Elosztott frekvenciakiosztási algoritmusok tervezése és vizsgálata

BIRÓ PÉTER–LÓJA KRISZTINA–SÜTŐ MÁRTON

Budapesti Műszaki és Gazdaságtudományi Egyetem (BME) matematikushallgatók

Bevezetés

Egy általános tendencia a mobil, celluláris hálózatok fejlődésében a mind kisebb és kisebb méretű cellák alkalmazása. Minderre egyrészt az egyre nagyobb frekvenciák használata, másrészt az egyre nagyobb kapacitás iránti igények miatt van szükség. A GHz tartományban a rádióhullámok terjedése a fény terjedéséhez hasonlítható, ahol a rádióhullámok könnyen elnyelődnek a különféle tereptárgyakon és akadályokon. Ebben a sorban rendszerint közvetlen rálátás szükséges az adó és a vevő között a kapcsolat érdekében. Mindez a terepviszonyoktól függően kisebb és kisebb cellák alkalmazását teszi szükségessé, ugyanakkor a kapacitás növelésének is az egyik módja a kisebb cellák használata. Ilyenkor ugyanazon bázisállomás kapacitása egy kisebb területen belül áll rendelkezésre, amely lehetővé teszi nagyobb felhasználósűrűségű területek kiszolgálását.

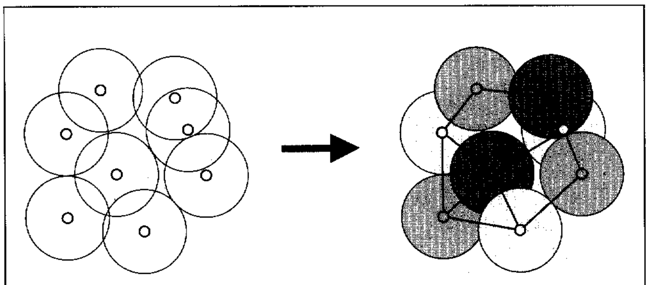
Az ilyen mikro- vagy pikócellás környezetek egy nagyváros sűrűn lakott vagy üzleti negyedeiben, illetve egy épületen belüli mobil rendszerként képzelhetők el. A konkrét környezettől és rendszertől függetlenül a cellák nagy száma miatt egy ilyen hálózat konfigurációja, cellatervezése, valamint a frekvenciaspektrum használatának koordinálása egyre nehézkesebbé válik. Továbbá ezen rendszerek egy része, tipikusan az épületen belüli rendszerek a felhasználó által is telepíthetők kell hogy legyenek (pl. mai Wireless LAN rendszerek), ahol nem várható el a felhasználótól, hogy részletes cella- és spektrumtervezést végezzen, vagy egyéb bonyolult konfigurációs műveleteket hajtson végre. Ehelyett megköveteljük a rendszertől, hogy autokonfiguráció útján a szükséges működési paramétereket saját maga automatikusan állítsa be.

Az egyik legfontosabb konfigurációs lépés a frekvenciák kiosztása úgy, hogy a szomszédos bázisállomások különböző frekvenciát használjanak és így a cellák közötti interferencia elkerülhető legyen. Ezt a jelenlegi cellás hálózatokban általában a hálózat telepítését megelőző, „off-line" módon végrehajtott frekvenciatervezéssel oldják meg.

Ezen cikk keretében az automatikus frekvenciaválasztás problémájával foglalkozunk, ahol a feladat a fenti, frekvenciakiosztási probléma megoldása elosztott, „on-line" módon. Olyan algoritmusok tervezésével és vizsgálatával foglalkozunk, amelyek az egyes bázisállomásokban működve képesek a frekvenciájukat úgy meghatározni, hogy a szomszédos bázisállomások által használt frekvenciák különbözők legyenek. Fontos különbség a hagyományos cellatervezéshez képest, hogy ezek az algoritmusok on-line módon és tipikusan lokális információt felhasználva működnek, míg egy off-line cellatervezést az egész hálózat figyelembevételével és esetleg időigényes, bonyolult algoritmusok használatával végzik.

1. Matematikai modell

A probléma matematikai modelljét az alább definiált, úgynevezett interferencia-gráf adja. A gráf pontjai felelnek meg az állomásoknak, két pont között pontosan akkor megy él a gráfban, ha a megfelelő adók egy adott kritikus távolságnál (d) közelebb vannak egymáshoz; $d \geq 2R$, ahol R a bázisállomások hatósugara.



