

XXXII. Országos Tudományos Diákköri Konferencia

Közösségek keresése hálózatokban gráf Voronoj diagramok segítségével

Szerző:

Deritei Dávid Babeş–Bolyai Tudományegyetem, Kolozsvár, Fizika Kar, Számítógépes
fizika szak, MSc, II. év

Témavezető:

Dr. Ercsey-Ravasz Mária, tudományos kutató
Babeş–Bolyai Tudományegyetem, Kolozsvár, Fizika Kar

Tartalom

1	Bevezető	3
2	Bevezetés a hálózat klaszterezésbe és közösség detektálásba	4
3	A gráf Voronoi diagramok és alkalmazásuk hálózatok particionálására	8
4	A távolság-mérték és a generátorpontok azonosítása	11
4.1	Az élklaszterezési együttható	11
4.2	A relatív lokális sűrűség	12
4.3	Az r paraméter	14
5	Ellenőrzési módszerek	15
5.1	Benchmark hálózatok	15
5.2	A modularitás függvény	16
6	Az eredmények tárgyalása	18
6.1	Benchmark hálózatokon való tesztelés	18
6.2	Valós hálózatokon való tesztelés	20
6.3	Skálázódás	22
7	Következtetések	24
7.1	Publikációk és konferenciák	26

Kivonat

Közösségek pontos és hatékony detektálása komplex hálózatokban egy kulcsfontosságú kihívás a hálózati tudományban és változatos alkalmazásaiban. Ilyenek a szociális hálózatok, sejtbiológia, számítógépes hálózatok stb. Ha meghatározunk egy távolságmértéket a hálózati csomópontok között és egy generátorpont készletet, a gráfokra kiterjesztett Voronoi diagrammok egyértelmű particionálását adják a hálózatnak, a csomópontok generátorpontokhoz való hálózati közelsége alapján. A kihívás a megfelelő távolságmérték generátorpontok azonosítása. Két csomópont közötti távolságot a közöttük levő legrövidebb úttal definiáljuk egy súlyozott élű hálón, ahol az élek "hossza" az úgynevezett élklasszterezési együttható inverze. Egy mértéket rendelünk minden csomóponthoz, amely a csomópont és az első szomszédai által alkotott algráf relatív sűrűsége. Generátor pontoknak tekintjük azokat a csomópontokat amelyeknek a legnagyobb a relatív sűrűsége egy adott r sugarú környezetben belül. Elvégezve a particionálást különböző r értékekre azonosítani lehet az optimális klaszterzést.

1 Bevezető

A hálózati tudomány a napjaink egyik legszéleskörűbben alkalmazott interdiszciplináris tudománya. Az orvostudománytól, a szociológián keresztül, terrorista szervezetek feltérképezéséig rengeteg alkalmazását találjuk. A fizika módszereinek alkalmazása a hálózati tudományban egészen új távlatokat nyitott meg ezen a területen.

A közösségek hatékony detektálása egy kulcsfontosságú kihívás a hálózati tudományban. Jelentőségét mi sem bizonyítja jobban mint széles körű alkalmazása a tudományok szinte teljes skáláján. Olyan esetekben ahol egy tanulmányozott rendszert hálózattal vagy gráffal modellezni tudunk, felmerül a közösségek azonosításának a problémája is. A közösségeket tartalmazó hálózatok paradigma-példái a szociális hálózatok. Az emberek

hajlamosak közösségeket kialakítani, ilyenek a baráti körök, családi közösségek, munkahelyi közösségek stb. Ezeknek a közösségeknek a detektálása fontos lehet szociológiában, piackutatásban, közgazdaságban, kommunikációs problémák megoldásában. Erre nagyon jó példa amazon.com vagy más hasonló oldalak rendszere, amely a termékeinek célzott ajánlását aszerint oldja meg, hogy milyen érdeklődési körbe, közösségbe tartozunk. A sejtbiológiában felfedezett fehérjék közti kölcsönhatások által kialakított hálózatban, a fehérjék különböző funkcionális csoportjai is úgynevezett klaszterekbe tömörülnek, és ezek hatékony feltérképezésének nagy jelentősége van a sejtek működésének megértésében vagy betegségek, pl a rák kutatásában is. Hasonló klaszterek alakulnak ki a World Wide Web, azaz az internetes weboldalak hálózatában, ilyenkor a különböző link-farmok és más közösségek azonosítása fontos információ lehet a világháló kutatóinak, fejlesztőinek [1]. Jelenlegi kutatásunk kihívása egy több tízmilliós nagyságrendű tudományos publikációkat és hivatkozásait tartalmazó hálózat klaszterezése. Célunk a különböző tudományterületek, esetleges hibridizálódó és átfedő tudományterületek pontos azonosítása. Mivel egy hatalmas hálózattal van dolgunk, első célunk egy pontos de komplexitás szempontjából hatékony algoritmus kifejlesztése.

2 Bevezetés a hálózat klaszterezésbe és közösség detektálásba

Az irodalomban rengeteg módszert és koncepciót találunk a közösségek detektálására, ezekből fogunk néhány alapvető és kutatásunk szempontjából jelentős módszert bemutatni. A probléma ezekkel a módszerekkel és általában a közösség detektálással, hogy még nincs egy univerzálisan elfogadott pontos matematikai definíciója annak, hogy mit értünk közösség alatt. Ennek egyik oka, hogy a különböző valós rendszerekben különbözőképpen jelennek meg a közösségek, továbbá jelen van az átfedő és a hierarchikus közösségek problémája is ami tovább nehezíti egy rigurózus matematikai definíció létrejöttét [1]. Emi-

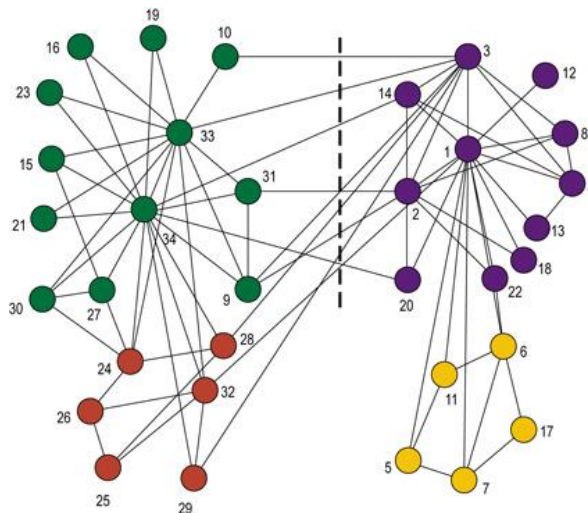
att a módszerek többnyire empirikusak és csak kísérleti hálózatokon lehet ellenőrizni a hatékonyságukat. Az általános kiindulási pont az, hogy közösségnek tekintjük azt a részét egy gráfnak (hálózatnak), amely csomópontjai között sűrűbben vannak élek (kapcsolatok), mint a gráf egészével, azaz több kapcsolat van a közösségen “belül” mint kifelé. Az elmúlt évek során több különböző definíció született, megkülönböztethetünk lokális, globális és csomópont hasonlóságon alapuló definíciókat.

A közösség detektáló algoritmusoknak az az alapvető feladata, hogy ezeket az algráfokat, régiókat kimutassa és azonosítsa csupán a hálózati topológia (adott esetben egy éllista vagy adjacencia mátrix) alapján. Megkülönböztetjük egy hálózat particionálását és közösségek detektálását. Particionálás alatt értjük azt, amikor minden csomópontot hozzárendelünk valamilyen közösséghez (particióhoz), de csakis egyhez, azaz tulajdonképpen felcsoportosítjuk a hálózatot különálló részekre. Első lépésben a saját módszerünk is ezt a szerepkört látja el.

