges o géphalmazra, legyen

$$mat^{in}(o) = \cup_{(\alpha,\beta)\in o}\alpha, \qquad mat^{out}(o) = \cup_{(\alpha,\beta)\in o}\beta.$$

Legyen $\delta_1[m_1]$ és $\delta_2[m_2]$ konzisztens döntési leképezések. Ekkor, $\delta_2[m_2]$ egy *kiterjesztése* $\delta_1[m_1]$-nek, ha $m_1 \subseteq m_2$ és $\delta_1(X) = \delta_2(X)$ minden $X \in m_1$

esetén; jelölésben $\delta_1[m_1] \leq \delta_2[m_2]$. Ha $\delta_1[m_1] \leq \delta_2[m_2]$ és $m_1 \subset m_2$, akkor *valódi kiterjesztésről* beszélünk és $\delta_1[m_1] < \delta_2[m_2]$ írunk. A kiterjesztési reláció reflexív, antiszimmetrikus és tranzitív; tehát egy parciális rendezés a konzisztens döntési leképezések $\Omega_{\mathbf{M}}$ halmazán. Jelölje $\Omega_{\mathbf{M}}^{\max}$ a parciális rendezés maximális elemeinek halmazát.

4.1. Tétel. *Minden $\emptyset \neq m \subseteq M \setminus R$ esetén, az m döntési leképezéseinek száma $2^{\sum_{X \in m} |\Delta(X)|}$.*

Jelölje $\tau(m)$ a konzisztens döntési leképezések számát m-en.

4.2. Tétel. *Minden $\emptyset \neq m \subseteq M \setminus R$ esetén, $\tau(m) = 2^{|\cup\{\Delta(X) : X \in m\}|}$.*

4.1. Megjegyzés. *Amikor $m = M \setminus R$, akkor $\tau(m) = 2^{|O|}$. Ez az észrevétel mutatja, hogy szoros a kapcsolat a maximális konzisztens döntési leképezések és az O részhalmazai között. Valóban, belátható hogy a $\gamma(\delta) = op(\delta)$ kölcsönösen egyértelmű leképezése az $\Omega_{\mathbf{M}}^{\max}$ halmaznak a $\varphi(O)$-ba, az O hatványhalmazába.*

A maximális döntési leképezések és a lehetséges megoldások közötti kapcsolatra tekintettel definiáljuk a ρ leképezést a következő módon: Bármely $(m, o) \in S(\mathbf{M})$ lehetséges megoldásra, legyen $\rho(m, o) = \delta$, ahol δ a következő:

$$\delta(X) = \{u : u = (\alpha, \beta) \in o \;\&\; X \in \beta\}$$

minden $X \in M \setminus R$-ra. Belátható, hogy ρ is kölcsönösen egyértelműen beleképezi $S(\mathbf{M})$-et az $\Omega_{\mathbf{M}}^{\max}$ -ba. Emiatt a $2^{|O|}$ egy triviális felső korlátja $|S(\mathbf{M})|$-nak.

Ha a lehetséges megoldásokra vonatkozó axiómák közül akár csak az (A2)-t vesszük is figyelembe, ez a korlát javítható. Legyen $(m, o) \in S(\mathbf{M})$ egy tetszőleges lehetséges megoldás, és legyen $\rho(m, o) = \delta$. Ekkor, (A2)-ből következik az alábbi tartalmazási reláció:

$$(\mathcal{A}'2) \qquad mat^{in}(op(\delta)) \subseteq mat^{out}(op(\delta)) \cup R.$$

Jelölje $\tau'(m)$ azon konzisztens döntési leképezések számát m-en, amelyek kielégítik a $(\mathcal{A}'2)$ feltételt. Ekkor $\tau'(m) \geq |S(\mathbf{M})|$.

Legyen $O = \{u_1, \ldots, u_n\}$, $M = \{X_1, \ldots, X_k\}$, $O(X_j) = \{u : u = (\alpha, \beta) \in O \;\&\; X_j \in \alpha\}$, minden $X_j \in M$, és $j \in \{1, \ldots, k\}$ esetén. Legyen

$$A_j = \{\delta : \delta \in \Omega_{\mathbf{M}}^{\max} \;\&\; X_j \in mat^{in}(op(\delta)) \setminus (mat^{out}(op(\delta)) \cup R)\}$$

Ekkor, $(\mathcal{A}'2)$ nem érvényes δ-ra, mert az $X_j \in mat^{in}(op(\delta))$ és $X_j \notin mat^{out}(op(\delta)) \cup R$. Minden $\emptyset \neq I \subseteq \{1, \ldots, k\}$, indexhalmazra legyen $A_I = \cap_{i \in I} A_i$, és külön legyen $A_\emptyset = \Omega_{\mathbf{M}}^{\max}$. If $I = \{i_1, \ldots, i_l\}$, ekkor

$$A_I = \left\{\delta : \delta \in \Omega_{\mathbf{M}}^{\max} \ \& \ \{X_{i_1}, \ldots, X_{i_l}\} \subseteq mat^{in}(op(\delta)) \setminus (mat^{out}(op(\delta)) \cup R)\right\}$$

Számoljuk ki a $\tau'(m)$ értékét a Szitaformula segítségével. Így felső korlátot kapunk $|S(\mathbf{M})|$-ra.

4.3. Tétel. [3]
$\tau'(m) = |\Omega_{\mathbf{M}}^{\max} \setminus (A_1 \cup A_2 \cup ... \cup A_k)| = \Sigma_{I \subseteq \{1, \ldots k\}} (-1)^{|I|} \cdot |A_I|$.

4.2. Megjegyzés. *Érdemes kiemelni, hogy a kapott korlát független a modellben a kívánt anyagok P halmazától. Tetszőleges $P \subseteq M \setminus R$ esetén egyaránt érvényes.*

Sajnos azonban az $|A_I|$ kiszámítása bonyolult általában. Úgy képzelhetjük el, hogy általában le kell fedni az $\{X_{i_1}, \ldots, X_{i_l}\}$ halmazt az $\alpha_{j_1}, \ldots, \alpha_{j_s}$ rendszerrel, miközben az $(\alpha_{j_t}, \beta_{j_t}) \in O$, $t = 1, \ldots, s$, műveleti egységekre, $\{X_{i_1}, \ldots, X_{i_l}\} \cap \beta_{j_t} = \emptyset$, $t = 1, \ldots, s$ teljesül. Ekkor az $|A_I|$ egyenlő az ilyen fedőrendszerek számával.

5. Kiszámolható korlátok speciális PNS osztályok lehetséges megoldásainak számára

Speciális szerkezetű P-gráfok esetén az $|A_I|$ kiszámítása könnyebb. Egy érdekes speciális PNS családot kapunk, ha minden gép szeparátor típusú, vagyis $|\alpha| = 1$ teljesül minden $u = (\alpha, \beta) \in O$ esetén. Tekintsük ismét az $I = \{i_1, \ldots, i_l\}$ indexhalmazt. Legyen $O^*(X_{i_j}) = O(X_{i_j}) \setminus (\cup_{i \in I} \Delta(X_i))$. Ekkor, $O^*(X_{i_j})$ azon műveleti egységek halmaza, amelyek nem állítanak elő semmit $\{X_t : t \in I\}$-ből és mindegyiküknek szüksége van az X_{i_j} bemeneti anyagra.

5.1. Tétel. [3] *Szeparátor típusú műveleti egységek esetén*

$$|A_I| = \left(\prod_{t=1}^{l} \left(2^{|O^*(X_{i_t})|} - 1\right)\right) \cdot 2^{|O \setminus (\cup_{i \in I} \Delta(X_i)) \setminus (\cup_{i \in I} O(X_i))|}$$

Egy strukturális modell, amelyben szeparátor típusú gépek vannak az ún. *Egyenes modell*, vagyis legyen

$$u_1 = (\alpha_1, \beta_1) \text{ with } \alpha_1 = X_1 \text{ and } \beta_1 = X_2,$$

$$u_k = (\alpha_k, \beta_k) \text{ with } \alpha_k = X_k \text{ and } \beta_k = X_{k-1},$$

és általában:

$$u_i = (\alpha_i, \beta_i) \text{ with } \alpha_i = X_i \text{ and } \beta_i = \{X_{i-1}, X_{i+1}\}, (2 \leq i \leq k-1).$$