1. ábra Az interferencia-gráf származtatása

Feladatunk a gráf pontjainak kiszínezése oly módon, hogy a szomszédos pontok mind különböző színűek legyenek. Továbbá a színezést végrehajtó algoritmus elosztott módon megvalósítható kell legyen úgy, hogy az állomások vagy kizárólag lokális információ felhasználásával, vagy a szomszédokkal folytatott minimális információcsere alapján választanak frekvenciát. Feltételezzük, hogy a bázisállomások képesek mérni az interferencia szintjét az összes létező frekvencián és ez alapján érzékelni, hogy a környezetükben mely frekvenciák szabadok és melyek nem. Ez adja az alap, lokális információt, amely minden esetben rendelkezésünkre áll. Ezek után egy érdekes kérdés annak vizsgálata, hogy a bázisállomások közötti további információcsere bevezetésével mennyiben javítható az egyes algoritmusok teljesítménye.

Az algoritmusok teljesítményének egyik legfontosabb mérőszáma a felhasznált színek száma, illetve a meghiúsult, úgynevezett „blokkolt" esetek aránya, amikor az algoritmus egy adott színszámkorlát mellett nem tudta a gráfot kiszínezni. Később az algoritmusok kiértékelésekor ezen mérőszámokat használjuk majd.

Az algoritmusok teljesítményét kétfajta gráf esetén vizsgáltuk. Egyrészt olyan gráfokon végeztük a számítást, melyeknek ismerjük a kromatikus számát, ekkor ugyanis pontosan tudjuk, hogy az algoritmus mennyit hibázott, azaz mennyivel több színt használt a minimálisan szükségesnél. Másrészt olyan gráfokat is vizsgáltunk, amelyek egy tipikus bázisállomás-elhelyezkedés esetén állhatnak elő. Ez utóbbi esetben két jellemző szempontot vettünk figyelembe. Az egyik az adott terület közel egyenletes lefedése, míg a másik az ad hoc jellegű telepítésből eredő véletlenszerűség.

2. A színezés elméleti korlátai

Egy gráf kiszínezhetőségének eldöntése ($k \geq 3$ konstans esetén) egyike az ismert NP-teljes problémáknak. De ugyanígy NP-nehéz probléma egy gráf kiszínezése akkor is, ha ismerjük a gráf kromatikus számát ($\chi(G)$-t), ahol χ és G, tehát $\chi(G)$ azt a legkisebb számot, ahány színnel létezik jó színezése a gráfnak. Mindez azt jelenti, hogy – bár a számítástudomány talán legnagyobb sejtése: $P \neq NP$ még nem nyert bizonyítást – nem tudunk adni olyan polinom idejű algoritmust, ami általánosságban mindig optimálisan oldja meg a feladatot. Sőt Johnson tétele [5] szerint általános esetben még multiplikatív hibán belüli színezhetőséget sem tudunk garantálni, ha ugyanis létezne olyan polinom idejű algoritmus, ami legfeljebb $c\chi(G)$ ($c>0$ konstans) színt használ minden G gráfra, akkor létezne polinom idejű algoritmus $\chi(G)$ meghatározására is.

Vannak ugyan olyan gráfosztályok, melyekre sikerült pozitív tételeket kimondani. A legismertebb ezek közül a síkba rajzolható gráfok osztálya, ahol az Appel és Haken által bizonyított híres 4-szín tétel szerint minden gráf kiszínezhető legfeljebb négy színnel. Érdekes számunkra Borodin tétele [5] is, mely szerint minden olyan gráf, amelyet le lehet úgy rajzolni a síkon, hogy minden élét legfeljebb csak egy másik metszhet, mindig 6-színezhető.

Sajnos az általunk vizsgált eset az állomások rendezetlen elhelyezkedése miatt ennél általánosabb, és így, korlátozó feltételek híján nem remélhető olyan polinom idejű algoritmus, amely mindig optimálisan oldaná meg a feladatot. Általános tapasztalat, hogy az exponenciális algoritmusok körülbelül 20 pontig még képesek kezelni az ilyen típusú feladatokat, de ezenfelül már rendkívül gyorsan növekszik a futásigényük. Mindezek okán célunk arra korlátozódott, hogy olyan polinomrendű algoritmusokat keressünk, melyek hipotetikusan jól közelítik az optimális megoldást, ennek megfelelően algoritmusaink futási ideje csupán másodpercekben mérhető.

3. Az algoritmusok bemutatása

Az egyes színező algoritmusokat három fő szempont szerint csoportosítottuk. Az első a gráf pontjainak bejárása, a második szempont a színválasztási stratégia, míg a harmadik az úgynevezett javító eljárás lehetősége.