A közösség detektálás már egy jóval összetettebb probléma, ez esetben igyekszünk a különböző közösségek hierarchikus felépítését és az esetleges átfedéseket is detektálni. Hierarchikus felépítés alatt értjük azt mikor egy közösségen belül több kisebb közösség is kialakul, azokon belül még kisebb közösségek és így tovább. Például a kutatásunk során tanulmányozott tudományos publikációk hálózatában a legnagyobb közösségeket a különböző tudományterületek jelentik (fizika, biológia, orvostudomány stb.), de ezeken belül is kialakulnak a kisebb klaszterek, ilyenek lehetnek a különböző specifikus területek (pl. fizikán belül atomfizika, szilárdtest fizika, húrelmélet stb.), amelyek a közösségükön belül jóval gyakrabban hivatkoznak, mint más területek képviselőire. Ugyanitt tettenérhetjük a közösségek átfedésének jelenségét is, az olyan területek esetében amelyek több tudományág hibridizációi (pl. biofizika, orvosi fizika stb.), vagy a különböző inderdiszciplináris tudományágakon keresztül (pl. hálózati tudomány). Ellenőrizni egy módszert, ha valós hálózaton teszteljük az úgynevezett meta-adatok alapján lehet, azaz a valóságból gyűjtött közösségi információ alapján. Erre szemléletes példa a Zachary féle karate klub közösségi

hálózata (1. Ábra).

Egy másik nagyon fontos kritériuma egy közösség detektáló algoritmusnak, a számítógépes



Ábra 1: A Zachary féle karate klub tipikus példájává vált a közösségi hálózatoknak. A klub egy vezetőségbeli konfliktus után felbomlott és a vezetőség különböző képviselőinek pártfogói külön közösségekre szakadtak. A hálózat a baráti kapcsolatok alapján épül fel, és egyértelműen szemlélteti a közösségi struktúrát.

komplexitás, azaz annak a becslése, hogy mennyi erőforrásra van szüksége a feladat elvégzéséhez. Az algoritmusok skálázódása általában n -től, a csomópontok számától, és m -től, az élek számától függ, $O(n^\alpha m^\beta)$ alakban. A módszerek esetében mindig igyekszünk az α és a β értékét minimalizálni. Ez leginkább akkor merül fel ha nagyon nagy hálózatokat szeretnénk kielemezni. Kutatásunk során igyekeztünk egy átlagon felüli költség/hatékonyság aránnyal operáló módszert kidolgozni.

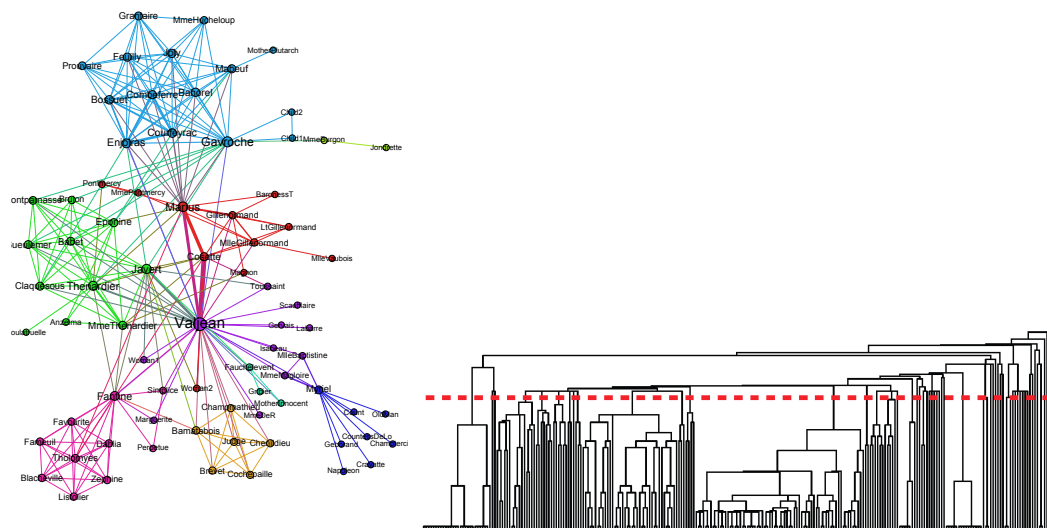
Ha lokális klaszter-definíciókból indulunk ki (klikkek keresése [2], k -core [3], k -plex [4], erős és gyenge klaszter definíciók [5] stb.) általában NP-teljes problémákba botlunk. Globális definíciók esetén hasonló a helyzet, az egyik legnépszerűbb globális definícióra, az úgynevezett modularitás függvényre [6] a későbbiekben részletesen kitérünk.

A hasonlóságon alapuló definíciók lényege, hogy általában valamilyen távolságot definiálnak a csomópontok között (pl legrövidebb út, Manhattan távolság stb.) [7] [8] és ezek alapján kategorizálják az egymáshoz leginkább közel lévő, avagy hasonló csomópontokat. Ennek

előnye, hogy a hasonlóság információt ad az esetleges átfedő közösségekről (fuzzy kategorizálás).

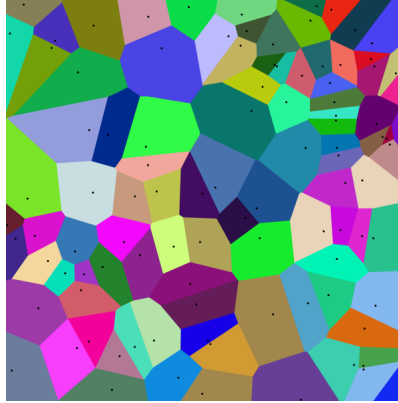
A nagy komplexitást és előnytelen skálázódást kiküszöbölendő számos nem-determinisztikus algoritmust dolgoztak ki (pl. random walk, escape probability, label propagation stb.), amelyek nagy pontossággal és jó skálázódással operálnak [9, 10, 18, 20]. Azonban egy időben fejlődő hálózatonál, amelynél gyakorta kell újraszámolni a klaszterezést ezek nem túl megbízhatóak, mert ugyanarra a hálózatra is hajlamosak különböző eredményt adni. Ugyan a különbségek általában minimálisak, de a hiba egy csomópontonként növekedő hálózatonál jelentős lehet.

Fontos megemlíteni az úgynevezett osztó és agglomeráló algoritmusokat is. Ezek egy-egy mértéket rendelnek minden élhez/csomóponthoz amely azt jellemzi, mennyire intra (klaszteren belüli) vagy inter (klaszterek közötti, klasztereket összekötő) egy él/csomópont. Ez alapján elkezdik vagy “fentről lefele”, kiszedni (osztó) vagy “lentől felfele” berakni (agglomeratív) az elemeket, az őket jellemző mérték szerinti sorrendben. Osztó algoritmusok esetén a teljes hálóból indulunk ki és addig iteráljuk az elemek kiszedését, ameddig a hálózat darabjaira nem hull. Agglomeráló algoritmusok esetén pont az ellenkező folyamat játszódik le, itt addig építjük a hálózatot amíg teljes nem lesz. Ezáltal létrehozhatjuk az úgynevezett dendrogramot is, amelyet megfelelő helyen “elvágvá” megkapjuk a klaszterezést, és jó esetben a hierarchiát is. Kutatásunk során több ilyen algoritmust is teszteltünk különböző mértékeket használva, mint pl. a betweenness [6,11], az élklaszterezési együttható (edge clustering coefficient) [5], stb. , de kizárólagosan egyik sem bizonyult elég hatékonnak céljaink szempontjából.



3 A gráf Voronoi diagramok és alkalmazásuk hálózatok particionálására

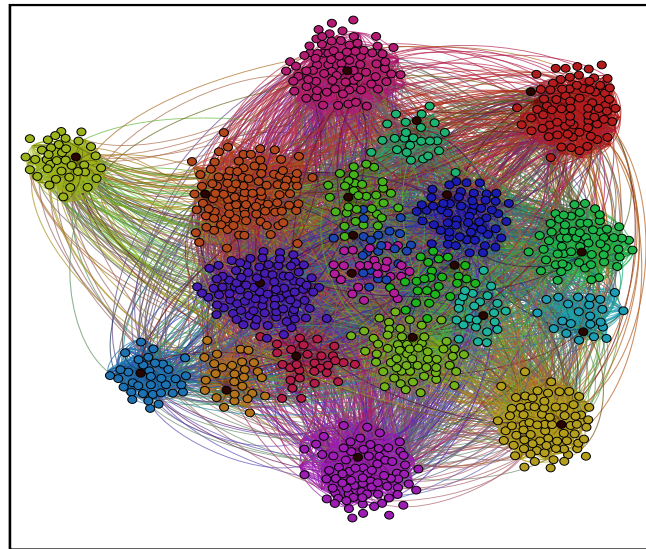
alkalmazzák, pl. nagyvárosok közlekedési útvonalainak optimalizálásánál [13].