Másik, szimmetrikusabb modellünk az ún. *Lánc modell.* Csak két változtatásra van szükség, legyen:

$$\beta_1 = \{X_2, X_k\} \text{ and } \beta_k = \{X_{k-1}, X_1\}.$$

5.2. Tétel. [4] *Az Egyenes modellben* $|S(\mathbf{M})| \leq L^{(1)}$,

ahol

$$L^{(1)} = 2^k + \sum_{1 \leq j \leq \frac{k+1}{2}} (-1)^j \cdot \left[\sum_{\substack{0 \leq r \leq j-1 \\ k-3j+r+2 \geq 0}} \binom{j-1}{r} \cdot \binom{k-2j}{j-r-2} \cdot 2^{k-3j+r+2} + \right.$$
$$+ \sum_{\substack{0 \leq r \leq j-1 \\ k-3j+r+1 \geq 0}} \binom{j-1}{r} \cdot \binom{k-2j}{j-r-1} \cdot 2^{k-3j+r+1} +$$
$$\left. + \sum_{\substack{0 \leq r \leq j-1 \\ k-3j+r \geq 0}} \binom{j-1}{r} \cdot \binom{k-2j}{j-r} \cdot 2^{k-3j+r} \right].$$
$$= 1 + \sum_{2 \leq t \leq k} \sum_{1 \leq q \leq \min\{\frac{t}{2}; k-t+1\}} \binom{t-q-1}{q-1} \cdot \binom{k-t+1}{q}.$$

5.3. Tétel. [4] *A Lánc modellben* $|S(\mathbf{M})| \leq C^{(1)}$,

ahol

$$C^{(1)} = 2^k + \sum_{1 \leq j < \frac{k}{2}} (-1)^j \cdot \sum_{\substack{0 \leq r \leq j-1 \\ k-3j+r \geq 0}} \frac{k}{j} \cdot \binom{j}{r} \cdot \binom{k-2j-1}{j-r-1} \cdot 2^{k-3j+r} + e_k =$$

$$= 1 + \sum_{2 \leq t \leq k} \left[\sum_{1 \leq q \leq \min\{\frac{t}{2}; k-t\}} \binom{t-q-1}{q-1} \cdot \binom{k-t-1}{q-1} + \right.$$

$$+\sum_{2\le i\le t}\sum_{1\le q\le \frac{t-i}{2}+1}\left(\begin{array}{c} t-i-q+1 \\ q-1 \end{array}\right)\cdot\left(\begin{array}{c} k-t-1 \\ q-1 \end{array}\right)+$$

$$\left.+\sum_{1\le i\le k-t}\sum_{1\le q\le \min\{\frac{t}{2};k-t-i+1\}}\left(\begin{array}{c} t-q-1 \\ q-1 \end{array}\right)\cdot\left(\begin{array}{c} k-t-i \\ q-1 \end{array}\right)\right]+1,$$

ahol

$$e_k=\begin{cases} (-1)^{\frac{k}{2}}\cdot 2 & \text{, if } k \text{ páros,} \\ 0 & \text{, if } k \text{ páratlan.} \end{cases}$$

Az általános esetben amikor a Szitaformulával is nehéz számolni a direkt leszámolás is nehéz. A vizsgált két modellben sikerült Szitaformula nélkül is megadni a korlátot adó értékeket.

$$L^{(2)}=1+\sum_{2\le t\le k}\sum_{1\le q\le \min\{\frac{t}{2};k-t+1\}}\left(\begin{array}{c} t-q-1 \\ q-1 \end{array}\right)\cdot\left(\begin{array}{c} k-t+1 \\ q \end{array}\right).$$

$$C^{(2)}=1+\sum_{2\le t\le k}\left[\sum_{1\le q\le \min\{\frac{t}{2};k-t\}}\left(\begin{array}{c} t-q-1 \\ q-1 \end{array}\right)\cdot\left(\begin{array}{c} k-t-1 \\ q-1 \end{array}\right)+\right.$$

$$+\sum_{2\le i\le t}\sum_{1\le q\le \frac{t-i}{2}+1}\left(\begin{array}{c} t-i-q+1 \\ q-1 \end{array}\right)\cdot\left(\begin{array}{c} k-t-1 \\ q-1 \end{array}\right)+$$

$$\left.+\sum_{1\le i\le k-t}\sum_{1\le q\le \min\{\frac{t}{2};k-t-i+1\}}\left(\begin{array}{c} t-q-1 \\ q-1 \end{array}\right)\cdot\left(\begin{array}{c} k-t-i \\ q-1 \end{array}\right)\right]+1$$

Összegezve a két úton kapott eredményeket egy kombinatorikus azonossághoz jutottunk, $L^{(1)}=L^{(2)}$ és $C^{(1)}=C^{(2)}$.

6. A PNS bottleneck és k-összeg változatai

A [6] cikkben megmutattuk, hogy a PNS k-összeg változata jól-megoldható, legalábbis rögzített k-ra.

Legyen $\mathbf{M} = (P, R, O)$ egy PNS redukált strukturális modellje.

6.1. Bottleneck PNS-probléma

Elméleti és gyakorlati szempontból is fontos foglalkozni azzal a PNS problémával, amikor azt a lehetséges megoldást keressük, amikor a legköltségesebb műveleti egység a lehető legolcsóbb. Formálisan, meg szeretnénk oldani a (3) optimalizálási problémát

$$\min\{\ max\{w(u) : u \in o\} : (m, o) \in S(\mathbf{M})\}. \tag{3}$$

Ehhez legyen $O = \{u_1, \ldots, u_n\}$. Az általánosság megszorítása nélkül feltehető, hogy $w(u_1) \leq w(u_2) \leq \cdots \leq w(u_n)$. Minden $i(\leq n)$ pozitív egészre, legyen $O_i = \{u_1, \ldots, u_i\}$ és $M_i = mat(O_i)$. Továbbá, legyen $\mathbf{M}_i = (P, R, O_i)$.

6.1. Lemma. *Ha $1 \leq i \leq n$ egész, és $S(\mathbf{M}_i)$ rendelkezik nem üres maximális struktúrával, és az $S(\mathbf{M}_{i-1})$ pedig nem, akkor u_i benne van az $S(\mathbf{M}_i)$ minden lehetséges megoldásában.*

A fenti lemma segítségével megmutattuk [6], hogy (3) optimum értéke $w(u_i)$.

A következő eljárás megoldást ad (3)-ra.

Procedúra 1.

- Inicializálás

  Legyen $i = 1$ és $O_i = \{u_1\}$.

- Iteráció (i.).

  Legyen $M_i = mat(O_i)$. Alkalmazzuk a Maximális Struktúra Generáló (MSG) eljárást $\mathbf{M}_i = (P, R, O_i)$-re. Ha létezik a maximális struktúra, akkor az eljárás befejeződött; A maximális struktúra egy optimális megoldás és az optimum értéke $w(u_i)$. Ellenkező esetben, legyen $O_{i+1} = \{u_1, \ldots, u_{i+1}\}$, $i := i + 1$, és folytassuk a következő iterációval.

A Procedúra 1. hatékonysága tovább javítható, időigénye $q(n) \cdot \log(n)$ lesz, ahol $q(n)$ az MSG időkomplexitását jelöli (ismeretes, hogy az algoritmus polinomiális).

6.2. A PNS k-összeg változata

Legyen k egy rögzített pozitív egész. Olyan lehetséges megoldást szeretnénk találni, melyben a k legköltségesebb műveleti egység költségeinek összege minimális. Az általánosság megszorítása nélkül most is feltesszük, hogy $w(u_1) \leq w(u_2) \leq \cdots \leq w(u_n)$.

Jelölje $u_{i_1}, \ldots, u_{i_k}$, a k legdrágább gépet az (m, o) lehetséges megoldásban. A *k-összeg* PNS *probléma* ekkor

$$\sum_{t=1}^{k} w(u_{i_t}) : (m, o) \in S(\mathbf{M}), \tag{4}$$

A (4) megoldása érdekében, a műveleti egységek súlyai szerint vezessünk be egy lineáris rendezést az O, legfeljebb k elemű részhalmazain, jelöljük ezt $\preceq$ -vel. Legyen

$$\{u_{i_1}, \ldots, u_{i_r}\} \preceq \{u_{j_1}, \ldots, u_{j_s}\} \Leftrightarrow \sum_{t=1}^{r} w(u_{i_t}) \leq \sum_{t=1}^{s} w(u_{i_t}).$$

Azonos összeg esetén a rendezést további szabályokkal pontosítjuk.

Bármely $\{u_{i_1}, \ldots, u_{i_r}\} \subseteq O$ esetén, legyen

$$O_{\{u_{i_1}, \ldots, u_{i_r}\}} = \{u_{i_1}, \ldots, u_{i_r}\} \cup \{u_t : u_t \in O \;\&\; w(u_t) \leq w(u_{i_1})\},$$

feltesszük, hogy az u_{i_1} indexe a legkisebb az $\{u_{i_1}, \ldots, u_{i_r}\}$ halmazban. Továbbá legyen $\mathbf{M}_{\{u_{i_1}, \ldots, u_{i_r}\}} = (P, R, O_{\{u_{i_1}, \ldots, u_{i_r}\}})$. Ekkor fennáll a következő állítás.

