

BEFOLYÁS TERJEDÉS OPTIMUMAINAK HÁLÓZATÁRÓL

HOMOLYA VIKTOR, VINKÓ TAMÁS

A gráfokon értelmezett befolyás terjedés maximalizálás feladatát vizsgáljuk egy újfajta megközelítésben. Bevezetünk egy hegymászó-jellegű algoritmust, amelynek futtatásával értelmezhetjük a szóban forgó diszkrét optimalizálási feladat lokális optimumait, valamint az azok közötti kapcsolatokat. Az így adódó módszer néhány lehetséges alkalmazására, a keresési tér értékészletének felderítésére adunk példát különböző véletlen gráfokon végzett futtatási eredményekből.

1. Bevezetés

Legyen adott egy irányított, súlyozott hálózat, továbbá egy terjedési modell és egy k pozitív egész szám. Feladatunk, hogy kiválasszuk azt a k darab csúcsot, amelyekből a terjedési modellt indítva és végrehajtva a lehető legtöbb csúcsot elérjük. Az így kapott probléma, a befolyás terjedés maximalizálás, gyakran használt eszköz például a vírusmarketingben. Gazdasági szempontból egy példa lehet a következő: korlátozott mennyiségben van reklámozásra költségünk, maximalizálni szeretnénk az információ terjedést a termékünkéről, tehát a lehető legtöbb emberhez szeretnénk eljuttatni. A gráf reprezentálja a közösséget, melyben terjesztenénk az információt. Mint szociális hálózat, nem tartalmaz többszörös éleket és hurkokat, valamint minden egyed különböző mértékben tud hatni másokra.

A befolyás terjedés hálózatokon, mint diszkrét optimalizálási feladat először Kempe és szerzőtársai [10] cikkében jelent meg, amely Domingos és Richardson korai munkáján [6] alapszik. Ebben a mostanra klasszikussá vált cikkben adták meg a szerzők a dolgozatunkban is használt független kaszkád (independent cascade, IC) valamint a lineáris küszöb (linear threshold, LT) modelleket. Továbbá, ebben a cikkben javasolták a mohó algoritmust, amely a feladat matematikai tulajdonságából adódóan legalább $1 - 1/e$ közelítést ad az optimális értékre. Megmutatták, hogy a probléma az alap terjedési modellekben (IC és LT), melyeket figyelembe vettek, NP-nehéz.

Amennyiben a megadott modellek naiv implementációjával kísérletezünk, hamar rájöhethetünk, hogy már akár százas nagyságrendű csúcs-számmal rendelkező gráfra is nagyon sokáig tart lefuttatni a mohó algoritmust (vagy szinte bármilyen

más optimalizáló eljárást). Ennek egyik oka, hogy a terjedési modellek sztochasztikus alaptulajdonságát Monte Carlo szimulációval tudjuk kiátlagolni (azaz empirikus várható érték közelítését számolni). Ezért hamar megjelentek, és azóta is az ide vonatkozó kutatások egyik fókuszába tartoznak azok a cikkek, amelyek a feladat gyorsabb megoldását tűzték ki célul. Leskovec és szerzőtársai [12] az LT modellre adnak egy ún. *lazy-forward* mohó algoritmust CELF néven. Ennek egy továbbgondolt változata a CELF++, amely az eredeti mohó eljárás iterációinak számát csökkenti nagyságrendekkel [7]. További gyorsítási módszereket is találunk a szakirodalomban, például olyat, amely közelítő eljárást javasol [11], vagy pedig valamilyen strukturális tulajdonságot, például közösségek jelenlétével hozza összefüggésbe a leghatékonyabban aktivizáló csúcsok kiválasztását [14].

Az alapfeladat helyett, ahol tehát egy megoldást akkor tekintünk jobbnak, ha minél több befolyás alá vont csúcsot eredményez, másképpen is megfogalmazhatunk ide kapcsolódó optimalizálási problémákat, vagy változtathatunk az alapgráf tulajdonságain. Ilyen lehet például az, amikor nem csak pozitív, hanem negatív súlyokat is megengedünk. Különösen érdekes a dinamikusan változó gráfok esete is [1]. Ebben a dolgozatban, csak az eredeti alapfeladattal foglalkozunk.

Célunk, hogy a befolyás terjedés maximalizálás, mint kombinatorikus optimalizálási feladat szerkezeti tulajdonságait mélyebben megértsük. Ehhez felhasználjuk a lokális optimumok hálózatának (local optima network, LON) koncepcióját. A módszerrel tulajdonképpen az eredeti hálózathoz, mint a keresési tér értelmezési tartományából, egy speciális, hegymászó-jellegű optimalizáló eljárás felhasználásával készítünk egy másik hálózatot, a LON-t, amelynek segítségével a keresési tér értékkészletének különböző strukturális tulajdonságait vizsgálhatjuk.

