



Jelzéscsökkentő mobilitás kezelési algoritmusok következő generációs mobil hálózatokban

Ph.D. értekezés tézisei

Simon Vilmos

Tudományos vezető:
Dr. Imre Sándor, DSc.
egyetemi tanár
Híradástechnikai Tanszék
Budapesti Műszaki és Gazdaságtudományi Egyetem

Budapest, 2008

1 Bevezetés

A következő generációs mobil hálózatok legizgalmasabb műszaki kihívásai közé tartozik a szolgáltatás minőségének (QoS) és az állandó kapcsolatnak a biztosítása úgy, hogy a felhasználók mobilitásából okozta változásokból a felhasználókat semmilyen hátrány vagy szolgáltatás minőségi romlás ne érje. A cellaméretök csökkenése miatt növekvő cellaváltások száma csak úgy kezelhető, amennyiben hatékony mobilitás kezelési eljárások kerülnek bevezetésre. E okból kifolyólag a mobil terminálok (MT) nyomon követése meghatározó fontosságú az észrevétlen és szolgáltatáskiesést nem okozó cellaváltások biztosítása szempontjából [1]. A mobil hálózatok a cellás elv segítségével tudnak hatékonyan gazdálkodni a rendelkezésükre álló szűk keresztmetszetet jelentő frekvenciasávval, a frekvencia újrafelhasználásának elvét alkalmazva, így találva megoldást a mobil terminálok és szolgáltatások számának rohamos növekedése által felmerült gondokra.

Hívás vagy adatcsomag érkezése esetén a hálózatnak be kell azonosítani a mobil terminál pillanatnyi tartózkodási helyét, ami az aktuális tartózkodási övezet bázisállomásainak végigkérdezése által történik meg, ezt paging-nek nevezi a szakirodalom [2]. A cellánál tágabb tartózkodási övezet bázisállomásainak halmaza a Location Area (LA). Minden LA határ átlépésnél a mobil terminálok tájékoztatják a hálózatot az új tartózkodási övezetükről, frissítve a helyzet-nyilvántartó adatbázisokat (helyzet frissítés ún. location update folyamat). A mobil terminálok Location Area-n belüli cellaváltásáról nem kell tájékoztatni a hálózatot, csak akkor ha ez a cellaváltás egyben Location Area váltás is. Így abban az esetben ha hívás érkezik, a hálózatnak a LA minden celláját végig kell pásztázni a paging eljárással, hogy beazonosítsa, hogy a mobil terminál a LA-n belül melyik cellában tartózkodik. A hálózati jelzésterhelést a helyzetfrissítés (location update) és paging műveletek nagy mértékben növelik, így e két művelet összehangolása és az általuk okozott költségek közötti egyensúly keresése a Location Management (Helyzet-nyilvántartás) terület feladata.

Széleskörben elterjedt location update eljárás a körzet-alapú megoldás (zone-based), amelynek során a hálózat lefedettségi területét Location Areakra tagolják úgy, hogy minden LA-t meghatározott cellacsoportok alkotnak. Abban az esetben ha a mobil terminál átlépi az adott LA határát, értesíti a hálózatot az átlépésről. A cellák csoportosítása Location Areakra előzetesen történik meg a felhasználók mobilitási mintái alapján. A jelenleg széleskörűen elterjedt mobil technológiák túlnyomó része ezt a körzet-alapú helyzetnyilvántartási megoldást használja, köztük a GSM [3], IS-41 [4] és az UMTS [5], különböző paging stratégiákkal karöltve. Figyelembe véve a mobil terminálok számának ugrásszerű növekedését és a második generációs mobil hálózatokról a harmadikra történő átállást, kiemelten fontos a Location Areák és a felettük található hálózati hierarchia körültekintő tervezése, mivel ezzel jelentősen csökkenthetőek a helyzetfrissítési és paging költségek.

2 Kutatási célkitűzések

A Location Management egyik fő feladata a helyzetfrissítés és paging eljárások által okozott jelzéstöbbletek közötti egyensúly megtalálása. A körzet-alapú helyzetnyilvántartási megoldásnál ezt a kompromisszumot a hatékony Location Area tervezéssel lehet megtalálni. Ez veti fel a kérdést, vajon milyen méretű és összetételű Location Area-kat kell tervezni ahhoz, hogy mindkét eljárás költsége csökkenjen.

A LA méret növelésének és csökkentésének is megvan a maga előnye. Egyfelől, ha minnél több cellát csoportosítunk egy LA-ba, akkor a LA-k közötti váltások száma csökkeni fog, így a helyzetfrissítő üzenetek száma is redukálódni fog. Viszont ebben az esetben egy bejövő hívás sok paging üzenetet fog generálni [6], ugyanis egy nagyobb halmazban kell megtalálni a mobil felhasználót. Ezek a keresések a gerinchálózatot és a vezeték nélküli részét is leterhelik

a hálózatnak. A másik esetben, ha csökkentjük a cellák számát LA-ként, akkor nem kell ilyen széleskörű keresést kifejteni, viszont a LA váltások száma növekedni fog. Így jelentős többletterhelés éri a gerinchálózatot, miközben a felhasználható rádiós sáv szélesség is lecsökken, ráadásul a helyzetnyilvántartó adatbázisok módosítása is költséggel jár. Komoly hátránya még a szükségtelen helyzetfrissítéseknek a mobil terminálok megnövekedett energiafogyasztása is. Ebből következik, hogy a LA tervezés visszavezethető a helyzetfrissítési és a paging költség közötti hatékony egyensúlykeresésre.