6.2. Lemma. *Ha $\{u_{i_1}, \ldots, u_{i_r}\} \subseteq O$, és $S(\mathbf{M}_{\{u_{i_1}, \ldots, u_{i_r}\}})$ maximális struktúrája létezik, valamint minden $\{u_{j_1}, \ldots, u_{j_s}\} \subseteq O$ és $\{u_{j_1}, \ldots, u_{j_s}\} \preceq \{u_{i_1}, \ldots u_{i_r}\}$ esetén, $S(\mathbf{M}_{\{u_{j_1}, \ldots, u_{j_s}\}})$ maximális struktúrája nem létezik, akkor az $S(\mathbf{M}_{\{u_{i_1}, \ldots, u_{i_r}\}})$ maximális struktúrája egy optimális megoldása (4)-nek, és az optimum értéke pedig $\sum_{t=1}^{r} w(u_{i_t})$.*

A Lemma 6.2 alapján, a következő eljárás megadja a (4) optimális megoldását.

Procedúra 2.

- *1. lépés* Előállítjuk a $\preceq$ lineáris rendezést.
- *2. lépés* Legyen $i = 1$.
- *3. lépés* Tekintsük a rendezés szerinti i. részhalmazát az O-nak. Legyen $\{u_{i_1}, \ldots, u_{i_r}\}$ ez a részhalmaz. Futtassuk az MSG algoritmust az $\mathbf{M}_{\{u_{i_1},\ldots,u_{i_r}\}}$-re. Ha létezik a maximális struktúra, akkor az eljárás befejeződött; a kapott maximális struktúra egy optimális megoldás. Egyébként legyen $i := i + 1$ és lépjünk újra a 3. lépésre.

A Procedúra 2. időigénye $\sum_{t=1}^{k} \binom{n}{t} \cdot q(n)$ ahol $q(n)$ itt is az MSG időkomplexitását jelöli.

Érdemes megjegyezni, hogy a [6] cikkben bemutatott technika alkalmazható még (4)-nél általánosabb célfüggvényre is. Ha $z(w(u_{i_1}), \ldots, w(u_{i_k}))$ tetszőleges függvénye a legnagyobb költségeknek, akkor

$$\min\{z(w(u_{i_1}), \ldots, w(u_{i_k})) : (m, o) \in S(\mathbf{M})\}, \tag{5}$$

hasonlóan kezelhető.
Ebben az esetben a lineáris rendezés kicsit más alakot ölt:

$$\{u_{i_1}, \ldots, u_{i_r}\} \preceq \{u_{j_1}, \ldots, u_{j_s}\} \Leftrightarrow$$

$$z(w(u_{i_1}), \ldots, w(u_{i_k})) \leq z(w(u_{j_1}), \ldots, w(u_{j_k})).$$

A bottleneck PNS probléma jól-megoldható, bár a Minsum PNS probléma NP–teljes. A fenti módszerünk a k-összeg változatra polinomiális „kicsi”, rögzített k-ra.

7. Egy kehely-típusú algoritmus alkalmazása a minimális éllefedési problémára

Bizonyos esetekben ki lehet használni a tényt, hogy tudjuk a műveleti egységekről, hogy kevés anyagi ponthoz kapcsolódnak. A [9] és [10] munkánk célja az volt, hogy olyan heurisztikus algoritmusokat fejlesszünk ki, amelyek egyrészt ilyen egyszerűsített PNS feladatokra vonatkoznak, másrészt minden lépésben optimálisan döntenek.

7.1. A műveleti egységek helyettesítése egyszerűbbekkel

Egy műveleti egységet *egyszerűnek* nevezünk, ha összesen legfeljebb három anyaghoz kapcsolódik, tehát a bemeneti és kimeneti anyagok együttes száma legfeljebb 3. Tetszőleges P-gráfhoz megadunk egy vele ekvivalens hálózatot, amely csak egyszerű gépeket tartalmaz. A helyettesítéssel kapott P-gráfot *egyszerűsített* PNS *problémának* nevezzük. A konstrukció során új gépeket és anyagokat kell bevezetni, de a feladat mérete nem növekszik meg drasztikusan. Legyen $\mathbf{M} = (P, R, O)$ egy PNS probléma strukturális modellje és legyen $u = (\alpha, \beta) \in O$ egy műveleti egység, melyre $\alpha = \{X_1, \ldots, X_n\}$ és $\beta = \{Y_1, \ldots, Y_m\}$. Ha egy u nem egyszerű, akkor definiáljuk a következő műveleti egységeket $u_1, \ldots, u_{n+m}$ a következőképpen
$u_1 = (\{X_1, X_2\}, \{Z_1\})$,
$u_i = (\{X_{i+1}, Z_{i-1}\}, \{Z_i\}) \qquad i = 2, 3, ..., n-1$,
$u_n = (\{Z_{n-1}, Z_{n+m}\}, \{Z_n\})$,
$u_{n+j} = (\{Z_{n-1+j}\}, \{Z_{n+j}, Y_j\}) \qquad j = 1, 2, ..., m$,
ahol $Z_1, Z_2, \ldots, Z_{n+m}$ jelöli az új anyagokat $Z_1, Z_2, \ldots, Z_{n+m} \notin M \cup O$. Helyettesítsük most az u-t az $u_1, u_2, \ldots, u_{n+m}$ gépekkel az O-ban és legyen $O^* = O \backslash \{u\} \cup \{u_1, u_2, \ldots, u_{n+m}\}$ az új géphalmaz, és a $\{Z_1, Z_2, \ldots, Z_{n+m}\}$ halmazt uniózzuk az M-el. Végezzük el a hasonló helyettesítést minden nem egyszerű műveleti egységre egymástól függetlenül, csupa új anyagokat és gépeket definiálva. Legyen $\mathbf{M}^* = (P^*, R^*, O^*)$ a kapott új struktúra, ahol $P^* = P$ és $R^* = R$.

7.1. Tétel. *Létezik bijektív leképezés $S(\mathbf{M})$-ből $S(\mathbf{M}^*)$-ba.*

7.2. Az MCEC egy alkalmazása egyszerűsített PNS feladatokra

Kifejlesztettünk egy heurisztikus algoritmust az egyszerűsített PNS feladatok megoldására. Minden iterációban lesz egy változó kívánt anyagok halmaza; ez kezdetben a P. Az eljárás ehhez a halmazhoz fog mindig egy G gráfot hozzárendelni a következő módon: Két kívánt anyagot pontosan akkor kötünk össze egy éllel, ha létezik olyan műveleti egység, nevezzük *szülőnek*, amely két kimeneti anyaga éppen ez a két anyag. Lehetséges több szülő is. A G gráf ezen éléhez rendelt költség legyen a minimális súlyú szülő műveleti egység súlya. Az izolált csúcsokat elhagyva a kapott gráfra alkalmazzuk a minimális súlyú éllefedési algoritmust. A kapott minimális súlyú éllefedésben szereplő élekhez tartozó szülő műveleti egységek bemeneti anyagai, valamint az izolált csúcsokként le nem fedett anyagok közül mindazok a következő iteráció kívánt anyagai lesznek, amelyeket nem gyárt egyetlen már korábban kiválasztott gép sem. Az eljárás akkor ér véget, ha üres lesz a kívánt anyagok aktuális halmaza. Ekkor az egyes iterációs lépésekben kiválasztott műveleti egységek alkotják a lehetséges megoldásunkat. Mi motiválta algoritmusunkat? A PNS problémára hatékony megoldási eljárásra nincs esélyünk, mivel az NP-nehéz [2] probléma. Jól kiválasztott speciális problémaosztályokra találtunk jó megoldásokat [7]. Általában azonban be kell érnünk heurisztikus megoldó eljárássokkal. Ebben az algoritmusban azonban legalább minden egyes lépésben optimálisan próbáljuk előállítani a kívánt anyagok aktuális halmazát. Erre kitűnő eszköz az MCEC problémára vonatkozó, azonos nevű algoritmus (l. [34] és [35]). Erre a fontos, Edmonds híres párosítási algoritmusához hasonló algoritmusra kevés alkalmazást találtunk, sőt az algoritmus matematikai leírásán túl nem találtunk használható megvalósítást sem.

7.3. Heurisztikák egyszerűsített PNS problémákra

Legyen (P, R, O) egy egyszerűsített PNS probléma strukturális modellje.

R: a nyersanyagok halmaza,

P: a kívánt anyagok halmaza, természetesen

$P \cap R = \emptyset$,

$M \supseteq P \cup R$ az összes anyag halmaza.

Minden $u \in O$ műveleti egység $(1, 2)$ vagy $(2, 1)$ típusú.

Minden $m_q \in M \setminus R$ esetén létezik

$(\alpha, \beta) \in O$ amelyre $m_q \in \beta$.