Ábra 3: Az euklidészi térben számított “klasszikus” Voronoi diagramok. A sík minden pontja a hozzá legközelebb álló generátorponthoz tartozik, így generálva a különböző színekkel jelölt Voronoi cellákat.

Amint kutatásunkból kiderült, a gráf Voronoi cellák kiváló particionálását adják egy közösségi struktúrával rendelkező gráfnak. A 4. Ábrán egy benchmark hálózat klaszterezett ábrája látható. A klaszterek látványosan elkülönülnek az úgynevezett Force Atlas algoritmusnak köszönhetően, amely molekuláris dinamika alapokon működő algoritmus, egyszerűen vonzó erőt fejt ki a kapcsolt csomópontok között és taszító erőt a nem kapcsolt csomópontok között. Így a sűrűbben csatolt klaszterek, amelyek tulajdonképpen a közösségek, látványosan elkülönülnek [14]. Természetesen ezt az algoritmust csak kis hálózatokon, többnyire szemléltetés céljából érdemes használni. Első próbálkozásként a látványosan elkülönülő klaszterekben véletlenszerűen “kézzel” kiválasztottunk egy-egy generátorpontot (fekete csomópontok), és ezeket használva legeneráltuk a megfelelő gráf Voronoi cellákat, majd ez alapján megszíneztük a csomópontokat. Az eredmény nagyon látványos, a klaszterek tökéletesen particionálódnak. Fontos megemlíteni, hogy ebben az esetben az élek súlya egységnyi azaz nincs hossz rendelve az élekhez, ennek ellenére a klaszterezés mégis nagyon jól működött. Ez annak tudható be, hogy a hálózat sűrű és az inter-intra élek számának aránya elég kicsi. Ez azt sugallja, hogy ha kidolgozunk

egy klaszterezési módszert az nagyon robusztus lehet, mivel most a klasztereken belül véletlenszerűen választottunk ki egy-egy csomópontot és a klaszterezés így is tökéletes lett.



Ábra 4: Egy 2000 csomópontból álló hálózat, amelyben a Force Atlas algoritmusnak köszönhetően elkülönülő klaszterekben kézzel kiválasztott generátorpontokra futtattuk le a gráf Voronoi cellákat azonosító algoritmust, majd a csomópontokat megszíneztük a cellák szerint. A fekete csomópontok az általunk véletlenszerűen kiválasztott generátorpontok.

Természetesen a valós hálózatok nem ilyen tökéletesek, és sok véletlenszerű átkötés lehet egymástól egyébként “távol” eső klaszterek között is, és ezek, mivel a távolság nem euklidészi felboríthatják a particionáltságot. Ezért célszerű egy távolságmértéket bevezetni, amely alapján súlyozzuk az éleket, és a legrövidebb távolság számításakor ezt az él-hosszt is figyelembe vesszük majd. Ésszerű ötletnek tűnik egy olyan mértéket rendelni az élekhez, amely azzal arányos, hogy egy él mekkora valószínűséggel inter, azaz klaszterek közötti, avagy intra, azaz klaszteren belüli. Így a klaszterek közötti élek “hosszabbak” lesznek, és a Voronoi cellák még inkább hajlamosak lesznek a klasztereken belül kialakulni. Ennek az él-skálázó kritériumnak tökéletesen megfelelnek az előző fejezetben

érintett osztó és agglomeráló algoritmusok mérőszámai.

A másik kihívás, az a megfelelő generátorpontok azonosítása. Az előző példában már a klaszterek előzetes ismeretében választottunk ki egy-egy generátorpontot minden klaszteren belül. Ami szükséges, az egy olyan módszer, amely csak a hálózati topológiából kiindulva képes úgy kiválasztani generátorpontokat, hogy lehetőleg minden klaszteren belül, csak egy legyen, így a particionálás nagy valószínűséggel tökéletes lesz. A következő fejezetben bemutatunk mindkét problémára egy-egy viszonylag egyszerű, de jól működő alternatívát. A dolgozat végén tárgyaljuk, hogy ezeken hogyan lehetne még javítani.

4 A távolság-mérték és a generátorpontok azonosítása

4.1 Az élklaszterezési együttható

Az egyik népszerű és nagyon hasznos él-jellemző mérték, az a 2004-ben Radicchi által bevezetett élklaszterezési együttható (edge clustering coefficient-vagy *ECC*) [5]. Amellett, hogy lokálisan kiszámítható, így lineárisan skálázódik az élek számával ($O(m)$), nagyon hatékony klaszterező mértéknek bizonyult kutatásunk során, úgy osztó, mint agglomeratív megközelítésben, főleg sűrű hálók esetén. Az *ECC* definíció szerint egy éleket jellemző mérték, amelyet a következőképpen számítunk ki: a háromszögek száma, amelyekben a vizsgált él benne van, osztva a háromszögek számával, amelyekbe potenciálisan beletartozhatna, ismerve a két szomszédos csomópont fokszámát. Az i csomópontot a j csomóponttal összekötő él *ECC*-je:

$$C_{i,j} = \frac{z_{i,j} + 1}{\min[(k_i - 1), (k_j - 1)]} \quad (1)$$

ahol $z_{i,j}$ a háromszögek száma, amelyekbe az él beletartozik, és $\min[(k_i - 1), (k_j - 1)]$ a potenciális háromszögek száma amelyek magukba foglalhanák, azaz a kisebb érték a két adjacens csomópont fokszámai közül, mínusz egy (a vizsgált él). Az ötlet a mérték mögött

az, hogy a közösségeket összekötő élek általában kevés vagy nulla háromszögben vannak jelen, viszont az adjacens csomópontjainak nagy valószínűséggel lehet nagy fokszáma, azaz a $C_{i,j}$ értékük kicsi. Azonban a klasztereken belüli élek nagyobb valószínűséggel alkotnak háromszögeket (sűrűbbek a kapcsolatok). Minél kisebb a $C_{i,j}$ érték, annál valószínűbb, hogy egy él klaszterek közötti. A számlálóban szereplő 1-es tagra azért van szükség, hogy kiküszöböljük a 0-ás értékű együtthatókat (pl. levél-szerű csomópotok esetén, amelyek a klaszterezés szempontjából nem túl jelentősek). A módszernek van egy másik változata is, amelyben háromszögek helyett négyszögeket vizsgálnak, de tulajdonképpen bármilyen sokszögre általánosítható a módszer. Az 5. Ábrán néhány egyszerű példán keresztül részletezzük az *ECC* kiszámolását és tulajdonságait.

Az élklaszterezési együttható szerinti osztó módszer fő problémája, hogy bár gyors, ritka hálózatokon nem túl hatékony, mert azokban a hálózatokban eleve kevesebb háromszög található. Ennek közvetett hatása van a Voronoi cellák szerinti particionálásra is, mikor súlyként alkalmazzuk.

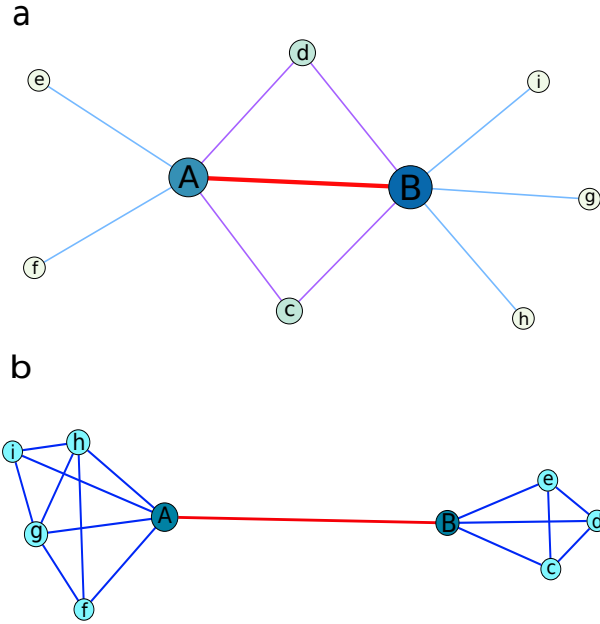
A Voronoi cellák kiszámításakor az *ECC* inverzét használjuk mint távolságmérték (súly), így egy él annál hosszabb lesz, minél inkább klaszterek közötti, így a Voronoi cellák hajlamosabbak lesznek a klasztereken belül kialakulni.