2. Definíciók és jelölések

Ebben a szakaszban leírjuk a későbbiekben használt legfontosabb definíciókat, jelöléseket. A feladatban használt súlyozott, irányított gráfot $G(V, E, W)$ -vel jelöljük, ahol V a csúcsok halmaza, E az élek halmaza és $W : E \rightarrow [0, 1]$ egy súlyfüggvény. A csúcsoknak azt a k elemű részhalmazát, amelyet a feladat szerint meg kell keresnünk, az angol nyelvű szakirodalmat követve, *seed halmaznak* nevezzük. A gráfnak azon csúcsait, amelyeket a befolyás terjedési modell elér, *aktív* vagy *fertőzött* csúcsoknak nevezzük. A továbbiakban először bemutatjuk a cikkünkben használt terjedési modellt, röviden leírjuk a mohó algoritmust [10], majd bevezetjük a LON építésre használt hegymászó-jellegű algoritmust.

2.1. Befolyás terjedési modell

Mint említettük, Kempe és szerzőtársai [10] két terjedési modellt vezettek be és vizsgáltak, amelyekből jelen cikkben csak az egyiket használjuk: a *független kaszkád* (independent cascade, IC) modellt. Ebben az iteráció alapú modellben a

gráf minden élének van egy paramétere p ($0 \leq p \leq 1$), amely az élen való terjedés valószínűségét jelzi. Egy iterációban a frissen fertőzött csúcsok megpróbálják aktivizálni a (ki)szomszédait az él paramétere szerint. Amennyiben egy v csúcs nem tudta fertőzni w szomszédját először, akkor nem fogja tudni később sem, csak egyszer próbálkozhat. Ha egyik aktív csúcs sem tud új csúcsot megfertőzni, akkor az iteráció véget ér.

A teljesség kedvéért ismertetjük az eredetileg Granovetter [8] által bevezetett LT modellt is. A modell azt valósítja meg, hogy egy közösségbeli egyén nagy valószínűséggel átvesz szokásokat vagy követ egy trendet, ha a vele kapcsolatban lévők ezt már megtették (csordaszellem jelenség). Minden csúcs bemenő élsúlyainak összege legfeljebb 1. Továbbá minden csúcshoz hozzárendelünk egy $\theta \in [0, 1]$ *küszöbértéket*. Egy $v \in V$ csúcs fertőzött lesz, ha a belé mutató fertőzött csúcsok éleinek súlyösszege nagyobb, mint a csúcs θ értéke.

Mivel az IC és az LT modellek tartalmazznak sztochasztikus paramétereket, amely az ismeretlen hatásokat testesíti meg, így a kiértékeléséhez szükséges R -szer megismételni a szimulációt, majd a kapott értékek átlagát venni.

Az időbeli hatékonyságért az IC modellt nem az eredetileg megfogalmazott módon számítottuk, hanem a Hajdu és szerzőtársai [9] által javasolt változatot követve készítettünk egy implementációt. Készítünk R darab másolatot az eredeti G gráfról. Az él súlyait valószínűségeként alkalmazzuk. Másolatonként minden csúcsból fertőzést szimulálunk a szomszédokba. Ha a fertőzés egy másolatban nem tudott az élen átjutni, akkor az élet abban a gráfban töröljük. Ha ki akarunk értékelni egy seed halmazt, akkor csak gráfkereső algoritmusokat (mélységi vagy szélességi keresőt) kell indítanunk a seed halmaz minden csúcsából minden másolat gráfban, majd összeszámolni (multiplicitás nélkül) hány különböző csúcsot értek el egy gráfon belül és venni az átlagát ezen értékeknek. Ez tehát a várható (befolyás) értéke a seed halmaznak. Az R határozza meg a pontosságát ennek a várható értéknek. Ezen másolat gráfokat megtarthatjuk a többi kiértékeléshez is, nem kell újakat generálni.

2.2. Mohó algoritmus

Mivel a befolyás terjedés maximalizálás feladata az IC terjedési modell mellett szubmoduláris [10], ezért egy mohó algoritmus az optimális megoldás approximációját $1 - 1/e$ faktorral megadja. Ebben a kontextusban az algoritmus kezdetben készít egy S üres halmazt, beállítja a $k = 1$ -et és megoldja a problémát. A legjobb értékű csúcsot, a megoldást, behelyezi az S -be. Ezután növeli a seed halmaz lehetséges méretet eggyel és kiértékeli az összes $S \cup v$ ($v \in V \setminus S$) seed halmazt. A legjobb értékhez tartozó v csúcsot S -be helyezi. Addig ismétlődik ez, míg S mérete el nem éri az eredeti problémában megadott k elemszámot.

2.3. Egy hegymászó-jellegű algoritmus

Az imént tárgyalt mohó algoritmusban nem definiálható a lokális optimum fogalma. Ezért bevezetünk egy – az eredeti feladat megoldására alkalmas, de jelen cikkben nem arra használt – optimalizáló eljárást, amely jellegéből adódóan egy hegymászó algoritmus.