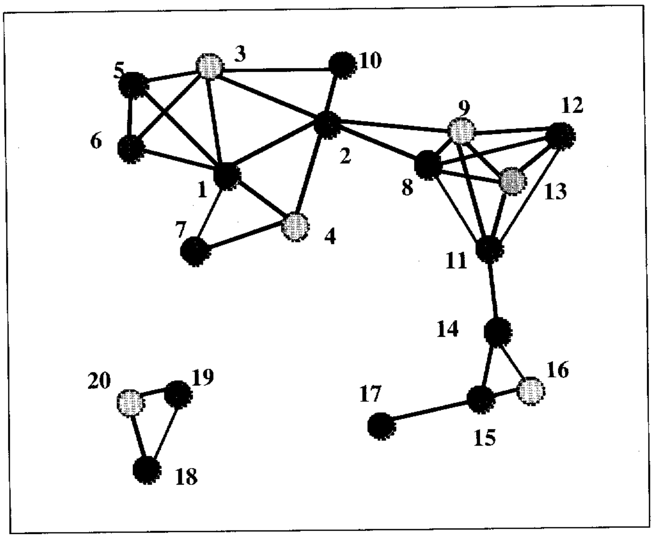
A szimulációk során megfigyeltük, hogy egy színező algoritmus hatékonysága azon múlik, hogy milyen sorrendben járja be a gráf pontjait. A színválasztás módja, illetve egy esetleges javító eljárás használata csak kisebb mértékben befolyásolják a hatékonyságot, de ennek ellenére szerepük nem elhanyagolható.

Bejárás: A véletlen bejárás mellett három alapbejárást vizsgáltunk, melyekhez szükséges a teljes gráf ismerete, majd ezt követően ezekből fejlesztettünk ki négy olyan bejárási módot, melyek a lokalitás feltételének is megfelelnek és így valóban megoldásai a kitűzött feladatnak.

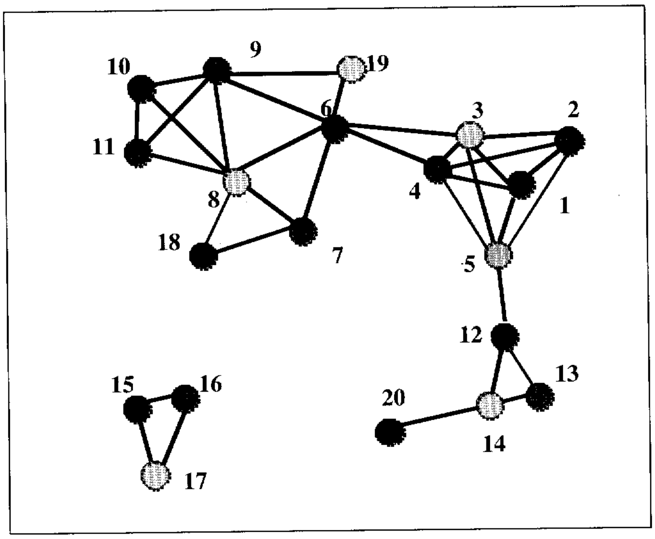
- *Véletlen bejárás:* A pontokat véletlen sorrendben járjuk be. Ez gyakorlatilag annak a megfelelője, hogy az állomásokat előzetes tervezés nélkül, véletlenszerűen kapcsoljuk be. Jelen cikk egyik témája pontosan annak megmutatása, hogy ez a gyakorta használt eljárás mekkora pazarlással jár a felhasznált színek számát tekintve.
- *Törlős bejárás:* A gráf pontjai közül sorban kitöröljük a legkisebb fokszámút, és közben visszafelé sorszámozzuk őket. Amelyik pont utoljára marad, azt fogja kiszínezni először az algoritmus. Az interferencia-gráfok esetén látni fogjuk, hogy ez az eljárás különösen hatékonynak bizonyul.
- *Fokszám szerinti bejárás:* Az algoritmus a fokszámok szerint csökkenő sorrendben számozza meg a pontokat. Először tehát a legnagyobb fokút, és utoljára a legkisebb fokút fogja kiszínezni.
- *Globális bejárás:* A pontok bejárásának sorrendjét az eddigiektől eltérően dinamikusan határozza meg, vagyis figyelembe veszi, hogy addig mely pontok és miként lettek kiszínezve. Mindig azt a pontot fogjuk kiszínezni előbb, melynek a szomszédjai már a legtöbb színt elhasználták, vagyis amelyiknek a legkevesebb a választási lehetősége. A pontok fokszáma csak másodsorban dönt.