A heurisztika leírása:

Legyen (P, R, O) egy, a feltételeket kielégítő PNS probléma strukturális modellje, azaz minden $u \in O$ gép $(1, 2)$ vagy $(2, 1)$ típusú.

RM_i az i-edik lépés elvégzése után gyártani kívánt anyagok,

PM_i az i-edik lépés után rendelkezésre álló anyagok,

$G_i(N_i, A_i)$ $(c(e) : e \in A_i)$

Pontosan akkor kapcsolunk össze két, $m_q, m_r \in RM_{i-1}$ csúcsot egy e éllel, ha létezik $(\alpha, \beta) \in O$ és $m_q, m_r \in \beta$ $\quad (\beta = \{m_q, m_r\})$.

$(\alpha, \beta)_{q,r}$:= azon minimális súlyú gépek egyike, melyek gyártják mind m_q-t és m_r-t, feltéve, hogy létezik ilyen gép.

$c(e) := w(\alpha, \beta)_{q,r}$ ha $e = (m_q, m_r)$

$A_i := \{e:\ e = (m_q, m_r)\ m_q, m_r \in RM_{i-1} m_q \neq m_r$
és $\exists(\alpha; m_q, m_r) \in O\}$
Ha $A_i = \emptyset$ akkor G_i nem létezik.
$N_i := \{$Az A_i-beli élek végpontjai$\}$ $(\subseteq RM_{i-1})$

Az algoritmus lépésenkénti leírása:

1./ Inicializálás: $O_0 = \emptyset,\ RM_0 = P,\ PM_0 = R$
$i = 0$

2./ Konstruáljuk meg a G_i gráfot a fent leírt módon. Ha létezik G_i, akkor $\Longrightarrow$ 3.
Ha G_i nem létezik, akkor $\Longrightarrow$ 4.

3./ Alkalmazzuk G_i-re az MCEC algorimust, legyen E_i a minimális súlyú lefedő élhalmaz.
$O_i := O_{i-1} \bigcup \{(\alpha, \beta)_{q,r} : (m_q, m_{r)} \in E_i\}$
$RM_i := (P \bigcup mat^{in} O_i) \setminus mat^{out} O_i$
$PM_i := R \bigcup mat^{out} O_i$
Ha $RM_i = \emptyset$, akkor $O^H := O_i$ a gépek heurisztika által adott megoldási halmaza. Készen vagyunk.
Ha $RM_i \neq \emptyset$ akkor $i := i + 1,\ \Longrightarrow$ 2.

4./ $u^i :=$ azon minimális súlyú gépek egyike, mely gyártja valamely RM_{i-1}-beli anyagot.
$O_i := O_{i-1} \bigcup u^i$
$RM_i = (P \bigcup mat^{in} O_i) \setminus mat^{out} O_i$
$PM_i = R \bigcup mat^{out} O_i$
Ha $RM_i = \emptyset$, akkor $O^H := O_i$ a gépek heurisztika által adott megoldási halmaza. Készen vagyunk.
Ha $RM_i \neq \emptyset$ akkor $i := i + 1, \Longrightarrow$ 2.

A [9] és [10] cikkekben további, módosított algoritmusokat is definiáltunk és elemeztünk.

8. Domináló halmazok de Bruijn gráfokban

A *perfekt d-domináló halmaz* fogalma először Biggs [1] cikkében merült fel. Biggs ott *perfekt d-kódnak* nevezte el ugyanazt a fogalmat, amit mi itt perfekt d-domináló halmaznak hívunk. Később bevezette a távolság-tranzitív gráf fogalmát és erre a gráfosztályra fontos szükséges feltételeket állapított perfekt d-kód létezésével kapcsolatban. A [31] cikkükben M. Livingston és Q. F. Stout egyéb gráfosztályokkal is foglalkozott. Az alkalmazási lehetőségek oldaláról is vizsgálták azokat a hálózatokat, amelyek összekapcsolt komputerek topológiáját adják gyakran. Létezési tételeket bizonyítottak, többek között bizonyos de Bruijn gráfokról is. Fontos további nyitott kérdéseket fogalmaztak meg végtelen de Bruijn gráf osztályokról. Bebizonyítottuk a [11] cikkben két sejtésüket. Ebben a pontban a tett megállapításainkat foglaljuk össze.

8.1. Alapfogalmak

Legyen $\mathcal{A}$ egy ábécé, $|\mathcal{A}| = n$. A de Bruijn gráfokat definiáljuk a következő módon:

$$B(n,k) = (V(n,k), E(n,k))$$

ahol $V(n,k) = \mathcal{A}^k$ a gráf csúcsainak halmaza az ábécé betűiből alkotott összes k hosszú szóból áll. Az irányított élek $E(n,k) = \mathcal{A}^{k+1}$ azonosíthatók a $k+1$ hosszú szavakkal, mert az $x_1x_2\dots x_k$ csúcsból az $y_1y_2\dots y_k$ csúcsba pontosan akkor vezet irányított él, ha $x_2x_3\dots x_k = y_1y_2\dots y_{k-1}$.

Egy $G = (V,E)$ gráfban egy y csúcsot *dominál* egy x csúcs (másképpen x *dominálja* y-t) ha létezik egy irányított él az x-től az y-ba, vagy $x = y$. A csúcsok egy $D \subseteq V$ részhalmaza *domináló halmaza* G-nek, ha a G minden csúcsát dominál legalább egy D-beli csúcs. A G legkisebb elemszámú domináló halmazának méretét hívjuk a G *dominálási számának.* A G-ben bármely ekkora domináló halmazt *minimális domináló halmaznak* nevezzünk. Ha a G mindegyik csúcsát a D pontosan egy csúcsa dominál, akkor a D-t a G *perfekt domináló halmazának (PDS)* nevezzük. Egy x csúcs *d-dominál* egy y csúcsot, ha létezik G-ben legfeljebb d hosszú irányított út x-től y-ig. A csúcsok egy $D \subseteq V$ részhalmaza *d-domináló halmaza* G-nek, ha a G minden csúcsát d-dominál legalább egy D-beli csúcs. Egy ilyen D halmaz *perfekt d-domináló halmaz (d-PDS)* ha a G mindegyik csúcsát csak egyetlen D-beli pont d-dominál.

8.2. Minimális domináló halmazok

Általában gráfokban minden fontos dominálással kapcsolatos kérdés NP–teljes. Sikerült konstruktívan előállítani minimális domináló halmazokat különböző paraméterű de Bruijn gráfokban.

8.1. Tétel. [11] *A $B(2,k)$ de Bruijn gráfban egy minimális domináló halmaz $\left\lceil \frac{2^k}{3} \right\rceil$ csúcsot tartalmaz.*

8.2. Tétel. [11] *A $B(n,k)$ de Bruijn gráfban egy minimális domináló halmaz $\left\lceil \frac{n^k}{n+1} \right\rceil$ csúcsból áll.*

M. Livingston és Q. F. Stout igazolta [31] a következő eredményt (Theorem 2.12).

8.1. Állítás. *Tetszőleges $d \geq 1$ esetén ha a k pozitív egész $(d+1)m$ vagy $(d+1)m-1$ alakú, illetve $k < d$, jelölje T_k a $B(2,k)$ csúcsainak következő részhalmazát:*
(i) $T_1 = T_2 = \ldots = T_d = \{0\}$,
(ii) $T_{(d+1)(m+1)-1} = T_{(d+1)m-1} \cup \{j : 2^{(d+1)m-1} \leq j \leq 2^{(d+1)m} - 1\}$,
(iii) $T_{(d+1)m} = T_{(d+1)m-1} \cup \{2^{(d+1))m} - 1 - s : s \in T_{(d+1)m-1}\}$.
Ekkor a T_k perfekt d-domináló halmaza $B(2,k)$-nek.

A [31] cikkben megfogalmazták a következő sejtést: *Talán nincsenek is a fent definiálttól különböző perfekt d-domináló halmazok. Speciálisan $B(2,k)$-ban talán egyáltalán nem is található 2-domináló halmaz, amikor $(k-1)$ a 3 többszöröse.* A következő tétel állítja, hogy az utóbbi sejtés igaz.

8.3. Tétel. [11] *A $B(2,k)$ de Bruijn gráfban nincs perfekt 2-domináló halmaz, ha $(k-1)$ a 3 többszöröse.*

M. Livingston és Q. F. Stout a [31] cikkben az irányítatlan esettel is foglalkozik. Ha elhagyjuk az élek irányítását a $B(n,k)$-ban, akkor megkapjuk az irányítatlan de Bruijn gráfot. Jelölje tehát $B^*(n,k)$ az irányítatlan de Bruijn gráfot. Most tehát az $x_1x_2\ldots x_k$ és $y_1y_2\ldots y_k$ csúcsok pontosan akkor vannak összekötve, ha $x_2x_3\ldots x_k = y_1y_2\ldots y_{k-1}$ vagy $x_1x_2\ldots x_{k-1} = y_2y_3\ldots y_k$ teljesül. Hasonlóan adható meg a *dominálás* (és *d-dominálás*) fogalma is. A [31] munkában olvasható a tény, hogy $B^*(2,k)$ rendelkezik perfekt domináló halmazzal (PDS) k=1 vagy 2 esetén, de nincs PDS k=3, 4 vagy 5-re. Beláttuk [11], hogy $B^*(2,k)$ csak k=1 vagy k=2-re rendelkezik PDS-sel.