Az eljárás első lépésként egy adott (véletlenszerűen vagy más módon választott) S seed halmazból kezdi a keresést. A második lépésben kiértékeli az aktuális seed halmazt, választ véletlenszerűen az S halmazból egy csúcsot és azt leváltja az egyik (be)szomszédjára, amely jelenleg nem eleme S -nek. Ha az így kapott új $\hat{S}$ seed halmaz értéke jobb, mint az előző, akkor megtartjuk, azaz legyen $S := \hat{S}$ és ismételjük a második lépést. Ha nem, akkor visszatérünk a S seed halmazra és keresünk egy új, megfelelő szomszéd csúcsot, amellyel elvégezzük a kiértékelést és lehetséges cserét ugyanúgy, mint az előbb. Amennyiben a seed minden lehetséges szomszédjaiból alkotott kombinációt bejártuk és egyik sem rendelkezik nagyobb befolyásértékkel, akkor a seed halmazt *lokális optimumnak* nevezzük. Ebben az esetben a hegymászó eljárásunk véletlenszerűen választ egy elemet a seed halmazból és lecseréli egy csúcsra (nem feltétlen szomszédra) a gráfból, amely nem szerepel benne. Ezen csere után az elejétől ismételjük az algoritmust.

Minden kiértékelést (seed elemeket és befolyásértéket) bejegyzünk egy táblázatba a hozzátartozó lokális optimummal (utólagosan), amelybe az algoritmus eljutott a szomszédokra való lépésekkel. Amennyiben egy már látott seed halmazzal lép (ezt a táblázatból tudjuk), akkor nem számoljuk ki újra a befolyásértéket, csak kikeressük a hozzá tartozó lokális optimumot. Amennyiben nincs hozzá bejegyezve, mert rosszabb értékkel rendelkezett és nem rajta keresztül lépett tovább, illetve amennyiben kört alkotva visszajutott, akkor csak a befolyásértéket adjuk vissza. A befolyásérték számítása determinisztikus, mivel változatlanul a rögzített R darab másolat gráfon hajtódna végre. Az eljárás ilyen módon történő determinizálása csökkenti a számításigényt. Minden látogatott seed halmazt legfeljebb egy lokális optimumhoz rendelünk, mely a szomszédságban lévő seed halmazok befolyásértékétől függ.

Mint azt említettük, tehát az egyik fő különbség a mohó eljárás és az imént definiált hegymászó-jellegű módszer között, hogy az utóbbival definiálhatunk lokálisan optimális megoldásokat a problémában. Ez a lehetőség vezet minket a következő fejezethez, amiben tárgyaljuk hogyan lehet hálózatot építeni a lokális optimumokból.

3. Lokális Optimumok Hálózata - LON

Az imént bevezetett hegymászó-jellegű algoritmus végrehajtása során kaphatunk olyan csúcs halmazokat, amelyek lokális optimumként értelmezhetők. Ezen lokális optimumokból készítünk egy gráfot, a lokális optimumok hálózatát (Local

Optima Network, LON). A LON közelítéseket vizsgálták folytonos [15] és kombinatorikus optimalizálási problémákban [5]. A befolyás terjedés maximalizálás az utóbbihoz tartozik, habár alkalmaztunk néhány változtatást és új definíciókat vezettünk be. Minden, a hegymászó által megtalált lokális optimum egy csúcra a LON-ban. Két csúcra között él helyezkedik el, ha a kereső algoritmus egymás után talál rájuk: egy lokális optimumból kilépve (egy elemet lecserélve) eljutott a másik lokális optimumba. Ez egyben irányítást is ad a LON-ban. Mivel a hegymászó képes a LON már meglévő élén többször is áthaladni az építés alatt, ezért a LON súlyozott gráf.

Megállási feltételek. A LON gráf építése tulajdonképpen tetszőleges ideig tarthat. Definálni kell megállási feltételeket. Ha az építés elegendő hosszú ideje fut, akkor megfigyelhetjük, hogy egyes csúcsok befoka számottevően nagyobb a többinél. Olyan megállási feltételeket próbáltunk alkalmazni, melyekkel a LON-ban megjelennek kiemelkedő élsúlyok. Ezen feltételek a következők:

- azon pontok száma meghaladja a θ_v értéket, melyek bemenő élek számának θ_e szorosa kisebb a csúcshoz tartozó élsúlyok összegénél;
- a felderített pontok száma (amelyet ismerünk a nyilvántartáshoz alkalmazott táblázatból) meghaladja az összes lehetséges seed kombinációk számának θ_M -szeresét;
- ha a LON csúcsainak száma az eredeti G csúcsainak számának kétszeresénél több;
- a LON-ban megjelenő maximális élsúly nagyobb, mint a G éleinek számának negyede;

Kitétel ezen feltételek mellett, hogy az első feltételben említett megfelelő csúcsok száma több legyen mint 10.