Saját fejlesztésű bejárások

- *Szélességi bejárás:* Az előző eljárás szerint, de csak lokális környezetből (a szomszédok közül) választ új pontot, irányított szélességi bejárás során.
- *Mélységi bejárás:* Hasonlóképpen az előzőhöz, csak irányított mélységi bejárást használ.
- *Lokális fokszámos bejárás:* Minden olyan pontot, melyek fokszáma nagyobb a szomszédainál, sor-



2. ábra Szélességi bejárás



3. ábra Törlős bejárás

rendben megszámozzuk, majd a maradék pontokon ismételjük meg az előző eljárást, így kapjuk meg a pontok bejárásának sorrendjét.

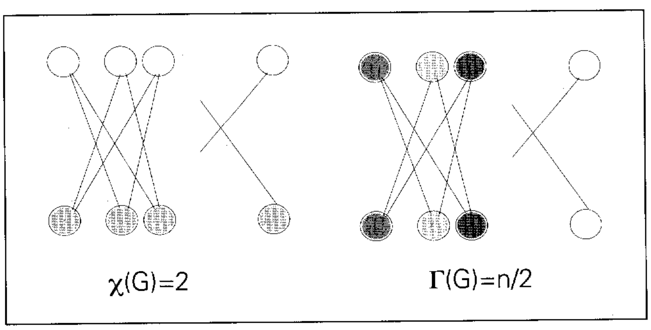
- *Lokális törlős bejárás:* Minden olyan pontot, melynek fokszáma kisebb a szomszédainál, visszafelé megszámozunk, majd töröljük a gráfból. A maradék gráfon folytatjuk az előző eljárást, amíg a pontok el nem fogynak.

A következő ábrákon a törlős és a szélességi bejárás szerinti színezésekre láthatunk egy példát. A pontokat jelző számok a bejárás sorrendjét, míg a színek a pont választott színét jelzik (2., 3. ábra).

Színválasztás: Tesztjeink során több színválasztást is kipróbáltunk, és végül két olyat tartottunk meg, melyek a leghatékonyabb bejárások szerint a legjobb eredményre vezettek.

- *Mohó színezés:* Egy pont színének (egy állomás frekvenciájának) meghatározására a legegyszerűbb, de mégis viszonylag hatékony, és ezért gyakran használt eljárás; melynek során a megsorszámozott színek közül az új pont mindig a legkisebb lehetséges, – a szomszédok által nem használt – színt választja.

Hátránya, hogy a bejárás módja nagymértékben befolyásolja az algoritmus működését. Legjobb esetben,



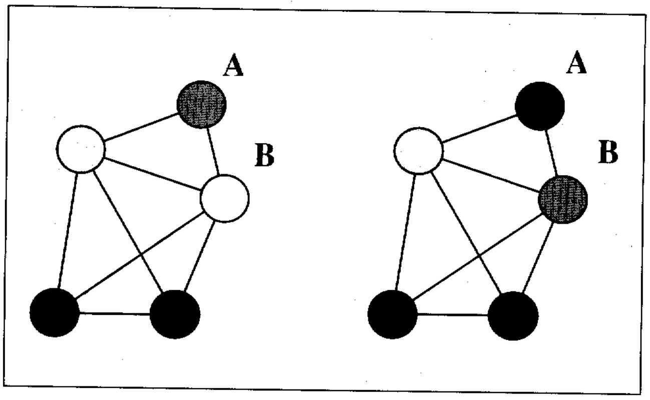
4. ábra Az optimális és legrosszabb színezés esete páros gráf esetén

ha például egy, a kromatikus számmal történő színezés színosztályai szerint megy végig a pontokon, a mohó színezés optimális lesz. A legrosszabb esetek vizsgálatának széles körű irodalma van. Azt a legnagyobb k számot, amelyre létezik a G gráfnak olyan mohó színezése, amely pontosan k színt használ, a G gráf Grundy-számának (G(G)) nevezzük Elméleti kitekintés helyett lássunk példaként egy speciális páros gráfot, ahol a kromatikus szám n/4-szeresét is elérheti a felhasznált színek száma (n a gráf pontjainak száma), abban az esetben ha a legszerencsétlenebb sorrendben járjuk be a gráf pontjait.

Ez az arány Lovász Saks, Trotter [6] tétele alapján maximális, vagyis ennél szélsőségesebb példa már nem képzelhető el. Christen és Selkow [3] több további példát is mutattak a G(G)/c(G) arányának maximumára és más gráfszínezési paraméterek szélső értékeire.