4.2 A relatív lokális sűrűség

A generátorpontok azonosításához egy szintén lineárisan skálázódó mérőszámot alkalmazunk. Ehhez először az úgynevezett relatív lokális sűrűséget (relative local density) kell kiszámolni, amely egy csomópontokhoz rendelhető érték. A mérőszám egy csomópont adott szintű szomszédsága által generált algráf relatív sűrűsége alapján számítható ki, a következő képlet szerint:

$$d_i = \frac{m}{m + k} \quad (2)$$

ahol d_i az i csomópont relatív sűrűsége, m a szomszédság által generált algráfon belüli élek száma, illetve k az algráfból kifelé mutató élek száma. A szomszédság méretét, vagyis



Ábra 5: **a)** A piros él (A és B csomópont között) élklasterező együttthatóját a következőképpen számoljuk ki: a háromszögek száma amelyekbe beletartozik (lila élek, c és d csomópontokkal) plusz egy, osztva a potenciális háromszögek számával amelyekbe beletartozhatna. Ez a kisebb fokszámú adjacens csomópont, jelen esetben az A, fokszáma mínusz 1 (a vizsgált él). Összes lehetséges háromszögek alatt azt az esetet értjük ha az A csomópont összes szomszédja (e, f) közös szomszéd lenne a B csomóponttal, mint a c és a d . Így a vizsgált él ECC-je $\frac{2+1}{4} = \frac{3}{4}$. Hasonlóan kiszámolva a **b)** ábrán pirossal kiemelt él együttthatóját $\frac{1}{3}$ -ot kapunk. Ez az egyszerű példa is szemlélteti, hogy a klaszterek közötti élek ECC-je jóval kisebb mint a klasztereken belül lévő éleké. Mikor távolságmértékként használjuk a Voronoi cellák megállapításakor, akkor arra van szükség, hogy a klaszterek közötti élek hosszabbak legyenek és ezért használjuk az együtttható inverzét.

azt, hogy hányad rendű szomszédokig megyek el, az algráf kialakításakor az l paraméter adja meg, amelyet legtöbbször $l = 1$ -es értékre rögzítettünk, azaz csak az első rendű szomszédok szerinti algráf relatív sűrűségét számoljuk ki.

A módszerünk azokat a csomópontokat jelöli meg generátorpontoknak, amelyeknek a relatív sűrűsége lokálisan maximális, vagyis nagyobb a körülötte r sugarú tartományon

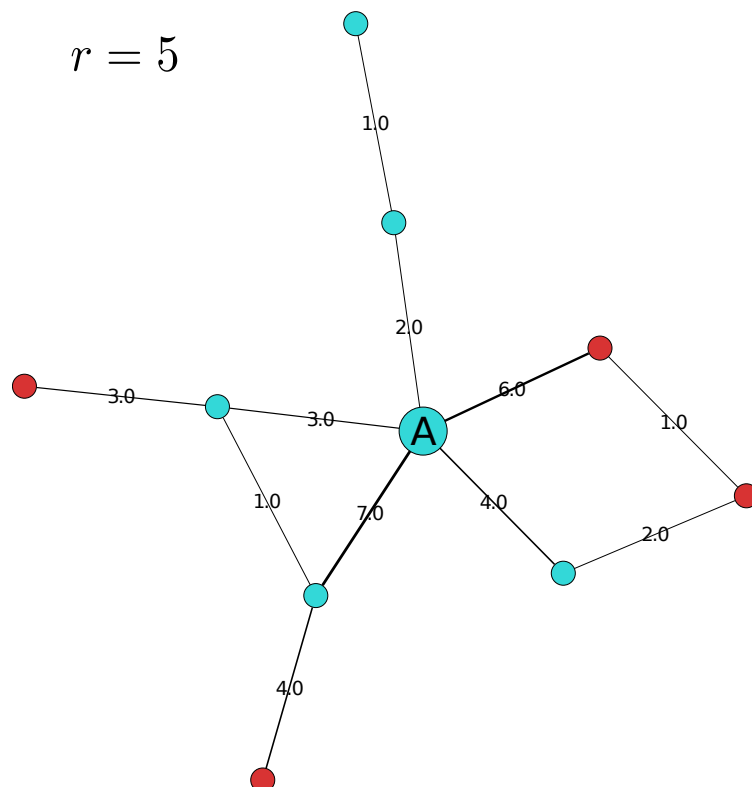
belül található összes csomópont relatív sűrűségénél. Ez az r hatótávolság, vagy sugár egy nagyon fontos paramétere a jelenlegi módszerünknek, a következőkben alaposabban ismertetjük a tulajdonságait.

4.3 Az r paraméter

A előző alfejezetben bevezetett r paraméter egy kulcsfontosságú paramétere jelenlegi munkánknak. A könnyebb érthetőség kedvéért a 6. Ábrán szemléltetjük, hogy hogyan működik tulajdonképpen a generátorpontok azonosítása.

Az r változtatásával azonosíthatjuk az optimális közösségi struktúrát és jó esetben a hierarchikus szerkezetet is. Intuitív, hogy az r növelésével csökkeni fog a generátorpontok száma, ugyanis egyre több kisebb lokális maximum “olvad össze” a nagyobb lokális maximumokkal. A generátorpontok számának csökkenésével pedig a partíciók száma is csökken és a megfelelő sugár eltalálásával maximalizálhatjuk a klaszterezés minőségét, sőt akár a hierarchikus szerkezet is megnyilvánulhat.

A Voronoi cellák alapján történő particionáláshoz szükséges egy távolságmérték definiálása és a generátorpontok azonosítása. A módszer nagyon rugalmas, mindkettőre többféle megoldást alkalmazhatunk. A távolságmérték esetében a legrövidebb út maradhatna egyszerűen a súlyozatlan hálók esetében alkalmazott definíció (az élek száma a legrövidebb út mentén), ellenben sokat növeli a particionálás pontosságát ha valamilyen mérték alapján minden élhez egy hosszúságot rendelünk. Mi itt egy viszonylag egyszerű mértéket, az élklaszterezési együtthatót használtuk, de bármikor találhatunk jobb módszert is. A generátorpontok hatékony azonosítása particionálás szempontjából még inkább fontos, de a módszer erre sem annyira érzékeny és az általunk megadott módszer csak egy első alternatíva, amelyet lehet tovább javítani.



Ábra 6: A középen A-val jelölt csomópont környezetét vizsgáltuk, úgy, hogy azokat a pontokat kerestük, amelyek a súlyozott gráfon r távolságon belül vannak, úgy hogy távolságként két pont közötti legrövidebb utat definiáltuk. A feltüntetett súlyokat figyelembe véve a kézzel jelölt csomópontok tartoznak az $r = 5$ sugáron belüli tartományba. Akkor lesz egy csomópont generátorpont, ha az r sugarú tartományon belül (jelen esetben kék) csomópontokhoz viszonyítva a 4.2 egyenletbenben definiált lokális sűrűsége az összes többinél nagyobb, azaz lokális maximumnak tekinthető.

5 Ellenőrzési módszerek

5.1 Benchmark hálózatok

Egyik elsőrendű célunk a módszer hatékonyságának tesztelése volt különböző méretű és típusú hálózatokon. Erre a Lancichinetti, Fortunato, és Radicchi által létrehozott “benchmark” hálózat generáló szoftver csomagot használtuk. [15] Egy fontos probléma

a közösség detektáló algoritmusoknál a különböző módszerek pontosságának a mérése, és ezek egymással való összehasonlítása. A benchmark csomag által képesek vagyunk irányított, irányítatlan és súlyozott hálózatokat generálni, hierarchikus, elkülönülő vagy átfedő közösségi struktúrával. A program rugalmas, és paraméterek széles skálája szerint tudunk hálózatokat generálni, ilyenek az N - a csomópontok száma, $\langle k \rangle$ - az átlag fokszám, k_{max} - a maximális fokszám, γ - a fokszámeloszlás hatványkitevője, β - a közösségek méreteloszlásának a kitevője, μ - a “keverési” (mixing) paraméter (megadja az inter és intra élek arányát), s_{max} - közösségek maximális mérete s_{min} - közösségek minimális mérete.