H gráfok. A LON-ban a befok jellemzi egy lokális optimum vonzáskörzetét. Mivel több olyan pont van, melyekre csak a sztochasztikus lépések vezettek, de ritkán, így ezen valószínűtlen, érdektelen pontokat töröljük a LON-ból. Későbbiekben nevezzük ezt a szűrt hálózatot H -nak. A szűrés során az élsúlyokat vizsgáljuk a következő módon. Az 1 súlyú élektől kezdjük. Amennyiben az adott súlyú élek egymás utáni törlésével 2-nél nagyobb méretű komponensek jelennek meg, akkor nem hajtjuk végre ezen élek törlését és abbahagyjuk a szűrést. Ha a törlésekkel nem vágtunk le nagyobb komponenst, akkor növeljük a súlyértéket, amely szerint törlünk.

4. Numerikus kísérletek

Ebben a szakaszban bemutatunk néhány olyan jellegű elemzést, amely a fentiekben leírt módszerek alapján készült lokális optimumok hálózataira vonatkozik. Célunk tehát, hogy a befolyás terjedés maximalizálás probléma keresési terének értékkészletére adjunk jellemzést.

A kísérletek elvégzéséhez mesterségesen generált (szintetikus) véletlen hálózatokat használtunk. Három típust vizsgáltunk:

- Watts-Strogatz (WS) hálózatokat, amelyeket kisvilág gráfoknak is nevezünk [16]
- Barabási-Albert (BA) hálózatokat, amelyek építéséhez a preferenciális kapcsolódás mechanizmust használjuk, ahol a már magas foksámú csúcsok nagyobb eséllyel kapnak újabb éleket [2];
- Cooper-Frieze (CF) hálózatokat, amelyek a web gráfok növekedési mechanizmusát hivatottak modellezni [4].

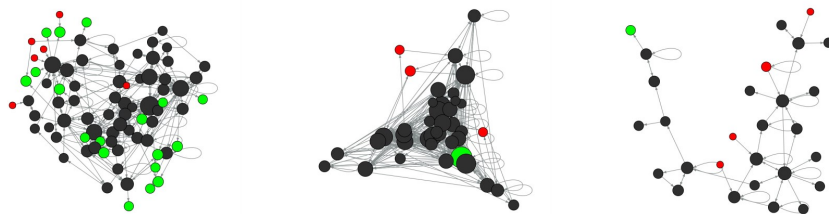
Ezekből a gráfokból több példányt is készítettünk, különböző csúcs- és élszámmal. Kísérletezéseink során azonban arra jutottunk, hogy az $n = 120$ csúcsszámú input gráfokkal kaptunk olyan eredményeket, amelyekben a lokális optimumok hálózatain az élsúlyok nagyobbak voltak 1-nél. Így végül ezekből a gráfokból válogattunk ki olyan eredményeket, amelyek jól reprezentálják az egyes típusokon kapott eredményeket. Az így kiválogatott gráfok élszámai rendre 960, 474, és 622 voltak a WS, BA és CF hálózatok esetén. Az input gráfok éleihez a súlyokat a $(0, 0, 7]$ intervallumon egyenletes eloszlású véletlenszám generátorral rendeltük hozzá.

A futtatásokat a $k = 2$ seed halmaz méretre végeztünk el, a megállási feltételek (lásd 3. szakasz) pedig a következők voltak: $\theta_v = \sqrt{|V(G)|}$, $\theta_e = 2$, és $\theta_M = 0, 6$.

4.1. H gráfok vizsgálata

A 2.3. szakaszban bevezetett hegymászó-jellegű eljárás futtatásával kapott lokális optimumok hálózataiból a 3. szakaszban ismertetett szűrési eljárással kapjuk meg a H gráfokat. A három különböző input gráftípusból válogatott inputokra kapott H gráfokat az 1. ábrán láthatjuk¹. A H gráfok csúcsainak mérete az adott csúcs befokával egyenesen arányos, míg a színek a hamarosan bevezetendő és használt közelségi központiség értékeket reprezentálják. Ezekről az ábrákról egyelőre annyi megállapítható, hogy a három különböző input gráfra három különböző szűrt lokális optimumok hálózatát kaptuk, amely azonnal sugallja a befolyás terjedés maximalizálás feladat diverz jellegű viselkedését.

¹Szeretnénk hangsúlyozni, hogy itt nem az eredeti G input gráfok vizualizációit láthatjuk.



(a) Watts-Strogatz

(b) Barabási-Albert

(c) Cooper-Frieze

1. ábra. Lokális optimumok hálózatai, szűrt változatok (H gráfok)

A 2. ábrán az egyes H gráfok behatóbb elemzését láthatjuk. A vizsgálati módszer lehetővé teszi, hogy rálátást szerezzünk az egyes lokális optimumok vonzásokörzeteiről. Ehhez a hálózatkutatásban használt közelségi központiság (closeness centrality) fogalmát használtuk. Ennek kiszámítása a