8.4. Tétel. [11] *$k > 2$ esetén nincs PDS $B^*(2,k)$-ban.*

9. A HPPIT probléma

Imreh Balázs szerzőtársunk, tanárunk, szeretett kollégánk 2006 augusztus 8-án elhunyt. A [12] cikket szerzőtársaimmel az ő emlékének ajánljuk.

Több fontos kombinatorikus optimalizálási probléma keres hálózatokban optimális költségű utakat, körutakat. Ezek közül az egyik legnépszerűbb a teljes irányított hálózatokban minimális költségű Hamilton-út keresése. Ennek körút változata a híres utazó ügynök probléma (Travelling Salesman Problem TSP). Egy másik jól ismert probléma az optimális sorrend (Linear Ordering Problem LOP) megtalálásának feladata, ahol a célfüggvényünk az előremutató irányított élek súlyainak összege, ezt szeretnénk maximalizálni.

9.1. A TSP és LOP problémák általánosítása

Bemutatunk itt egy új modellt, amelynek a célfüggvénye a fenti két probléma célfüggvényeiből alakult ki. A motivációt a bevezetőben említett gyakorlati kérdéshez hasonlók jelentették. Tegyük fel, hogy egy jármű néhány helyet meglátogat, és a korábbiakból későbbiekbe szállíthat valamit. Ha a cél a haszon maximalizálása a megtett út költségét leszámítva belőle, akkor kapjuk a (Min-cost Hamiltonian path problem with internal transport, rövidítve $HPPIT$) problémát.

A HPPIT két NP-nehéz probléma közös általánosítása, tehát maga is NP-nehéz. Divatos, de egyben fontos is nehéz optimalizálási problémákra ügyes heurisztikákat definiálni, amelyek kevés számolási igénnyel kielégítő lehetséges megoldásokat adnak. Ha azonban nem vagyunk elégedettek a heurisztikus megoldás jóságával, akkor is felhasználhatjuk a kapott lehetséges megoldásokat arra, hogy velük gyorsítsunk fel pl. egy, a Branch and Bound kereteljárásra épülő optimális megoldó eljárást.

Az új HPPIT modellre a TSP vagy a LOP problémákra definiált heurisztikus algoritmusokat érdemes továbbfejleszteni. Ügyelnünk kell azonban arra, hogy a HPPIT bemenő adatai, a profit mátrix és az élekhez rendelt utazási költség mátrix összemérhető értékekből álljon, ne dominálljanak sem a várható haszon értékei, sem az élek költségei. Ilyen bemeneteknél várható, hogy az algoritmusaink láthatóan eltérő eredményt adjanak, ekkor érdemes a futási eredményeket egymással összehasonlítani.

10. Fogalmak és jelölések

Legyen G(V,A) egy irányított teljes gráf, ahol $V = v_0, v_1, \ldots, v_n$ a csúcsok halmaza, (v_0 a jármű telephelye, míg a további csúcsok jelentik azokat a helyeket, amelyeket útja során meg kell hogy látogasson). Adott továbbá két $(n+1) \times (n+1)$-es nemnegatív mátrix, B és D. B_{ij} a lehetséges hasznot jelenti, ha a v_i -ből a v_j -be történő szállítás létrejön, vagyis ha v_i az útvonalon megelőzi v_j -t ($B_{ii} = 0$ for each i). D_{ij} adja a v_i -ből v_j -be történő közvetlen eljutás költségét ($D_{ii} = 0$ for each i), tehát a hálózatban az élhez rendelt súlyt.

A HPPIT problémában a telephelyről indul a jármű, minden várost pontosan egyszer érint, majd visszatér a telephelyére. Egy lehetséges megoldás tehát azonosítható az $\{1,..,n\}$ halmaz egy p permutációjával. A p által meghatározott körútban a jármű a v_0 csúcsból indul és rendre $v_{p(1)}, v_{p(2)}, .., v_{p(n)}$ sorrendben látogatja meg a városokat. A célfüggvényt a következő formula adja

$$z(p) = \sum_{0<i<j<n+1} B_{p(i),p(j)} + \sum_{i=1}^{n} (B_{0,p(i)} + B_{p(i),0}) - \sum_{0 \le i < n} D_{p(i),p(i+1)} - D_{p(n),0},$$

és ezt szeretnénk maximalizálni. Természetesen a megoldásról úgy is beszélhetünk mint az összes csúcson áthaladó körről.

11. Heurisztikus algoritmusok a HPPIT problémára

A [12] cikkben bemutatunk 6 alapalgoritmust. Ezek különböző szabályok szerint építenek fel egy körutat. Ezután a kapott körutakat a szomszédsági keresés technikájával próbáljuk meg javítani. Itt is felidézünk egy alapalgoritmust.

11.1. Körútépítő technikák

Még mindig mohó módon, de most már mindkét irányba építsük a kört! Felváltva lépjünk, egyet hátulról, egyet elölről! Most is válasszunk azon lehetséges csúcsok közül, amelyek maximális haszonnal kecsegtetnek pillanatnyilag, az utazási részköltséget is számítva.

3. algoritmus

1. lépés (Az utolsó $v_{p(n)}$ és az első $v_{p(1)}$ csúcs meghatározása): Legyen $0 < k < n+1$ az az érték, melyre $\sum_{0<j<n+1} B_{jk} - D_{k0} = \max_{0<i<n+1} \sum_{0<j<n+1} B_{ji} - D_{i0}$. Ha egynél több ilyen k van, akkor válasszuk a legnagyobbat! Legyen $p(n) = k$, $F = \{k\}$ (F a már eddig rendezett csúcsok halmaza). Ha $n > 1$, akkor legyen $0 < k < n+1$, $k \notin F$ az az érték, melyre $\sum_{0<j<n+1, j \neq p(n))} B_{kj} - D_{0k} = \max_{0<i<n+1, i \notin F} \sum_{0<j<n+1, j \neq p(n)} B_{ij} - D_{0i}$. Legyen $p(1) = k$. Ha egynél több megfelelő k van, akkor válasszuk a legnagyobbat. Legyen $t = n-1$, $z = 2$, és $F = F \cup \{k\}$. Lépjünk a 2. lépésre!

2. lépés Ebben a lépésben a hátulról következő várost $p(t)$-t választjuk ki. Ha $z = n$, akkor az eljárás befejeződik, a p permutációt megadtuk, $v_0, v_{p(1)}, \ldots, v_{p(n)}, v_0$ a körút. Ha $z < n$, akkor legyen $p(t)$ a maximális k, $k \notin F$ melyre: $-D_{k,p(t+1)} + \sum_{0<j<n+1, j \notin F} B_{jk} =$
$\max_{0<i<n+1, i \notin F} \{-D_{i,p(t+1)} + \sum_{0<j<n+1, j \notin F} B_{ji}\}$.

Legyen $z = z+1$, $t = n-t+1$, $F = F \cup \{k\}$ és lépjünk a 3. lépésre.

3. lépés Ebben a lépésben meghatározzuk $p(t)$-t, az elölről következő indexet. Ha $z = n$, akkor az eljárás véget ér, p-t meghatároztuk, a körút a $v_0, v_{p(1)}, \ldots, v_{p(n)}, v_0$. Ha $z < n$, akkor legyen $p(t)$ a legkisebb k, $k \notin F$, melyre: $-D_{p(t-1),k} + \sum_{0<j<n+1, j \notin F} B_{kj} = \max_{0<i<n+1, i \notin F} \{-D_{p(t-1),i} + \sum_{0<j<n+1, j \notin F} B_{ij}\}$

Legyen $z = z+1$, $t = n-t$, $F = F \cup \{k\}$ és lépjünk a 2. lépésre.