A benchmark kimeneti állománya egy éllista, 'forrás cél' formátumban, egy állomány, amely a csomópontok listáját tartalmazza és hogy melyik közösségbe tartoznak (a pontosság teszteléséhez) és egy adatfájl néhány statisztikai adattal a generált hálózatról. A szoftvercsomag tartalmaz egy másik programot is, az úgynevezett információátfedést mérő (*mutual information*) programot, amely elvégzi az összehasonlítást két kiszámolt közösségi struktúra között (adott esetben a benchmark által megadott “jó” közösségi struktúra és a tesztelt algoritmus által kapott klaszterezés között). Egy 0 és 1 közötti értéket térít vissza, 1 lévén a tökéletes egyezés, és 0 a teljes eltérés. [16] A hálózatok ábrázolására a *Gephi* [14] nevű programot használjuk, amely képes nagy hálózatokat is hatékonyan ábrázolni, akár három dimenzióban is és rengeteg kezelési lehetőséget biztosít.

5.2 A modularitás függvény

A benchmark hálózatok mellett egy hálózat klaszterezésének minőségét vagy “helyességét” az úgynevezett minőségfüggvényekkel lehet ellenőrizni. Azoknak a hálózatoknak az esetében, ahol nem áll rendelkezésünkre külső adat a valós közösségekről, az ilyen minőségfüggvények használata szükségszerű. Egyik legnépszerűbb ilyen minőségfüggvény a 2004-ben Newman által bevezetett modularitás függvény [6]. Feltételezván, hogy egy véletlenszerű gráf nem rendelkezik közösségi struktúrával, összehasonlítva egy adott klaszterezés algráfjainak (partícióinak) élsűrűségét a vizsgált gráf véletlenszerű kötésekkal rendelkező változatának

(null modell) várt élsűrűségével megkaphatjuk a klaszterezés minőségét. Ez a várt élsűrűség függ a null modell megválasztásától, vagyis az aktuális gráf másolatától, ahol megtartunk néhány strukturális tulajdonságot de közösségi struktúra nélkül. A modularitás függvényt a következőképpen írhatjuk fel:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - P_{ij}) \delta(C_i, C_j) \quad (3)$$

ahol m a gráf éleinek a száma, A az adjacencia mátrix, P_{ij} a várt élszám i és j csomópont között a null modellben, és δ függvény egy Kronecker delta szerű függvény, amely egyet térít vissza, ha i és j ugyanabba a megadott klaszterbe tartoznak és nullát ha nem.

P_{ij} alakját a választott null modell határozza meg, ennek több változata lehet. Például megkövetelhetjük, hogy a null modell tartsa meg az összes élet az eredeti gráfnak, és helyezze el őket bármely két csomópont között azonos p valószínűséggel (Bernoulli féle véletlenszerű gráf). Azonban ez a null modell mivel Poisson eloszlás szerinti foksámeloszlással rendelkezik sajnos nem írja le túl jól a valós hálózatokat, így preferálandó egy olyan null modell amely megtartja az eredeti gráf foksámeloszlását is. Ebben az esetben annak a valószínűsége, hogy i és j csomópont kapcsolódik nem más mint $k_i k_j / 2m^2$, amelyből következik, a $P_{ij} = k_i k_j / 2m$ várható élszám érték. Innen a végső formája a modularitás függvénynek:

$$Q = \frac{1}{2m} \sum_{ij} (A_{ij} - \frac{k_i k_j}{2m}) \delta(C_i, C_j) \quad (4)$$

Ez a modularitás függvény legelfogadottabb formája és méréseink során is ezt a képletet használtuk.

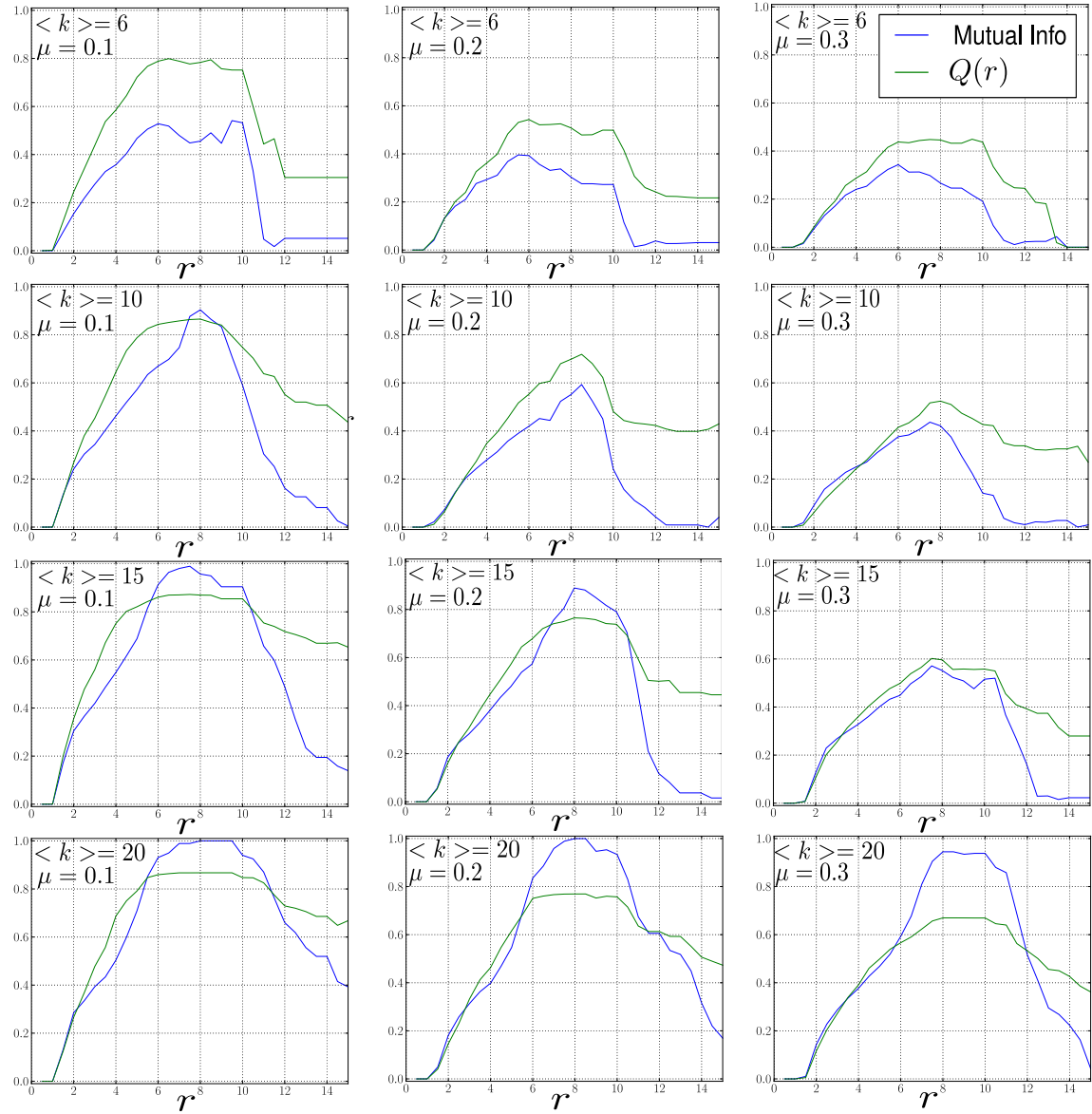
6 Az eredmények tárgyalása

Erdeményeinket kétféleképpen teszteltük: a benchmark, azaz automatikusan, paraméterek szerint generált hálózatokon, ahol eleve ismertük a közösségi struktúrát, illetve valós hálózatokon, ahol eredményeinket más módszerek eredményeivel hasonlítottuk össze. Kutatásunk során öt algoritmust választottunk ki, amelyek jó hatékonyság/komplexitás aránnyal rendelkeznek. A *GANXiS* [19], a Vincent D. Blondel és Jean-Loup Guillaume által bevezetett *Louvain* [20], két véletlenszerű mozgáson (random walk) alapuló algoritmus, a hálózat mérete szerinti lineáris skálázódással, a *Link Communities* [17] pedig egy félig determinisztikus tulajdonképpen éleket klaszterező módszer. Továbbá másik két stohasztikus módszert: *infomap* [22] és *label propagation* [21] is felhasználtuk bizonyos esetekben az összehasonlításra.

6.1 Benchmark hálózatokon való tesztelés

A módszerünket teszteltük a 5.1 alfejezetben tárgyalt benchmark hálózatokon. Ezeket különböző, számunkra releváns paraméterek szerint generáltuk.