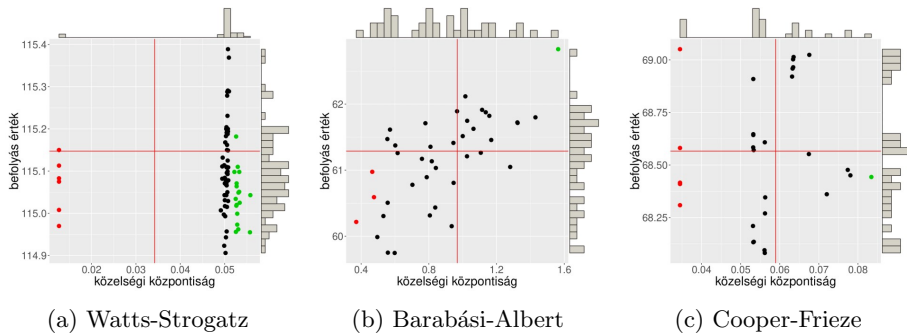
$$C(u) = \frac{n - 1}{\sum_{v \in V(H)} d(v, u)}$$

képlettel határozható meg, ahol n a csúcsok száma H -ban, $d(v, u)$ pedig a v -ből u -ba menő irányított legrövidebb út hossza². Az ábrákon pirossal a C értékek alsó 10 percentilisébe, zölddel pedig a felső 10 percentilisébe eső csúcsokat színeztük, a többi csúcs pedig feketével van feltüntetve. Ezt a színezést tartottuk meg egyben az 1. és 4. ábrákon is egyaránt.

A 2. ábrán tehát a H gráf csúcsainak közelségi központisági értékei és a befolyás értékük közötti összefüggést vizsgálhatjuk. Emlékeztetőül, a H gráf csúcsai az eredeti G input gráf (jelen példákban $k = 2$ elemű) csúcshalmazai. Ez az elemzés lehetővé teszi, hogy egy adott input gráfot megoldhatóság szerint kategorizáljuk könnyűnek vagy nehéznek. A magas befolyásérték jelzi a legjobb talált megoldáshoz való közelséget. Az alacsony C érték jelenti, hogy az adott pont nehezebben volt elérhető a hegymászó algoritmussal. Amennyiben a pontok többsége magas befolyás értékkel és magas C értékkel rendelkezik, akkor azt a feladatot könnyűnek nevezhetjük.

Mindhárom input gráfra jellemző, hogy a talált lokális optimumok értékben egymástól kevésben különböznek. A kérdés csak az, hogy vajon ezek mennyire esnek távolra egymástól?

²A d kiszámításánál az élek súlyainak reciprokát használtuk, követve így azt a konvenciót, hogy magasabb C érték jelenti azt, hogy az adott csúcs átlagosan közelebb van a többi csúcsához.



2. ábra. Közelégi központosság értékek a H gráfokban

A 2a. ábra alapján a vizsgált Watts-Strogatz gráf számos nagyon magas értékű seed halmazt tartalmaz, amelyek közül a legmagasabb befolyás értéket elérő egyben magas C értékkel is rendelkezik, ezért annak megtalálása viszonylag könnyű. Ugyanakkor láthatjuk, hogy van egy másik régió is, alacsonyabb fertőzés értéket eredményező és egyben nehezebben elérhető seed halmazokkal.

A 2b. ábra az előzőtől különbözik abban, hogy a közelégi központossági értékek nagyjából egyenletesen oszlanak el. Ez az értékkészletben történő egyenletes szétterülésükre utal. A legmagasabb befolyás értékkel rendelkező seed halmaz egyben a legmagasabb C értékkel is rendelkezik, így ennek megtalálására a hegymászó algoritmusnak jó esélye van.

A 2c. ábra a másik két ábrától annyiban különbözik, hogy több, befolyás értékben minimálisan különböző, a legjobb megoldáshoz közeli seed halmaz jelenlétét mutatja. Ezek közül az abszolút legjobb értékű egyben a legnehezebben elérhető is.

4.2. Csúcsalmazok egy távolsága: VSD

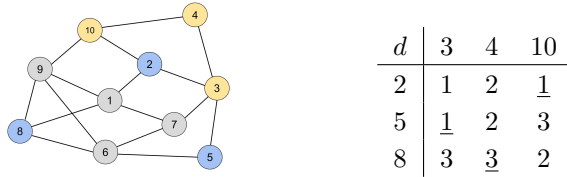
A keresés során alkalmas seed halmazok megtalálása a cél. Ahhoz, hogy az így megtalált halmazok közötti összefüggéseket vizsgáljuk, bevezetünk egy csúcsalmazok távolságának kifejezésére alkalmas mértéket, amelyet VSD -vel (*Vertex Set Distance*) jelölünk. Legyenek V_1 és V_2 a G gráf csúcsalmazának k -elemű részalmazai. Legyen

$$VSD(V_1, V_2) = \min_{\pi \in S_k} \left(\sum_{i=1}^k d(V_{1_i}, V_{2_{\pi(i)}}) \right),$$

ahol $d : V(G) \times V(G) \rightarrow \mathbb{N}_0$, a gráfon vett távolság, és S_k az első k természetes számhoz tartozó permutációk halmaza.

Annak a formális bizonyítását, hogy az így definiált VSD valóban távolság fogalom, a függelékben találja az olvasó.