- Okos színezés: A lehetséges színek közül azt választja, amelyet a pont szomszédainak a szomszédai a legtöbben használnak, másodsorban pedig a legkisebb sorszámút. Az elgondolás lényege, hogy megpróbáljuk minél többször újrahasználni az addig felhasznált színeket. Az okos színezés technikai feltétele, hogy az állomásoknak tudniuk kell kommunikálni egymással.

Javítás: Abban az esetben, amikor az aktuális pontnak új színt kellene választania, tehát az algoritmus által addig használt színek mindegyike megtalálható a pont szomszédai közt, akkor egy visszalépéses eljárás megvizsgálja, hogy lehet-e javítani. Javítani akkor tudunk, ha a szomszédok között van egy olyan színosztály, amelyben minden pont át tud térni egy lehetséges másik színre. Ha ez fennáll, akkor a pontunk megkaphatja a színváltó színosztály régi színét, annak pontjai pedig mindannyian egy másik, már felhasznált színt választanak; így nem kell új színt használnunk. Az 5. ábrán látható példában a B pontnak nem választható szín úgy, hogy az ne ütközzön már korábban beszínezett szomszédos pontok színeivel. (Feltételezzük, hogy csak az ábrán látható négy szín áll rendelkezésünkre) Ekkor a javító eljárás szabályait követve az A pont áttér egy másik, a szomszédai által nem használt színre, mellyel lehetővé válik a B pont kiszínezése is.



5. ábra Példa a javító eljárás működésére

4. Eredmények a vizsgált gráfosztályok szerint

A fenti három szempont szerint az algoritmusok összes kombinációját elkészítettük (2×8×2=32 db algoritmus), majd különböző gráfosztályokból vett nagyszámú mintán futtattuk, és az átlagos színhasználatuk alapján rangsoroltuk őket, majd ezek alapján vontunk le következtetéseket a hatékonyságukra nézve. Először egyszerű véletlen gráfokat, majd ismert kromatikus számú véletlen gráfokat használtunk. Ezután a valóságot jobban modellező interferencia-gráfokat generáltunk, és ezeken vizsgáltuk az algoritmusok működését.

Véletlen gráfok:

Véletlen gráf: Adott pontszám és élbehúzási valószínűség megadásával generálódik.

Ismert kromatikus számú véletlen gráf: A gráf kromatikus számát előre beállíthatjuk.

Ezen gráfok generálása úgy történik, hogy a véletlen gráf pontjait adott számú (k) színosztályra bontjuk, a színosztályok közti éleket kitöröljük, majd hozzáadunk egy teljes k-s részgráfot, amellyel biztosítjuk, hogy a kromatikus szám pontosan k legyen.

Az eredmények mindkét gráfosztály esetén hasonlók, de az algoritmusok közti különbségek ismert kromatikus gráfok esetén számottevőbbek. A tesztek közül lássunk egy kiragadott és jellemző példát: A 32 algoritmus 1000 db ismert (5) kromatikus számú gráfon lefuttatva, amint a pontok száma 20 és 29 között egyenletesen, 100-as mintánként növekedett. A táblázatban szereplő értékek megadják, hogy hány esetben sikerült az adott algoritmusnak optimálisan kiszíneznie a gráfot, továbbá szerepel az is, hogy hány esetben használták a futás során az algoritmusok a javító eljárást:

Optimális színezések száma	Véletlen		Törlős		Fokszám		Globális	
	mohó	okos	mohó	okos	mohó	okos	mohó	okos
Első futtatásra	230	292	709	773	621	691	899	910
Javítós	663	663	878	889	827	903	924	927
Jav. száma	1921	1645	311	234	704	590	46	38

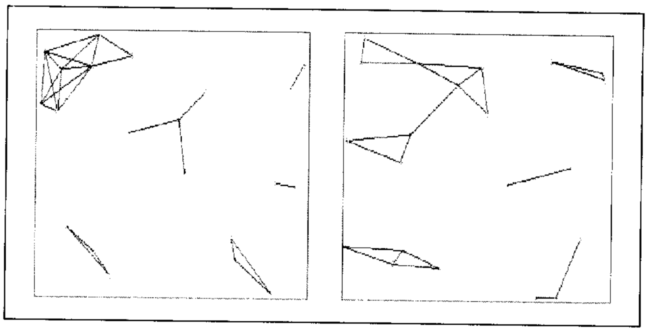
Optimális színezések száma	Szélességi		Mélységi		Foklokál		Törlokál	
	mohó	okos	mohó	okos	mohó	okos	mohó	okos
Első futtatásra	497	780	787	849	621	667	542	663
Javítós	818	856	873	910	804	815	764	829
Jav. száma	543	170	156	107	702	616	698	526