A 7. Ábrán a különböző paraméterekkel generált hálózatokra kapott eredményeinket mutatjuk. Azt találtuk, hogy a Voronoi cellákon alapuló particionálás csak a k átlag foksám és a μ keverési paraméterre érzékeny, az összes többi paraméter változtatása nem zavarja a módszer hatékonyságát. A megfelelően magas k átlag foksám esetén az összes többi paramétertől függetlenül a particionálás jónak bizonyult. Ezért elsősorban a két kritikus paraméter szempontjából tanulmányozzuk tovább a viselkedést. A többi rögzített paraméter: $N = 10000$, $maxk = 200$, $\gamma = -2$, $\beta = -1$, $minc = 50$, $maxc = 500$. Ezek a rögzített paraméterek nagyjából megfelelnek a valós hálózatok paramétereinek. Két mérőszámot vizsgáltunk, a 5.2 alfejezetben már bevezetett modularitás függvényt és a benchmark hálózatok ismert közösségi struktúráját a programunk klaszterezésével összehasonlító információátfedési mértékét (mutual information). Ezt a két mérőszámot vizsgáltuk az r paraméter függvényében. A 1. táblázatban feltüntettük a legmagasabb



Ábra 7: A különböző benchmark hálózatokra elvégzett particionálások. A kék görbék a $[0,1]$ intervallumban mozgó információátfedési mérőszámot jelölik az r paraméter függvényében, amely azt adja meg hogy a benchmark által megadott közösségi struktúrát mennyire találjuk el (ahol 1 a tökéletes egyezés és 0 a teljes eltérés). A zöld görbék a modularitás függvényt ábrázolják az r függvényében. Észrevehető, hogy a két görbe viselkedése között erős korreláció van.

Hálózat	GANXiS	link communities	Louvain	Voronoj	Benchmark (eredeti)
$k = 6, \mu = 1$	0.8766616414	0.8156419763	0.8766616414	0.8013974096	0.8766385564
$k = 6, \mu = 2$	0.7660152086	0.5056297412	0.7725731877	0.542287516	0.772555445
$k = 6, \mu = 3$	0.6728873013	0.3646941413	0.6761960654	0.459360364	0.6765785679
$k = 10, \mu = 1$	0.8706721308	0.8566960202	0.8706721308	0.8658929951	0.8706721308
$k = 10, \mu = 2$	0.7664388552	0.7174722034	0.7600170176	0.7062928832	0.7664388552
$k = 10, \mu = 3$	0.6700173602	0.5744939643	0.6700173602	0.5359841829	0.6700173602
$k = 15, \mu = 1$	0.873727278	0.8559066993	0.873727278	0.872543222	0.873727278
$k = 15, \mu = 2$	0.7710062171	0.7283057282	0.7710232113	0.7637180402	0.7710062171
$k = 15, \mu = 3$	0.6735912086	0.5429961816	0.675368437	0.6093561324	0.675368437
$k = 20, \mu = 1$	0.8672872127	0.8508388219	0.8672872127	0.8672872127	0.8672872127
$k = 20, \mu = 2$	0.7694293396	0.7305448375	0.7694293396	0.7693499493	0.7694293396
$k = 20, \mu = 3$	0.6753304197	0.5981891287	0.6753304197	0.6707843432	0.6753304197

Táblázat 1: A teszhálózatokra kapott maximális modularitás összehasonlítva három módszerrel, és a benchmark program által iniciálisan adott eredeti klaszterezés modularitása (amely szerint generálta a hálózatot). A Voronoj szerinti particionálás jó eredményeket ad, kivéve a ritkább hálózatoknál.

modularitást elért particionálásokat, összehasonlítva a korábban felsorolt módszerek eredményeivel.

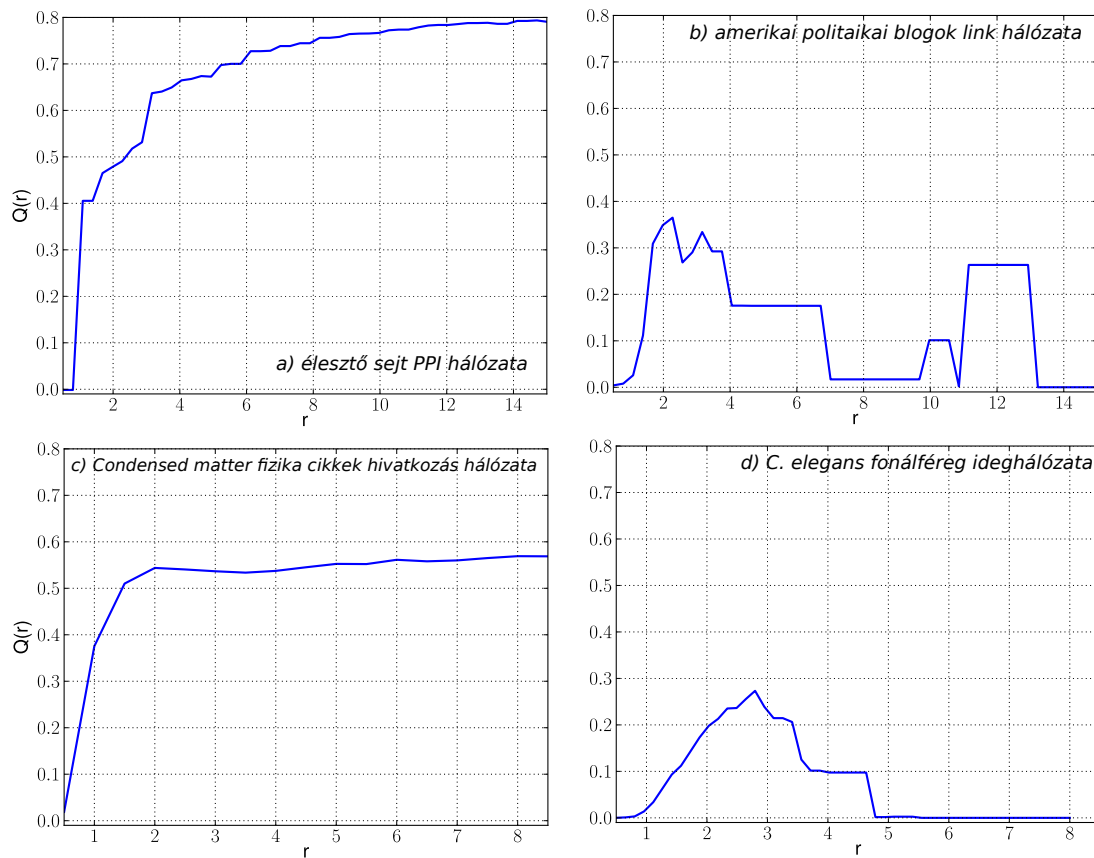
6.2 Valós hálózatokon való tesztelés

Ma már rengeteg alaposan feltérképezett valós hálózat áll rendelkezésünkre. Ezekre is alkalmaztuk a Voronoj cellákon alapuló particionálást. A minőségi ellenőrzést a 5.2 alfejezetben tárgyalt modularitás függvényrel végeztük, és eredményeinket összehasonlítottuk más, nagyon hatékonynak bizonyuló módszerek eredményeivel. A 8. Ábrán különböző valós hálózatok r paraméter szerinti elemzését tüntettük fel.

Különösen érdekes az *amerikai politikai blogok* (8. b Ábra) hálózata, amelynél egyértelműen látszik a hierarchikus szerkezet az r függvényében látható két csúcs. A két hierarchia szintet a 9. Ábrán szemléltetjük.