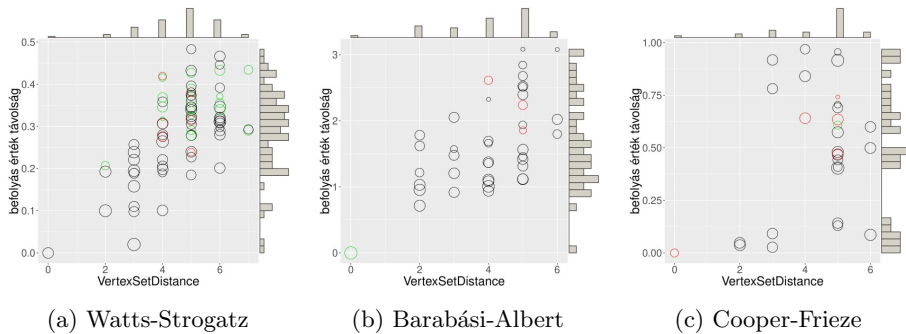
A VSD kiszámítását úgy végezzük, hogy egy hozzárendelési feladatként értelmezzük a problémát. Az egyik seed halmaz elemeit szeretnénk hozzárendelni a másik seed halmaz elemeihez. A költségek mátrixa az eredeti G gráf súly nélküli változatából számított távolság részmatrix lesz, amely csak az érintett elemekhez kapcsolódó értékekből áll. Az irányítottságtól eltekinthetünk, mert minden élhez létezik annak ellenkező irányú párja. A megoldás értéke lesz a VSD .



3. ábra. VSD kiszámítása hozzárendelési feladatként

4.2.1. Példa. Egy könnyen áttekinthető példát a 3. ábrán láthatunk, ahol G egy 10 csúcsból álló gráf, a két csúcshalmaz pedig $S_1 = \{2, 5, 8\}$ (az ábrán kék színnel jelölve) és $S_2 = \{3, 4, 10\}$ (az ábrán sárga színnel jelölve). A kapcsolódó hozzárendelési feladatot a jobb oldali táblázat írja le, amiből a $VSD(S_1, S_2) = 5$ megoldás adódik.

Az így bevezetett VSD távolság mértéket használhatjuk annak a kiemzésére, hogy a megtalált lokális optimumok közül a legmagasabb értékűhöz viszonyítva a többiek (VSD értelemben) milyen távol helyezkednek el az eredeti G gráfban. Ezeket az értékeket a legjobb megoldás befolyás értékétől vett távolsággal vetjük össze.



4. ábra. VSD értékek a G input gráfokon

A 4. ábrán a három input gráfra kapott eredményeket láthatjuk. A körök mérete az adott seed halmaz *vonzáskörzetével* arányos: a hegymászó, mint lokális optimalizáló futtatása során hányszor jutottunk egy lokális optimumba. Az origóban tehát az adott G gráfban talált legjobb megoldás helyezkedik el, a távolságokat ehhez a seed halmazhoz viszonyítjuk. A 4b. ábra szerint a BA hálózat (ezen példányának) értelmezési tartománya ún. *big valley* szerkezetű [3], amelyben a lehetséges megoldások a globális/legjobb megoldástól vett távolságai (esetünkben: a VSD értékek) és az ezekhez tartozó függvényértékek között erős a korreláció. A globális optimumot ugyan körbeveszik a lokális optimumok, ezek azonban mind egyre rosszabb értékűek, ahogy távolodunk a legjobb megoldástól. Ehhez hasonlít a 4a. ábrán látható WS hálózat is, bár abban találtunk egy, a legjobb megoldáshoz nagyon közeli befolyás értékű seed halmazt is. Végül a 4c. ábrán újra igazolást nyer a CF hálózatok a másik két típustól vett különbözősége, hiszen itt összesen hét darab magas értékű lokális optimumot találtunk, amelyek az egyébként nehezen megtalálható (erre a piros színezés utal) legjobb megoldástól, VSD értelemben, egyre távolabb helyezkednek el.

5. Konklúzió

A szakirodalomban sok szempontból vizsgált, erősen alkalmazás orientált problémát, a gráfokon értelmezett befolyás terjedés maximalizálás megoldásainak tulajdonságait vizsgáltuk. Megközelítésünk újdonságát az adta, hogy a felhasznált hegymászó-jellegű algoritmussal a keresési tér olyan leképezését vizsgáltuk, amely lehetővé teszi a feladat strukturális tulajdonságainak mélyebb megértését. Ehhez felhasználtuk a lokális optimumok hálózatának (LON) fogalmát. Ennek segítségével három, szerkezetileg különböző véletlen gráfon, mint inputon kimutattuk, hogy a probléma különböző értékkészlet struktúrákhoz vezet. Az így kapott eredmények lehetőséget adnak célzott keresési eljárások definiálására, ennek részletes kidolgozását jövőbeli kutatási tervként jelöljük ki.

Függelék

Megmutatjuk, hogy a 4.2. szakaszban bevezetett VSD függvény nemnegatív, szimmetrikus és teljesíti a háromszög egyenlőtlenséget, azaz valóban távolság fogalom.