1. táblázat Teszteredmények egy ismert kromatikus számú gráfosztályon

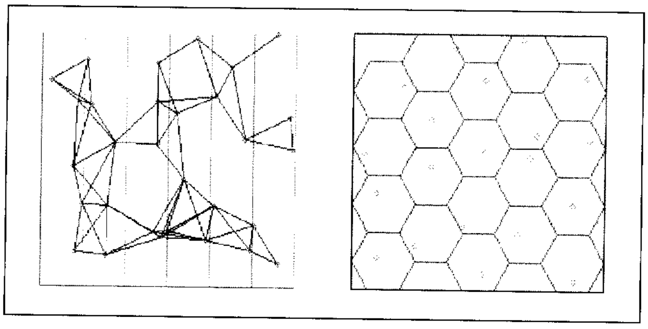
A legjobb algoritmusnak a teljes gráf ismeretében a „Globális okos javítós" változat bizonyult, míg lokális esetben a „Mélységi okos javítós" algoritmus. Az okos színezés minden esetben jobb, mint a mohó; a javító eljárás szerepe főként a kevésbé hatékony bejárások esetén nagy.

Interferencia-gráfok

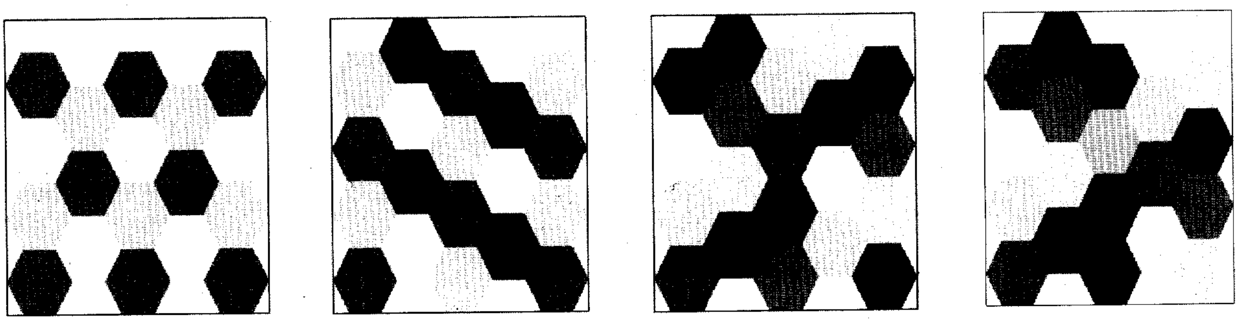
A valóságot jól modellező interferencia-gráfok korábban említett két legfontosabb tulajdonsága a viszonylag jó síklefedés és a véletlenszerűség. Az általunk generált öt osztály is e szempontok szerint halad a teljesen rendezetlen állapottól egészen az ideális síklefedést biztosító háromszögrács esetéig.



6. ábra Egyenletes és pontelhagyással módosított eloszlás



7. ábra Négyzetrács- és hatszögcellagráf



1) $2c \le d < 3C$ 2) $3C \le d < 4c$ 3) $4c \le d < K$ 4) $K \le d < 6c$

(c= a hatszög magasságának fele, C= a hatszög oldalának hossza, $K = \sqrt{(4.5C)^2 + c^2}$)

8. ábra A hatszögrács optimális színezése a kritikus távolság (d) változása mellett

a) *Egyenletes eloszlás* esetén: A pontokat egyenletes eloszlás szerint helyezzük el a síkon.

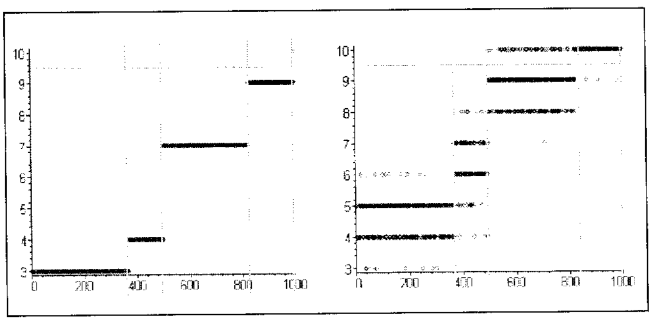
b) *Pontelhagyással módosított* eloszlás: Az egyenletesebb síklefedés érdekében, ha egy pont túl közel kerülne egy másikhoz, csak egy bizonyos valószínűséggel helyezzük el ott, amely valószínűség a már lerakott pontoktól távolodva egyenletesen nő.