12. Összefoglalás

A hálózati folyamatok szintézise témában adott egy anyaghalmaz, amelyet megadott átalakítók egy része segítségével elő szeretnénk állítani. A nyersanyagokon kívül tehát a gépek összes bemeneti anyagát, és a kívánt anyagokat is dominálnia kell valamely kiválasztott gépnek. Egy géphalmaz, amely teljesíti az eddig említett feltételeket csak akkor lehetséges megoldás, ha nincs közöttük felesleges, vagyis olyan, amely nem d-dominál legalább egy kívánt anyagot valamilyen d-re. A domináló csúcshalmazok szerepét hálózatokban csak a 8. fejezetben mutatjuk be az eredeti értelmében. A disszertáció harmadik témája a HPPIT probléma. A LOP probléma az egyik olyan, amelyből a HPPIT ered. Ebben, lényegét tekintve az a feladatunk, hogy megtaláljuk a csúcsoknak azt a sorrendjét, amelyre a legtöbb az előre mutató él. A második fejezetben összefoglaljuk a PNS alapvető fogalmait és felidézünk több, korábbi eredményt. A strukturális modell, a maximális struktúra definíciója, a lehetséges megoldások legfontosabb kombinatorikus tulajdonságainak leírása megtalálható a [14], [15], [16], [17],[18], [19], [21], [24], [25] cikkekben. Az első vizsgált optimalizálási feladat az volt, amikor a műveleti egységekhez rendeltünk költséget, és minimális összköltségű lehetséges megoldást kerestünk. Erre a PNS problémára először a Korlátozás és Szétválasztás technikájával sikerült megoldást adni.

Vajon meg lehet-e oldani hatékonyan egy PNS problémát? A kombinatorikus összefüggések megértésével tudunk-e polinomiális algoritmust definiálni a problémára? A 3. fejezetben egy polinomidejű visszavezetést adunk meg (l. a [2] cikket); a PNS egy speciális osztályára vezetjük vissza az NP–teljes halmazlefedési feladatot (3.1 tétel). A Korlátozás és Szétválasztás elvére épülő algoritmusokat mutat be a [15] cikk, amelyben bevezették a döntési leképezés fogalmát. A negyedik fejezetben tárgyaljuk az ún. konzisztens döntési leképezések számának korlátozására vonatkozó eredményeinket [3], [4], [5]. A Korlátozás és Szétválasztás elvű algoritmusok gyorsításában nagy jelentőségű a lehetséges maximális konzisztens döntési leképezések számának felülről történő jó becslése. Ha a lehetséges megoldások axiómái közül az egyiket vesszük csak figyelembe, akkor felső korlátot kaphatunk (4). Ha a Szitaformulát szeretnénk felhasználni a korlát kiszámításához, akkor $|A_I|$ meghatározása szükséges. Ez általában nem tűnik egyszerűnek. Használható a formula szeparátor típusú műveleti egységekre, az ún. Egyenes és Lánc modelleknél sikerült pontosan kiszámolni a korlátokat, amelyek elég jóknak is bizonyultak. Az eltérő számítási módszerek hozománya két kombinatorikus azonosság is.

A célfüggvény szerint csoportosítva kombinatorikus optimalizálási feladatokat, beszélhetünk Minsum, Bottleneck vagy k-összeg változatokról. Az utóbbi kettőt először a [6] cikkben vizsgáltam PNS feladatokra. Kiderült (6.1 Lemma), hogy a Bottleneck feladatra adható hatékony algoritmus (6. fejezet, 1. Procedúra). A PNS k-összeg változatára bizonyítottam a 6.2. Lemmát, az erre épülő eljárás (2. Procedúra) rögzített k-ra hatékony.

Minden P-gráfot át lehet alakítani, egy vele ekvivalens másik, egyszerűbb P-gráfra, amelynek lehetséges megoldásai alapján az eredeti feladat megoldásait visszakaphatjuk (7.1 Tétel). Az egyszerűsített P-gráfok csak három anyaghoz kapcsolódó gépeket tartalmaznak. Ilyen P-gráfokra sikeresen alkalmaztuk több heurisztikában az MCEC algoritmust, amely egy hálózatban minimális összsúlyú élhalmazzal fedi le a gráf csúcsait. A 7. fejezet heurisztikái minden lépésben a valódi optimumot adó MCEC algoritmust használják. Emiatt az egyes lépésekben nemcsak az időigényt tekintve működnek hatékonyan az algoritmusok. A [9] és [10] cikkekben publikáltam a módszert és elemeztük a számítógépes tapasztalatainkat nagy, generált feladatokon.

M. Livingston és Q. F. Stout a [31] cikkükben számos fontos gráfosztályra vizsgáltak dominálási kérdéseket. A de Bruijn gráfokra kimondott tételük konstrukciót adott bizonyos paraméterosztályok esetén, de nyitva hagyott további sejtéseket. A 8. fejezetben két tételben (8.2 és 8.4) igazoltuk, hogy nem létezhetnek perfekt domináló halmazok végtelen sok de Bruijn gráfban [11]. Minimális domináló halmazra általános konstrukciót adtunk.

A 9. fejezetben bevezettünk egy új, a valós élet által motivált kombinatorikus optimalizálási problémát, amely két híres NP–teljes probléma közös általánosítása [12]. Összesen 14 algoritmus változatot hasonlítottunk egymáshoz. Úgy gondolom egy körút során a belső szállítások által elérhető teljes haszon maximalizálása számos gyakorlati problémában megjelenik, emiatt a HPPIT problémát mind elméleti, mind gyakorlati szempontból sokan fogják még vizsgálni.

Thesis 1

3.1 Tétel. [2] A PNS_{w1} PNS osztály ekvivalens a halmazlefedési problémával.

3.2 Következmény. [2] A (2) PNS probléma NP–nehéz.

Thesis 2

4.2 Tétel. [3] Minden $\emptyset \neq m \subseteq M \setminus R$ esetén, $\tau(m) = 2^{|\cup\{\Delta(X):X\in m\}|}$.

4.3 Tétel. [3]
$\tau'(m) = |\Omega_{\mathbf{M}}^{\max} \setminus (A_1 \cup A_2 \cup ... \cup A_k)| = \Sigma_{I\subseteq\{1,...k\}}(-1)^{|I|} \cdot |A_I|$.

5.1 Tétel. [3] Szeparátor típusú műveleti egységek esetén

$$|A_I| = \left(\prod_{t=1}^{l}\left(2^{\left|O^*\left(X_{i_t}\right)\right|} - 1\right)\right) \cdot 2^{|O\setminus(\cup_{i\in I}\Delta(X_i))\setminus(\cup_{i\in I}O(X_i))|}.$$

5.2 Tétel. [4] Az Egyenes modellben $|S(\mathbf{M})| \leq L^{(1)}$.

5.3 Tétel. [4] A Lánc modellben $|S(\mathbf{M})| \leq C^{(1)}$.

Thesis 3

6.1 Lemma [6] Ha $1 \leq i \leq n$ egész, és $S(\mathbf{M}_i)$ rendelkezik nem üres maximális struktúrával, és az $S(\mathbf{M}_{i-1})$ pedig nem, akkor u_i benne van az $S(\mathbf{M}_i)$ minden lehetséges megoldásában.

1. Procedúra. [6]

6.2 Lemma [6] Ha $\{u_{i_1}, \ldots, u_{i_r}\} \subseteq O$, és $S(\mathbf{M}_{\{u_{i_1},\ldots,u_{i_r}\}})$ maximális struktúrája létezik, valamint minden $\{u_{j_1}, \ldots, u_{j_s}\} \subseteq O$ és $\{u_{j_1}, \ldots, u_{j_s}\} \preceq \{u_{i_1}, \ldots u_{i_r}\}$ esetén, $S(\mathbf{M}_{\{u_{j_1},\ldots,u_{j_s}\}})$ maximális struktúrája nem létezik, akkor az $S(\mathbf{M}_{\{u_{i_1},\ldots,u_{i_r}\}})$ maximális struktúrája egy optimális megoldása (4)-nek, és az optimum értéke pedig $\sum_{t=1}^{r} w(u_{i_t})$.

2. Procedúra [6]

Thesis 4

7.1 Tétel. [9] Létezik bijektív leképezés $S(\mathbf{M})$-ből $S(\mathbf{M}^*)$-be.

Algoritmusok 1–4.

Thesis 5

8.1 Tétel. [11] A $B(2,k)$ de Bruijn gráfban egy minimális domináló halmaz $\left\lceil \frac{2^k}{3} \right\rceil$ csúcsot tartalmaz.

8.2 Tétel. [11] A $B(n,k)$ de Bruijn gráfban egy minimális domináló halmaz $\left\lceil \frac{n^k}{n+1} \right\rceil$ csúcsból áll.

8.4 Tétel. [11] A $B(2,k)$ de Bruijn gráfban nincs perfekt 2-domináló halmaz, ha $k-1$ a 3 többszöröse.

8.5 Tétel. [11] $k > 2$ esetén nincs PDS $B^*(2,k)$-ban.

Thesis 6

A HPPIT probléma definíciója. [12]

Algoritmusok 1-4.