A VSD irányítatlan gráfokon értelmezett legrövidebb utakon alapszik, ami szimmetrikus. Az eredeti G gráf irányított, de minden élnek van párja az ellenkező irányba és a súlyok figyelmen kívül voltak hagyva (gráf távolság mindkét irányba megegyezik). Gráfbeli távolságok összege független a tagok sorrendjétől, ezért VSD szimmetrikus, azaz $VSD(V_1, V_2) = VSD(V_2, V_1)$ teljesül.

Ahogy előzőleg említettük, gráf távolságon alapszik, amelyek értéke mindig nemnegatív. Ezen nemnegatív számok összege is nem negatív. Legyen V_1, V_2 a G

gráf csúcsaiból álló, k különböző elemeket tartalmazó halmazok. A $VSD(V_1, V_2) = 0$ akkor és csakis akkor, ha $V_1 = V_2$. Ekkor minden V_1 beli elem 0 távolságra van a V_2 -beli megfelelőjétől (önmagától), a lehetséges úthosszak összegeinek minimuma pedig 0. Ha V_1 és V_2 egy elembe is térnek el, akkor legalább egy 0-nál hosszabb útnak kell lennie az elemek között, a VSD értéke nagyobb lesz 0-nál, amivel a nemnegativitást beláttuk.

Végül, a háromszög egyenlőtlenséghez a

$$VSD(V_1, V_2) \leq VSD(V_1, V_3) + VSD(V_2, V_3)$$

egyenlőtlenségnek teljesülnie kell $V(G)$ bármely k elemű V_1, V_2, V_3 részhalmazára. Az indirekt bizonyításhoz tegyük fel, hogy

$$VSD(V_1, V_2) > VSD(V_1, V_3) + VSD(V_2, V_3).$$

Mivel VSD szimmetrikus, ezért a $VSD(V_1, V_2) > VSD(V_1, V_3) + VSD(V_3, V_2)$ egyenlőtlenség is teljesül. Tudjuk, hogy $VSD(V_1, V_3)$ megegyezik a megfelelő hozzárendelési feladat minimális költségű megoldásának értékével. Ez igaz a V_3, V_2 közötti VSD értékre is. Ekkor vannak utak a V_1 és V_2 elemei között (legalább a V_3 elemeit érintők). Ezen utakból szerkesztett egy hozzárendelési feladat optimális megoldásának értéke legyen D . Meghatározott csúcsok érintésével nem csökkennek a távolságok, ezáltal D sem csökkenhet, ha a közvetlen legrövidebb utak helyett másokat használunk. Vagyis, ha D a legrövidebb utakból szerkesztett feladat megoldása, akkor $D \leq VSD(V_1, V_3) + VSD(V_3, V_2)$. Ugyanakkor $D = VSD(V_1, V_2)$, aminek nagyobbának kell lennie a két előző VSD érték összegénél, amely ellentmondás.

Köszönetnyilvánítás

Jelen kutatás eredményei az „Integrált kutatói utánpótlás-képzési program az informatika és számítástudomány diszciplináris területein” című, EFOP-3.6.3-VEKOP-16-2017-0002 számú projekt, valamint a TUDFO/47138-1/2019-ITM számú projekt támogatásaival készültek.

Hivatkozások

- [1] AGGARWAL, C.C., LIN, S. AND PHILIP, S.Y.: *On influential node discovery in dynamic social networks*, In: Proceedings of the 2012 SIAM International Conference on Data Mining, pp. 636-647. SIAM (2012). DOI: [10.1137/1.9781611972825.55](https://doi.org/10.1137/1.9781611972825.55)
- [2] ALBERT, R. AND BARABÁSI, A.L.: *Statistical mechanics of complex networks*, Reviews of Modern Physics, Vol. **74** No. **1**, pp. 47-97 (2002). DOI: [10.1103/RevModPhys.74.47](https://doi.org/10.1103/RevModPhys.74.47)