c) *Négyzetrácsgráf:* Ebben a modellben egy négyzetrács celláiba teszünk egy-egy pontot egyenletes valószínűséggel. A síklefedés egyenletesebb, de még jelen van a rendezetlenség.

d) *Hatszögcellagráf:* Az előzőhöz hasonlóan, most hatszögcellákba helyezzük el véletlenül, egyenletes eloszlás szerint a pontokat (6., 7. ábra).

e) *Háromszögrács:* A bázisállomások a hatszögcellák közepébe kerülnek, vagyis egy szabályos háromszögrácsot alkotnak. Ez az utolsó eset abból a szempontból különösen érdekes számunkra, hogy ezen gráfosztályon a szükséges színek száma csak a hatszögek oldalhosszának és a kritikus távolságnak az arányától függ.

A törlős bejárást használó algoritmus legfeljebb 9 színt használt a színezéshez mind az 1000 gráf esetén (az ábrán vékony folytonos vonal a 10-es érték alatt), a véletlen bejárást használó algoritmus azonban sok esetben túllépte ezt a korlátot. A törlős bejárás esetén minden gráf az adott intervallumnak megfelelő optimális színszámmal színezhető, véletlen bejárás esetén a



7. ábra A háromszög-rács színezésekor felhasznált színek száma Törlős és Véletlen bejárás esetén

felhasznált színek számában jelentős szórás figyelhető meg, ezt mutatják az egymás felett elhelyezkedő ponthalmazok, melyek némely esetben párhuzamos vonalként jelentkeznek.

A többi vizsgált interferencia-gráf esetén az eredmények az osztályoktól függetlenül mind hasonlóak lettek. Lássuk példaként egy négyzetrács gráfosztályon elvégzett tesztet, amelyben szintén 1000 gráfon futtattuk le mind a 32 algoritmust. A négyzetrácsgráf 6x5-ös, tehát minden esetben 30 pontból áll, a kritikus távolság pedig fut 1-től 2-ig, ahol 1 a négyzetcella oldalának hossza. A táblázatban a felhasznált színek átlagain túl szerepel az is, hogy az egyes algoritmusok összesen hányszor használták a javító eljárást.

Felhasznált színek átlaga	Véletlen		Törlős		Fokszám		Globális	
	mohó	okos	mohó	okos	mohó	okos	mohó	okos
Első futtatásra	5.368	5.252	4.680	4.677	4.867	4.789	4.712	4.709
Javítós	4.907	4.866	4.674	4.673	4.723	4.714	4.692	4.691
Jav. száma	1282	1007	12	9	385	241	32	28

Felhasznált színek átlaga	Szélességi		Mélységi		Foklokál		Törlokál	
	mohó	okos	mohó	okos	mohó	okos	mohó	okos
Első futtatásra	4.739	4.719	4.776	4.750	4.867	4.819	4.879	4.828
Javítós	4.700	4.696	4.716	4.707	4.744	4.734	4.770	4.752
Jav. száma	57	35	100	66	310	226	230	166

2. táblázat A háromszögrács színezésekor felhasznált színek száma Törlős (a) és Véletlen (b) bejárás esetén, ahol a vízszintes tengely pontjainak egy-egy lehetséges gráf felel meg (összesen 1000 db), amelyek mindegyikéhez hozzárendeljük a megfelelő (véletlen [b] vagy törlős [a]) bejárású algoritmus által a kiszínezésükhöz szükséges színek számát, miközben a kritikus távolság egyenletesen nő. Itt a függőleges tengelyre kellene írni, hogy a szükséges színek száma, a vízszintes tengely alá pedig: a gráfok száma

Bár a bonyolultabb algoritmusok közti különbségek a fenti interferencia-gráfok esetén már nem olyan látványosak, a legegyszerűbb, véletlen bejárást használó javítás nélküli esethez képest továbbra is javulás érhető el. A teljes gráf ismeretében működő algoritmusok között a „Törlős okos javítós" bizonyult a legjobbnak, de itt már mind a színválasztás, mind a javítás szerepe elhanyagolható. A lokális, tehát az eredeti feltevéseinknek megfelelő algoritmusok között a „Szélességi okos javítós" bizonyult a leghatékonyabbnak. A színválasztás és a javítás szerepe érzékelhető.