Az eredmények publikálása

Blázsik, Z., B. Imreh, A note on connection between PNS and set covering problems, *Acta Cybernetica*, **12**, 1996, 309-312. (MR1428741)

Blázsik, Z., Cs. Holló, B. Imreh, On Decision-Mappings Related to Process Network Synthesis Problem, *Acta Cybernetica*, **13**, 1998, 319-328. (MR1644388)

Blázsik, Z., Cs. Holló, B. Imreh, Explicit bound for the number of feasible solutions of special PNS-problem classes, *Pure Mathematics and Applications*, **9**, 1998, 17-27. (MR1677229)

Blázsik, Z., Cs. Holló, B. Imreh, Cs. Imreh, Z. Kovács, On Bottleneck and k-sum version of the Process Network Synthesis Problem, *Novi Sad Journal of Mathematics*, **3**, 2000, 11-19. (MR1776440)

Blázsik, Z., K. Keserű, Z. Kovács, Heuristics for simplified Process Network Synthesis problems with a Blossom-type Algorithm for the edge covering problem, Optimization Theory: Recent Developments from Matrahaza,(eds.: F. Gianessi, P. Pardalos, T. Rapcsák), Kluwer Academic Publishers, Dordrecht, 2001, 19-31. (MR1886425)

Blázsik, Z., K. Keserű, Z. Kovács, Heuristics for PNS problems and its empirical analysis, *Pure Mathematics and Applications*, **11**, 2001, 139-151. (MR1839923)

Blázsik, Z., Z. Kása, Dominating sets in de Bruijn graphs. Algebraic systems (Felix-Oradea, 2001). *Pure Mathematics and Applications*, **13**, 2002, 79-85. (MR1987200)

Blázsik Z., T. Bartók, B. Imreh, Cs. Imreh, Z. Kovács, Heuristics on a Common Generalization of TSP and LOP, közlésre elfogadva.

A disszertációval kapcsolatos valamennyi anyag on-line elérhető a

`http://www.inf.u-szeged.hu/~blazsik`

internetes címről.

Hivatkozások

[1] N. Biggs, Perfect codes in graphs, *J. Comb. Theory* (B), **15**, 1973, 289-296.

[2] Blázsik, Z., B. Imreh, A note on connection between PNS and set covering problems, *Acta Cybernetica*, **12**, 1996, 309-312.

[3] Blázsik, Z., Cs. Holló, B. Imreh, On Decision-Mappings Related to Process Network Synthesis Problem, *Acta Cybernetica*, **13**, 1998, 319-328.

[4] Blázsik, Z., Cs. Holló, B. Imreh, Explicit bound for the number of feasible solutions of special PNS-problem classes, *Pure Mathematics and Applications*, **9**, 1998, 17-27.

[5] Blázsik, Z., Cs. Holló, B. Imreh, Kiszámolható korlátok speciális PNS-problémaosztályok lehetséges megoldásai számára, *Új utak a magyar operációkutatásban, szerk. Komlósi, S., Szántai T., Dialóg Campus Kiadó, Budapest-Pécs,* 1999, 182-194.

[6] Blázsik, Z., Cs. Holló, B. Imreh, Cs. Imreh, Z. Kovács, On Bottleneck and k-sum version of the Process Network Synthesis Problem, *Novi Sad Journal of Mathematics*, **3**, 2000, 11-19.

[7] Blázsik, Z., Cs. Holló, B. Imreh, Cs. Imreh, Z. Kovács, On a well-solvable class of the PNS problem, *Novi Sad Journal of Mathematics*, **3**, 2000, 21-30.

[8] Blázsik, Z., Cs. Holló, Cs. Imreh, Z. Kovács, Heuristics for the Process Network Synthesis Problem, *New Trends in Equilibrium Systems, Mátraháza Optimization Days*, Kluwer Academic Publishers, 2000, 1-16.

[9] Blázsik, Z., K. Keserű, and Z. Kovács, Heuristics for simplified Process Network Synthesis problems with a Blossom-type Algorithm for the edge covering problem, Optimization Theory: Recent Developments from Matrahaza,(eds.: F. Gianessi, P. Pardalos, T. Rapcsák), Kluwer Academic Publishers, Dordrecht, 2001, 19-31.

[10] Blázsik, Z., K.. Keserű, and Z. Kovács, Heuristics for PNS problems and its empirical analysis, *Pure Mathematics and Applications*, **11**, 2001, 139-151.

[11] Blázsik, Z., Z. Kása, Dominating sets in de Bruijn graphs. Algebraic systems (Felix-Oradea, 2001). *Pure Mathematics and Applications*, **13**, 2002, 79-85.

[12] Blázsik Z., T. Bartók, B. Imreh, Cs. Imreh, Z. Kovács, Heuristics on a Common Generalization of TSP and LOP, közlésre elfogadva.

[13] Floudas, C. A., I. E. Grossmann, Algorithmic Approaches to Process Synthesis: Logic and Global Optimization, AIChE Symposium Series No. 304, **91** (Eds: L. T. Biegler and M. F. Doherly), 1995, 198-221.

[14] Friedler, F., K. Tarján, Y. W. Huang, and L. T. Fan, Graph-Theoretic Approach to Process Synthesis: Polynomial Algorithm for maximal structure generation, *Computer chem. Engng.* **17**, 1993, 924-942.

[15] Friedler, F., J. B. Varga, and L. T. Fan, Decision-Mappings: A Tool for Consistent and Complete Decisions in Process Synthesis, *Chem. Eng. Sci.*, **50** (11), 1995, 1755-1768.

[16] Friedler, F., J.B. Varga, E. Fehér, and L.T. Fan, Combinatorially Accerelated Branch-and-Bound Method for Solving the MIP Model of Process Network Synthesis, International Conference on State of the Art in Global Optimization: Computational Methods and Applications, Princeton, 1995.

[17] Friedler, F., J. B. Varga, E. Fehér, L. T. Fan, Combinatorially Accelerated Branch-and -Bound Method for Solving the MIP Model of Process Network Synthesis, *Nonconvex Optimization and its Applications*, (eds.: C. A. Floudas and P. M. Pardalos), Kluwer Academic Publishers, Norwell, MA, U.S.A., 1996, 609-626.

[18] Friedler, F., K. Tarján, Y. W. Huang, L. T. Fan, Graph-Theoretic Approach to Process Synthesis: Axioms and Theorems, *Chem. Eng. Sci.*, **47** (8), 1992, 1973-1988.

[19] Friedler, F., K. Tarján, Y. W. Huang, L. T. Fan, Combinatorial Algorithms for Process Synthesis, *Computer chem. Engng.*, **16**, 1992, 313-320.

[20] Friedler, F., J. Fulop, B. Imreh, On the reformulation of some classes of PNS-problems as set covering problems, *Acta Cybernetica*, **13**, 1998, 329-337.

[21] Friedler, F., L. T. Fan, B. Imreh, Process Network Synthesis: Problem Definition, *Networks*, **28**, 1998, 119-124.

[22] Grossmann, I. E., V. T. Voudouris, O. Ghattas, Mixed-Integer Linear Programming Reformulations for Some Nonlinear Discrete Design Optimization Problems, In: Recent Advances in Global Optimization (Eds: C. A. Floudas and P. M. Pardalos) Princeton University Press, New Jersey, 1992.

[23] Holló, Cs., Z. Blázsik, Cs. Imreh, Z. Kovács, On a Merging Reduction of the Process Network Synthesis Problem, *Acta Cybernetica*, **14**, 1999, 251-261.

[24] Imreh, B., F. Friedler, L. T. Fan, An Algorithm for Improving the Bounding Procedure in Solving Process Network Synthesis by a Branch-and-Bound Method, *Developments in Global Optimization*, ed. I. M. Bomze, T. Csendes, R. Horst, P. M. Pardalos, Kluwer Academic Publisher, Dordrecht, Boston, London, 1996, 301-348.

[25] Imreh, B., G. Magyar, Empirical Analysis of Some Procedures for Solving Process Network Synthesis Problem, *Journal of Computing and Information Technology*, **6**, 1998, 373-382.

[26] Imreh, B., *Kombinatorikus optimalizálás*, Novadat, Győr, 2000.

[27] Imreh, B., J. Fülöp, F. Friedler, A note on the Equivalence of the Process Network Synthesis and Set Covering problems, *Acta Cybernetica*, **14**, 2000, 497-502.

[28] Imreh, Cs., Jól megoldható PNS osztályokról, *Új utak a magyar operációkutatásban, szerk. Komlósi, S., Szántai T., Dialóg Campus Kiadó, Budapest-Pécs*, 1999, 168-181.

[29] Imreh, Cs., A new well-solvable class of PNS problems, *Computing*, **66**, 2001, 289-296.

[30] Karp R. M., Reducibility among Combinatorial Problems in Complexity of Computer Computations, *R. E. Miller and T. W. Thatcher, eds.*, Plenum Press, New York, 1972.