- [3] MORITA, M., OCHIAI, H., TAMURA, K. AND YASUDA, K.: *Multi-point search combinatorial optimization method based on neighborhood search using evaluation of big valley structure*, In IEEE International Conference on Systems, Man, and Cybernetics, pp. 2835-2840 (2015). DOI: [10.1109/SMC.2015.494](https://doi.org/10.1109/SMC.2015.494)
- [4] COOPER, C. AND FRIEZE, A.: *A general model of web graphs*, Random Structures & Algorithms 22, pp. 311-335 (2003). DOI: [10.1002/rsa.10084](https://doi.org/10.1002/rsa.10084)
- [5] DAOLIO, F., TOMASSINI, M., VÉREL, S. AND OCHOA, G.: *Communities of Minima in Local Optima Networks of Combinatorial Spaces*, Physica A: Statistical Mechanics and its Applications, Elsevier, Vol. **390** No. **9**, pp. 1684-1694 (2011). DOI: [10.1016/j.physa.2011.01.005](https://doi.org/10.1016/j.physa.2011.01.005)
- [6] DOMINGOS, P. AND RICHARDSON, M.: *Mining the Network Value of Customers*, In Proceedings of 7th International Conference on Knowledge Discovery and Data Mining, pp. 57-66, (2001). DOI: [10.1145/502512.502525](https://doi.org/10.1145/502512.502525)
- [7] GOYAL, A., LU, W. AND LAKSHMANAN, L. V. S.: *CELF++: Optimizing the greedy algorithm for influence maximization in social networks*, In: Proceedings of the 20th International Conference Companion on World Wide Web., pp. 47-48 (2011). DOI: [10.1145/1963192.1963217](https://doi.org/10.1145/1963192.1963217)
- [8] GRANOVETTER, M.: *Threshold models of collective behavior*, American Journal of Sociology, Vol. **83** No. **6**, 1420-1443 (1978). DOI: [10.1086/226707](https://doi.org/10.1086/226707)
- [9] HAJDU, L., KRÉSZ, M. AND BÓTA, A.: *Community based influence maximization in the Independent Cascade model*, Federated Conference on Computer Science and Information Systems (FedCSIS), IEEE, (2018). DOI: [10.15439/2018F201](https://doi.org/10.15439/2018F201)
- [10] KEMPE, D., KLEINBERG, J. AND TARDOS, É.: *Maximizing the spread of influence through a social network*, Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining. ACM (2003). DOI: [10.1145/956750.956769](https://doi.org/10.1145/956750.956769)
- [11] LEE, J.R. AND CHUNG, C.W.: *A fast approximation for influence maximization in large social networks*, In: Proceedings of the 23rd International Conference on World Wide Web Companion, pp. 1157-1162, (2014). DOI: [10.1145/2567948.2580063](https://doi.org/10.1145/2567948.2580063)
- [12] LESKOVEC, J., KRAUSE, A., GUESTRIN, C., FALOUTSOS, C., VANBRIESEN, J. AND GLANCE, N.: *Cost-effective outbreak detection in networks*, In: Proceedings of the 13th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pp. 420-429 (2007). DOI: [10.1145/1281192.1281239](https://doi.org/10.1145/1281192.1281239)
- [13] NEMHAUSER, G., AND WOLSEY, L. AND FISHER, M.: *An analysis of the approximations for maximizing submodular set functions*, Mathematical Programming, Vol. **14**, pp. 265-294 (1978). DOI: [10.1007/BF01588971](https://doi.org/10.1007/BF01588971)
- [14] RAHIMKHANI, K., ALEAHMAD, A., RAHGOZAR, M. AND MOEINI, A.: *A fast algorithm for finding most influential people based on the linear threshold model*, Expert Systems with Applications, Vol. **42** No. **3**, pp. 1353-1361 (2015). DOI: [10.1016/j.eswa.2014.09.037](https://doi.org/10.1016/j.eswa.2014.09.037)
- [15] VINKÓ, T. AND GELLE, K.: *Basin Hopping Networks of Continuous Global Optimization Problems*, Central European Journal of Operations Research, Vol. **25**, pp. 985-1006 (2017). DOI: [10.1007/s10100-017-0480-0](https://doi.org/10.1007/s10100-017-0480-0)
- [16] WATTS, D.J. AND STROGATZ, S.H.: *Collective dynamics of ‘small-world’ networks*, Nature, Vol. **393** No. **6684**, pp. 440-442 (1998). DOI: [10.1038/30918](https://doi.org/10.1038/30918)



Homolya Viktor programtervező informatikus diplomáját a Szegedi Tudományegyetemen szerezte. 2018 óta az SZTE Számítógépes Optimalizálás Tanszékén doktorandusz, ahol a kutatómunka mellett aktívan részt vesz az oktatási feladatok elvégzésében is. Kutatási területe az optimalizálási módszerek és feladatok elemzése a hálózattudomány eszközeivel.

HOMOLYA VIKTOR

Szegedi Tudományegyetem, Informatikai Intézet,
Számítógépes Optimalizálás Tanszék
homolyav@inf.u-szeged.hu



Vinkó Tamás programtervező matematikus, habilitált egyetemi docens. Diplomáját és fokozatait a Szegedi Tudományegyetemen szerezte. Posztdoktori kutatóként több mint 6 évet töltött Hollandiában az Európai Űrügynökségnél és a Delfti Műszaki Egyetemen. Kutatómunkájában a hálózattudomány és az optimalizálás határterületeit vizsgálja.

VINKÓ TAMÁS

Szegedi Tudományegyetem, Informatikai Intézet,
Számítógépes Optimalizálás Tanszék
tvinko@inf.u-szeged.hu

ON THE OPTIMA NETWORKS OF INFLUENCE MAXIMIZATION

VIKTOR HOMOLYA, TAMÁS VINKÓ

The influence maximization problem is investigated from a new perspective. A hill-climber algorithm is introduced for the purpose of detecting local optima and their relationships in this discrete optimization problem. The proposed method is demonstrated by exploring the target set of the search space resulted from computational experiments on structurally different random graphs.