Felvetődik a kérdés, hogy mi lehet az oka annak, hogy a Törlős algoritmus interferencia-gráfok esetén a legjobban közelíti meg az optimumot. A hipotetikus magyarázat erre a kérdésre, hogy a Törlős algoritmus során először az interferencia-gráf szélén lévő kis fokszámú pontokat hagyjuk el, és fokozatosan megyünk a síkrész belseje felé, míg a legvégén a színezés szempontjából kritikus csomók, klikkek maradnak. A színezés során tehát először beszínezzük a kritikus pontokat, a klikkeket és a nagyfokú pontokat, majd a végén a szélső pontokat már egyszerűen, a javító eljárás használata nélkül tudjuk kiszínezni. Sajnos ennek lokalizált változata már kevésbé hatékony, és ebben az esetben a bonyolultabb és az állomások között több kommunikációt igénylő irányított szélességi bejárás a jobb.

Összefoglalás

Munkánk során tehát sikerült olyan algoritmusokat alkotnunk, melyek az adott feltételek mellett a valóságot jól közelítő modelleken egyszerűen, gyorsan és az optimálist vélhetően jól közelítő eredményt adnak. Az algoritmus kiválasztásakor ugyanakkor nem szabad megfeledkeznünk arról sem, hogy mind az állomásokba beépített technológiai megoldásoknak, mind a köztük lévő kommunikációnak ára van. Épp ezért előfordulhat, hogy a tényleges megvalósításkor egy egyszerűbb, olcsóbb, de kevésbé hatékony változat összességében előnyösebb lehet.

Megemlítjük a témakör egy lehetséges továbbgondolását. Minthogy a rádióhullámok terjedése a háromdimenziós térben történik, ezért az interferencia-gráfok is természetes módon elképzelhetők a térben, amely a fenti probléma vizsgálatának szó szerint új dimenzióját jelentheti.

Végezetül szeretnénk köszönetet mondani Rácz Andrásnak (Traffic Lab. Ericsson), aki konzulensként két éven keresztül nagy segítségünkre volt munkánkban, melyet az ún. témalabor tárgy keretében végeztünk el.

Irodalomjegyzék

1. Biró Péter, Lója Krisztina, Sütő Márton. Elosztott frekvenciakiosztási algoritmusok tervezése és vizsgálata. (TDK-dolgozat) http//:www.math.bme.hu/~biro/tdk
2. Bollobás B. The chromatic number of random graphs Combinatorica, vol. 8, 49–55, (1988).
3. C. A. Christen, S. M. Selkow. Some perfect colouring properties of graphs. J. Combin. Theory Ser. B 27, 49–59 (1979).
4. R. Graham, M. Grötschel, L. Lovász. Handbook of Combinatorics 4.fejezet (1995).
5. T. R. Jensen, B. Toft. Graph Coloring Problems John Wiley and Sons, New York, (1995).
6. L. Lovász, M. Saks, W. T. Trotter. An on-line graph coloring algorithm with sublinear performance ratio. Discrete Math. 75, 319–325, (1989).
7. R. A. Murphey, P. .M. Pardalos, M. G. C. Resende. Frequency assignment problems. Handbook of Combinatorial Optimalization, Kluwer Academic Publishers, (1999).

Hír

Az Ericsson 3G készülékplatformokra vonatkozó szerződést kötött az LG Electronicsszal. Az Ericsson technológiáján alapuló saját, értéknövelt GPRS/UMTS kettős üzemmódú 3G készülékeket fejleszt, és a technológiáért licencdíjat fizet.

A mobilkészülék-ipar jelenleg átalakulás alatt áll. Ugyanazok a változások zajlanak le, mint korábban a PC-iparban: elmozdulás a teljes termékválasztékot szállító néhány vállalat jellemezte vertikális iparágtól a számos specializált vállalatot magában foglaló horizontális iparág felé, ahol az egyes vállalatok csak a termék bizonyos alkotórészeit állítják elő.

A technológiaplatformok magukban foglalják az összes komponens specifikációját, a nyomtatott áramköri kártyák vázlatát és a szoftvert, a gyártó pedig saját igényei szerint alakíthatja át azokat. Ez lehetővé teszi az LG Electronics számára, hogy minimális K+F beruházás mellett gyorsan vezessen be a piacra új termékeket, ugyanakkor jelentős készülékszállítóként lépjen fel, amikor az ügyfelek a világ minden táján áttérnek a 3G-re.