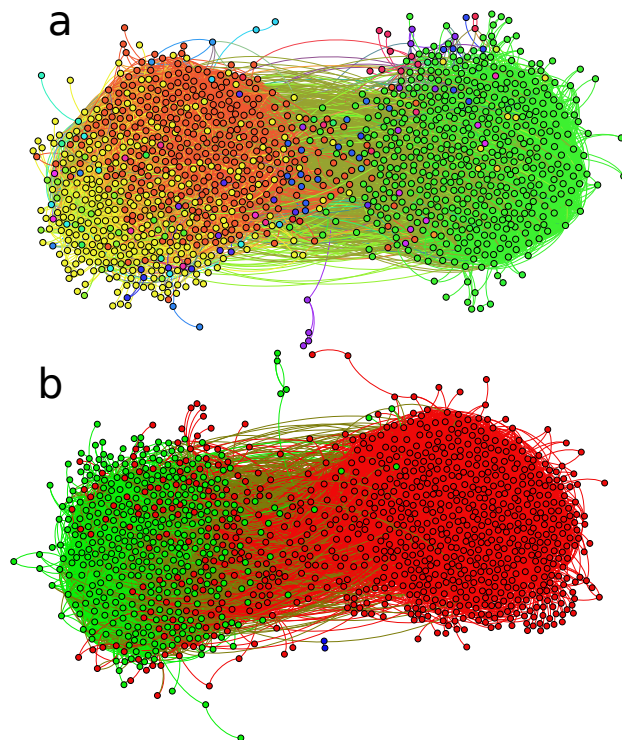
A 2. táblázatban az ellenőrzött valós hálózatokra kapott modularitás értékeket tüntettük fel a különböző módszerekkel.

A kapott eredményeket kielégítőnek tekinthetjük, és fontos megemlíteni, hogy a mod-



Ábra 8: A Q modularitás függvény az r paraméter függvényében különböző valós hálózatokon. Két féle viselkedést tapasztalunk: a modularitás elér egy (vagy több) lokális maximumot, majd lecsökken (b, d), vagy viszonylag hamar, már kicsi r értékeknél elér egy telítődést(a,c). Mindkét esetben az elért maximális modularitás jó egyezést mutat más módszerekével.

ularitás sok esetben viszonylag hamar, kicsi r értékek esetén telítődik (pl 8 a és c ábrák), így nem szükséges nagy r értékig elmenni egy jó pontosság elérése érdekében. Fontos kiemelni a 10. Ábrán látható grafikus összehasonlítást. Míg mindegyik módszer legalább egy hálónál nagyon rosszul szerepel, addig a mi módszerünk stabil, jellemzően átlag feletti teljesítményt mutat (lásd 10 Ábra (B)). Az egyedüli módszer amely mindegyik háló esetében jobban teljesít, a Louvain módszer, de ez a módszer eleve modularitást optimalizál, a függvényt, amelyet jelen esetben is alkalmazunk az összehasonlításra.



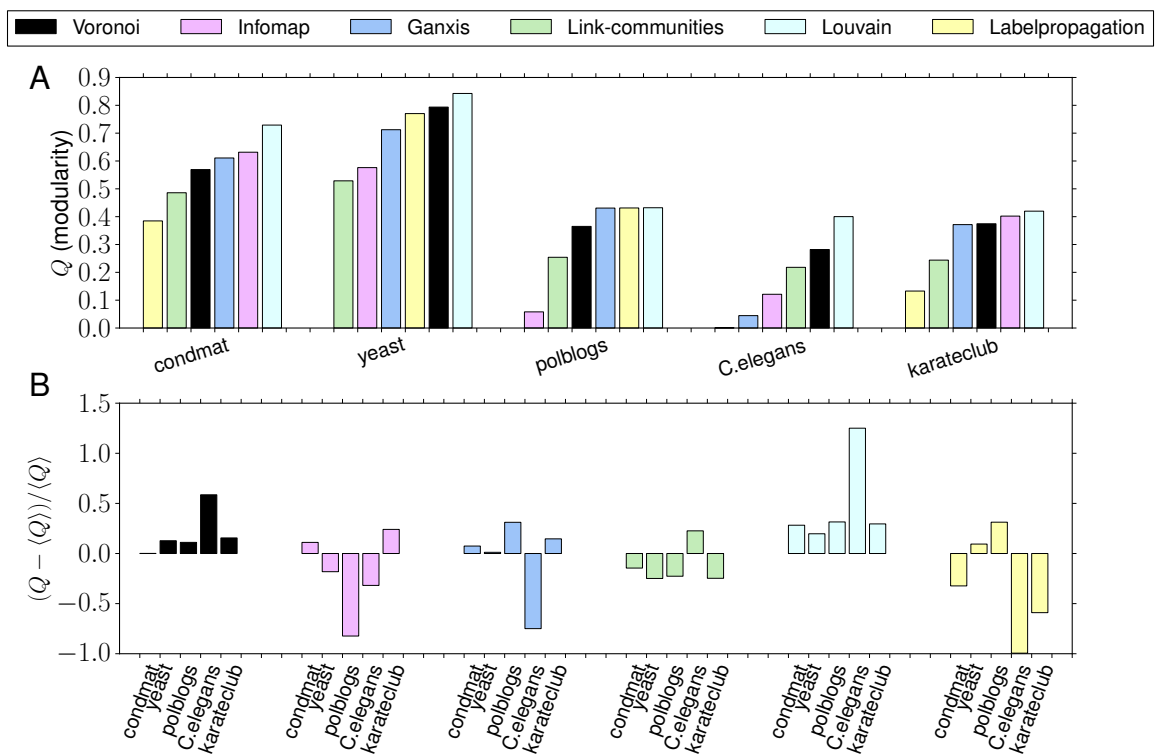
Ábra 9: A 8. **b** ábrán látható két lokális maximum a $Q(r)$ görbén két hierarchia szintnek felel meg. Ezt a két szintet ábráztuk, a partíciók szerint megszínezve a hálózatot az **a** (kisebbik r -nek megfelelő lokális maximum) és **b** (nagyobbik r -nek megfelelő lokális maximum) ábrán. A **b** változaton egyértelműen elkülönül a republikánus és a demokrata nézetű blogok, de ha lejjebb megyünk a hierarchiában (**a**) akkor a hálózat több kisebb közösségre szakad.

6.3 Skálázódás

Amint azt már a közösség detektáló algoritmusok tárgyalásakor kifejtettük, fontos, hogy egy algoritmus jól skálázódjon a rendszer méretével. A különböző energifüggvényekkel meg modularitás optimalizációval dolgozó módszerek ugyan jó eredményeket produkálnak, viszont általában $O(N^2)$ vagy még magasabb hatványú skálázódást mutatnak. A jelenleg lineáris és jó pontosságot elérő algoritmusok, mint amelyekkel az összehasonlításokat

Hálózat	Voronoj	GANXiS	Louvain	Link Communities
Condensed matter koll. háló	0.5691036792	0.610818373956	0.729021128092	0.2755691609
Zachary Karate Klub	0.3744247206	0.371466140697	0.419789612097	0.2168803419
C. Elegans fonálféreg	0.2819653286	0.3468256554	0.4001692938	0.1232699316
Politikai blogok	0.3650643136	0.4308840059	0.4320708228	0.1265607019
Élesztő sejt proteinhálój	0.7936063541	0.7123782429	0.8425942505	0.2844087283

Táblázat 2: Valós hálózatok klaszterezéseinek modularitás értékei. A táblázatban soronként a két legmagasabb értéket elért módszereket emeltük ki vastagított karakterekkel, és ebből aláhúztuk a nagyobbat. Látható, hogy átlagban a módszerünk jól szerepel.



Ábra 10: (A) A módszerünk legjobb modularitás értékei összehasonlítva öt népszerű módszerrel. A módszerek eredményei növekvő sorrendben helyezkednek el az adott hálózatokra. A Voronoi cellák szerinti klaszterezés átlagban a második legjobb eredményeket mutató módszer. (B) A módszerek relatív hibája, ahol $\langle Q \rangle$ a különböző hálókra számolt eredmények átlaga

végeztük nem determinisztikusak és sokszor más eredményeket adtak azonos futásokra. A gráf Voronoi cellák kiszámítására az irodalomban található egyik leggyorsabb módszer

az ún. *párhuzamos Dijkstra* algoritmust használja és legrosszabb esetben $O(m + n + (n - g)\log(n - g))$ szerint skálázódik [12], ahol n a csomópontok, m az élek, g pedig a generátorpontok száma. Ez egy közel lineáris algoritmusnak tekinthető.