[31] Livingston, M., Q. F. Stout, Perfect dominating sets, *Congr. Numer.*, **78**, 1990, 187-203.

[32] Lothaire, M., *Combinatorics on words*, Addison-Wesley, Reading, 1983.

[33] de Luca, A., On the combinatorics of finite words, *Theor. Comput. Sci.*, **218**, 1999, 13-39.

[34] Murty, Katta G., Clovis Perin, A 1-Matching Blossom-Type Algorithm for Edge Covering Problems, *Networks*, **12**, 1982, 379-391.

[35] Murty, Katta G., Network Programming, *Prentice Hall*, 1992.

Társszerzői nyilatkozat

Kijelentem, hogy ismerem *Blázsik Zoltán* PhD fokozatra pályázó *Domináló csúcsok szerepe hálózati folyamatok tervezésében* című disszertációját.
A disszertációban szereplő közös eredményekre vonatkozóan kijelentem, hogy nincsenek olyan eredmények, melyekhez való hozzájárulásunk oszthatatlan lenne.
A következő eredményekben a pályázó hozzájárulása volt a meghatározó:

- Blázsik Z., Cs. Holló, B. Imreh, Cs. Imreh, Z. Kovács, On Bottleneck and k-sum version of the Process Network Synthesis Problem, Novi Sad Journal of Mathematics, **30**, 2000, 11-20.
- Blázsik Z., K. Keserű, Z. Kovács, Heuristics for simplified Process Network Synthesis problems with a Blossom-type Algorithm for the edge covering problem, Optimization Theory: Recent Developments from Matrahaza, (eds.: F. Gianessi, P. Pardalos, T. Rapcsák), Kluwer Academic Publishers, Dordecht, 2001, 19-31.
- Blázsik Z., T. Bartók, B. Imreh, Cs. Imreh, Z. Kovács, Heuristic on a common generalization of TSP and LOP, közlésre elfogadva.

Nincsenek olyan disszertációban szereplő közös eredmények, melyekben az én hozzájárulásom lett volna meghatározó.

Végül kijelentem,hogy a pályázó doktori értekezésében szereplő eredményeket semmilyen fokozatszerző eljárásban nem használtam fel és ez a jövőben sincs szándékomban.

Szeged, 2008. január 11. Dr. Kovács Zoltán

Társszerzői nyilatkozat

Kijelentem, hogy ismerem *Blázsik Zoltán* PhD fokozatra pályázó *Domináló csúcsok szerepe hálózati folyamatok tervezésében* című disszertációját.
A disszertációban szereplő közös eredményekre vonatkozóan kijelentem, hogy nincsenek olyan eredmények, melyekhez való hozzájárulásunk oszthatatlan lenne.
A következő eredményekben a pályázó hozzájárulása volt a meghatározó:

- Blázsik Z., Cs. Holló, B. Imreh, Cs. Imreh, Z. Kovács, On Bottleneck and k-sum version of the Process Network Synthesis Problem, Novi Sad Journal of Mathematics, **30**, 2000, 11-20.
- Blázsik Z., T. Bartók, B. Imreh, Cs. Imreh, Z. Kovács, Heuristic on a common generalization of TSP and LOP, közlésre elfogadva.

Nincsenek olyan disszertációban szereplő közös eredmények, melyekben az én hozzájárulásom lett volna meghatározó.

Végül kijelentem, hogy a pályázó doktori értekezésében szereplő eredményeket semmilyen fokozatszerző eljárásban nem használtam fel és ez a jövőben sincs szándékomban.

Szeged, 2008. január 11. Dr. Imreh Csanád

Társszerzői nyilatkozat

Kijelentem, hogy ismerem *Blázsik Zoltán* PhD fokozatra pályázó *Domináló csúcsok szerepe hálózati folyamatok tervezésében* című disszertációját.
A disszertációban szereplő közös eredményekre vonatkozóan kijelentem, hogy a következő eredményekhez való hozzájárulásunk oszthatatlan:

- Blázsik, Z., Cs. Holló, B. Imreh, On Decision-Mappings Related to Process Network Synthesis Problem, *Acta Cybernetica*, **13**, 1998, 319-328.

- Blázsik, Z., Cs. Holló, B. Imreh, Explicit bound for the number of feasible solutions of special PNS-problem classes, *Pure Mathematics and Applications*, **9**, 1998, 17-27.

- Blázsik, Z., Cs. Holló, B. Imreh, Kiszámolható korlátok speciális PNS-problémaosztályok lehetséges megoldásai számára, *Új utak a magyar operációkutatásban, szerk. Komlósi, S., Szántai T., Dialóg Campus Kiadó, Budapest-Pécs,* 1999, 182-194.

A következő eredményekben a pályázó hozzájárulása volt a meghatározó:

- Blázsik Z., Cs. Holló, B. Imreh, Cs. Imreh, Z. Kovács, On Bottleneck and k-sum version of the Process Network Synthesis Problem, Novi Sad Journal of Mathematics, **30**, 2000, 11-20.

Nincsenek olyan disszertációban szereplő közös eredmények, melyekben az én hozzájárulásom lett volna meghatározó.

Végül kijelentem, hogy a pályázó doktori értekezésében szereplő eredmények közül az oszthatatlan eredményeket Ph.D disszertációmban felhasználtam (*4.2 Tétel.*,*4.3 Tétel.*, *5.1 Tétel.*, *5.2 Tétel.*, *5.3 Tétel.*). Egyéb eredményeket semmilyen fokozatszerző eljárásban nem használtam fel és ez a jövőben sincs szándékomban.

Szeged, 2008. január 11. Dr. Holló Csaba

Társszerzői nyilatkozat

Kijelentem, hogy ismerem *Blázsik Zoltán* PhD fokozatra pályázó *Domináló csúcsok szerepe hálózati folyamatok tervezésében* című disszertációját.
A disszertációban szereplő közös eredményekre vonatkozóan kijelentem, hogy nincsenek olyan eredmények, melyekhez való hozzájárulásunk oszthatatlan lenne.
A következő eredményekben a pályázó hozzájárulása volt a meghatározó:

- Blázsik Z., K. Keserű, Z. Kovács, Heuristics for simplified Process Network Synthesis problems with a Blossom-type Algorithm for the edge covering problem, Optimization Theory: Recent Developments from Matrahaza, (eds.: F. Gianessi, P. Pardalos, T. Rapcsák), Kluwer Academic Publishers, Dordecht, 2001, 19-31.

Nincsenek olyan disszertációban szereplő közös eredmények, melyekben az én hozzájárulásom lett volna meghatározó.

Végül kijelentem, hogy a pályázó doktori értekezésében szereplő eredményeket semmilyen fokozatszerző eljárásban nem használtam fel és ez a jövőben sincs szándékomban.

Szeged, 2008. január 11. Keserű Kornél

Társszerzői nyilatkozat

Kijelentem, hogy ismerem *Blázsik Zoltán* PhD fokozatra pályázó *Domináló csúcsok szerepe hálózati folyamatok tervezésében* című disszertációját.
A disszertációban szereplő közös eredményekre vonatkozóan kijelentem, hogy nincsenek olyan eredmények, melyekhez való hozzájárulásunk oszthatatlan lenne.
A következő eredményekben a pályázó hozzájárulása volt a meghatározó:

- Blázsik Z., T. Bartók, B. Imreh, Cs. Imreh, Z. Kovács, Heuristic on a common generalization of TSP and LOP, közlésre elfogadva.

Nincsenek olyan disszertációban szereplő közös eredmények, melyekben az én hozzájárulásom lett volna meghatározó.

Végül kijelentem, hogy a pályázó doktori értekezésében szereplő eredményeket semmilyen fokozatszerző eljárásban nem használtam fel és ez a jövőben sincs szándékomban.

Szeged, 2008. január 11. Bartók Tamás

Társszerzői nyilatkozat

Kijelentem, hogy ismerem *Blázsik Zoltán* PhD fokozatra pályázó *Domináló csúcsok szerepe hálózati folyamatok tervezésében* című disszertációját.
A disszertációban szereplő közös eredményekre vonatkozóan kijelentem, hogy nincsenek olyan eredmények, melyekhez való hozzájárulásunk oszthatatlan lenne.
A következő eredményekben a pályázó hozzájárulása volt a meghatározó:

- Blázsik, Z., Z. Kása, Dominating sets in de Bruijn graphs. Algebraic systems (Felix-Oradea, 2001). *Pure Mathematics and Applications*, **13**, 2002, 79-85.

Nincsenek olyan disszertációban szereplő közös eredmények, melyekben az én hozzájárulásom lett volna meghatározó.

Végül kijelentem, hogy a pályázó doktori értekezésében szereplő eredményeket semmilyen fokozatszerző eljárásban nem használtam fel és ez a jövőben sincs szándékomban.

Szeged, 2008. január 11. Dr. Kása Zoltán