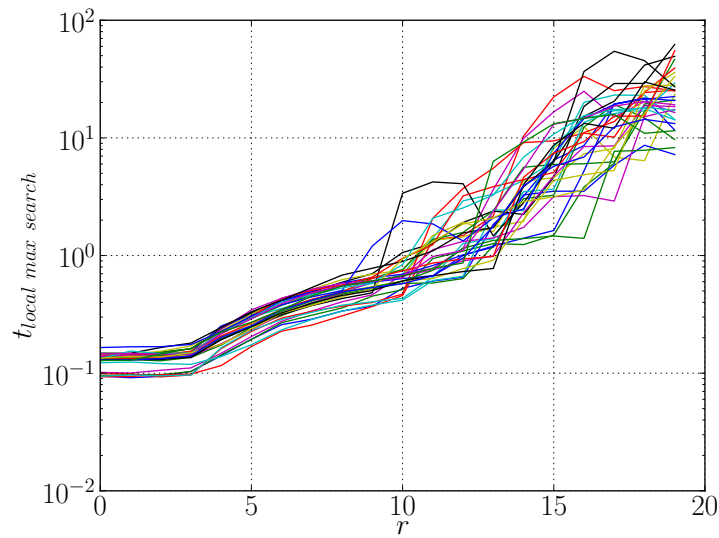
A jelenlegi probléma a lokális maximumok azonosítása. Ugyan a relatív lokális sűrűség és az élkaszterezési együttható kiszámítása is lineáris, a lokális maximumok azonosítása az r növelésével még mindig nagyon költséges. Ehhez egy Breadth-first search szerű algoritmust fejlesztettünk ki, amely figyelembe veszi az élek súlyait és beazonosítja egy csomópont környezetében r távolságon belül levő összes csomópontot, majd a sűrűségek összehasonlításával eldönti, hogy a csomópont lokális maximum-e avagy sem. A folyamatot még azzal is gyorsítottuk, hogy egyből a legnagyobb sűrűségű csomópontok ellenőrzésével kezdjük, és amint beazonosítottunk egy lokális maximumot, az összes többi csomópontot amely az r sugarú környezetében van, úgymond “kidobjuk”, azaz kivesszük az ellenőrizendő csomópontok listájáról. Az algoritmus azonban még így is nagyon költséges, közel exponenciális skálázódási tendenciát mutat az r paraméter növelésével (11. Ábra). Fontos megjegyezni, hogy mivel viszonylag kicsi r paraméter elegendő jó klaszterezés megtalálására, ez nem ront oskat az algoritmus hatékonyságán.

Jelenlegi méréseink elvégzéséhez ez a módszer még így is kiválóan szolgált, további gyorsításon és esetleges alternatívákon dolgozunk.

7 Következtetések

Mint az eredményeken láthattuk, a gráf Voronoi diagramok szerinti klaszterezés egy jól működő algoritmus és a módszer ebben a viszonylag egyszerű mértékeket használó formában is megállja a helyét a már elfogadott és népszerű algoritmusok között, néhány esetben túl is tesz rajtuk. A módszer legfontosabb előnyei:

- Egy elegáns és viszonylag precíz matematikai definícióként szolgálhat a közösségek



Ábra 11: Több ugyanolyan paraméterekkel generált benchmark hálózat relatív lokális sűrűség szerinti lokális maximum kereső algoritmusának az időbeni skálázódása az r függvényében.

meghatározására hálózatokban.

- Determinisztikusan kiszámítható, amely dinamikusán fejlődő hálózatoknál előny, lévén hogy egy véletlen bolyongáson alapuló algoritmussal szemben a végeredmény nem függ az indítási feltételektől.
- Rugalmas, mivel megfelelő esetleg hatékonyabb generátorpont azonosító módszerek és távolságmértékek bevezetésével tovább növelhető a pontossága.
- Robosztus abból a szempontból, hogy nem érzékeny arra, hogy egy klaszteren belül milyen generátor pontot választunk ki.
- Közel lineáris a skálázódás a hálózat méretének függvényében, amely az algoritmusok javításával tovább csökkenthető.
- Az r paramétert változtatva több skálán nyerünk információt, így hierarchikus struktúrát is fel lehet fedezni. Ugyanakkor látható, hogy az r növelésével a mod-

ularitás viszonylag hamar szaturálódik. Ez sokat javít a módszer gyorsaságán, mert azt jelzi, hogy viszonylag kicsi r paraméter elegendő egy jó klaszterezés megtalálásához.

- Megbízható eredmények úgy benchmark mint valós hálózatokon.

A módszer tovább javítható és jelenleg más távolságmértékek és generátorpont azonosító módszereket is tesztelünk. További ígéretes irányként mutatkozik egy stohasztikus módszer, amely a módszer robusztusságára támaszkodva véletlenszerűen választ ki generátorpontokat. Néhány iterációt követően a csomópont azonos közösségbe való tartozási valószínűségei egyértelmű információt adnak a közösségi és hierarchikus struktúráról és az esetleges átfedésekről.

7.1 Publikációk és konferenciák

A dolgozatban összefoglalt munkánkat a New Journal of Physics folyóiratban publikáltuk 2014-ben. A publikáció a <http://iopscience.iop.org/1367-2630/16/6/063007/article> linken ingyenesen elérhető és egy szemléletes videó-absztrakt is segít megérteni az alapkoncepciót. A cikk tartalmaz néhány új eredményt, szemléletesebb illusztrációkat, ábrákat, illetve koherensebb és érthetőbb szövegstruktúrával rendelkezik. Továbbá több jelen dolgozatban felmerülő kérdést és gyengepontot is kiküszöböltünk vagy magyaráztunk.

A módszert és eredményeinket Koppenhágában, a **NetSci 2013**-as konferencián is bemutattam a Hálózat Klaszterezés (Network Clustering) szekcióban. (Online programfüzet: http://netsci2013.net/wordpress/wp-content/uploads/2013/05/schedule_NetSci2013_detail-1.png)

Saját hozzájárulás: a módszer kiötlése és kifejlesztése, a szimulációk és a mérések implementálása és alkalmazása, ábrák és illusztrációk jelentős részének elkészítése, a cikk néhány technikaibb részének megírása.

Könyvészet

- [1] Fortunato, S., Phys. Rep. 486, 75-174 (2010)
- [2] Palla, G., I. Derenyi, I. Farkas, and T. Vicsek, 2005, Nature 435, 814.
- [3] Seidman, S. B., and B. L. Foster, 1978, J. Math. Sociol. 6, 139.
- [4] Seidman, S. B., 1983, Soc. Netw. 5, 269.
- [5] Radicchi, F., Castellano, C., CECConi, F., Loreto, V., Parisi, D., PNAS 101, 2658 (2004)
- [6] Newman, M. E. J., Girvan, M., Phys. Rev. E 69, 026113 (2004)
- [7] Burt, R. S., 1976, Soc. Forces 55, 93.
- [8] Wasserman, S., and K. Faust, 1994, Social network analysis (Cambridge University Press, Cambridge, UK).
- [9] Palmer, C. R., and C. Faloutsos, 2003, in Proceedings of PAKDD 2003, pp. 486–500.
- [10] Tong, H., C. Faloutsos, and J.-Y. Pan, 2008, Knowl. Inf. Syst. 14(3), 327.
- [11] Girvan, M., and M. E. J. Newman, 2002, Proc. Natl. Acad. Sci. USA 99(12), 7821.
- [12] M Erwig, F Hagen, Networks, 2000 - Citeseer
- [13] A. Okabe, T. Satoh, T. Furuta, A. Suzuki and K. Okano, International Journal of Geographical Information Science (2008)
- [14] Bastian M., Heymann S., Jacomy M., ICWSM (2009)
- [15] Lancichinetti, A., Fortunato, S. and Radicchi, F., Phys. Rev. E 78, 046110 (2008)
- [16] Lancichinetti, A., Fortunato, S., [arXiv:0904.3940](https://arxiv.org/abs/0904.3940) (2009)

- [17] Ahn, Y.-Y., Bagrow, J. P. and Lehmann S., *Nature* 466, 761 (2010)
- [18] Xie, J., Szymanski, B. and Liu, X., *IEEE ICDM workshop on DMCCI* (2011)
- [19] Xie, J. and Szymanski, B. K., *PAKDD* (2012)
- [20] Vincent D. Blondel, Jean-Loup Gouillaume, *Journal of Statistical Mechanics: Theory and Experiment* 2008 (10), P1000
- [21] Xie J and Szymanski B 2012 Towards linear time overlapping community detection in social networks *Advances in Knowledge Discovery and Data Mining* (Berlin: Springer) 25–36 (Lecture Notes in Computer Science vol 7302)
- [22] Rosvall M and Bergstrom C T 2008 Maps of random walks on complex networks reveal community structure *Proc. Natl Acad. Sci.* 105 1118–23