A körzet-alapú megoldáshoz használt LA tervezésnek komoly szakirodalma van, ezek a LA tervezési megoldások két nagy csoportba sorolhatóak. Az első csoportban lévő megoldásoknál egyenletes eloszlású felhasználó és cellaváltási rátákat feltételeznek a LA tervezés során [7], [8]. Ezen eredmények gyakorlati alkalmazhatóság erősen kétségbe vonható, ugyanis a hálózati és mobilitási karakterisztikák nem homogének a gyakorlatban [9]. A második csoportbeli LA tervezési megoldásoknál a hálózatot egy gráf írja le [10], [11], ahol a celláknak a gráf csomópontjai, míg a cellaváltási irányoknak a gráf élei feleltethetők meg. Így a LA tervezési kérdés egy gráf partícionálási problémává képezhető le. A LA tervezési módszerek többsége a Location Areankénti optimális cellaszám meghatározásával foglalkozik [12], [13], [14], nem a LA csoportok optimális összetételével.

Kutatásom célja ennek megfelelően olyan LA tervezési megoldások megalkotása volt, amelyek segítségével meg lehet határozni a cellák azon optimális csoportosítását, amely hatékony LA felosztást tesz lehetővé. Mivel az egyenletes eloszlású cellaváltási ráta nem mindig tekinthető valószerűnek, ezért olyan LA tervezési eljárásokat dolgoztam ki, amelyeket heterogén mobilitási környezetre, majd a bemutatott módosításokkal homogén mobilitási környezetre is lehet használni, ez teljesen újszerűnek számít a téma kutatásában. A heterogén esetben a helyzetfrissítési és az aggregált költség (helyzetfrissítési és paging költség súlyozott összege) csökkentése volt a cél, míg a homogén esetben a paging költség felső korlátként szerepelt, a helyzetfrissítési költségfüggvénnyel mint minimalizálandó célfüggvénnyel.

Ez a két változatos LA tervezési módszer (1.Téziscsoport) szolgál bemenetként a hierarchikus hálózati struktúrát tervező algoritmusomhoz (2.Téziscsoport), így a kettő egy cellás hálózat tervezési keretrendszer alkot.

További célként tűztem ki, hogy a központi infrastruktúra és állandó összeköttetés nélküli mobil ad hoc hálózatokra olyan információ terjesztési algoritmusokat dolgozzak ki, amelyek segítségével csökkenthető a hálózati többletterhelés (3.Téziscsoport). Így a jelenleg már széleskörűen elterjedt cellás hálózatok tervezési elvei mellett önszerveződő mobil hálózatokra dolgoztam ki költségcsökkentő algoritmusokat, mely önszerveződő hálózatok megjelenésével teljesen új korszak kezdődhet a mobil távközlés evolúciójában.

3 Kutatási módszerek

A bemutatott eredményekhez analitikus és szimulációs vizsgálatok során jutottam. Az analitikus részben költségfüggvény struktúrát definiáltam a LA és hierarchia tervező megoldásokhoz, gráfelméleti és folyadék modellel elemeztem a hálózatot és a mobil terminálok cellák és LA-k közötti mozgását, illetve valószínűségi fára képeztem le a hálózati hierarchiát, továbbá megmutattam a saját LA tervezési módszer előnyét más eljárással szemben.

A szimulátorban implementált LA és hierarchia tervező illetve információ terjesztési algoritmusok teljesítményhatékonyágát a kutatási terület megfelelő algoritmusáival mértem össze nagyszámú szimulációs vizsgálat során. A LA tervező algoritmusok bemenete egy mobilitás szimulátorral lett generálva, amely valószerű cellahatár átlépési ráta és bejövő hívás adatbázisokat szolgáltat.

4 Új tudományos eredmények

4.1 Location Area tervező algoritmusok

1. Téziscsoport ([D1], 2.1-2.7, 3.1-3.4, [B1], [B2], [J1], [J2], [C1], [C2], [C3], [C4], [C5], [C7]) *Definiáltam a paging és helyzetfrissítési költségfüggvényt heterogén és homogén mobilitási környezetre. Módszert adtam a Location Area-nkénti optimális cellaszám meghatározására homogén mobilitási környezetben. Mindkét mobilitási környezetre új Location Area tervezési algoritmusokat dolgoztam ki, a mobil felhasználók cellahatár átlépési valószínűségei alapján. Megmutattam analitikusan egyedi esetekre és szimulációs vizsgálatokkal általánosságban, hogy jelentős jelzsmennyiség csökkenés érhető el mindkét környezetre az új Location Area tervező algoritmusokkal.*

4.1.1 Location Area tervező algoritmusok heterogén mobilitási környezetre

A LA tervezéshez kötődő kutatások többsége a LA-kénti optimális cellaszám meghatározását tűzi ki célul, ezzel szemben ebben a téziscsoportban olyan LA tervező algoritmusokat mutatok be, amelyek megadják a cellák hatékony felosztását LA-kba egy heterogén mobilitási környezetben, ahol a cellahatár átlépési ráták nem egyenletes eloszlásúak. A szakirodalomban fellelhető publikációk jelentős része azzal az egyszerűsítéssel él, hogy ezek közel egyenletes eloszlást követnek, viszont ez nem minden esetben fedti a valóságot. Ezért a heterogén mobilitási környezetben a célom a helyzetfrissítési és paging költség súlyozott összegének csökkentése volt. Ezzel szemben az egyenletes eloszlású esetben (homogén környezet) a célom a helyzetfrissítési költség minimalizálása volt, a paging költséggel mint felső korláttal. Így olyan újszerű LA tervezési módszert adtam, amely mindkét mobilitási környezetre megtalálja a leghatékonyabb cellacsoportosítást.

A LA tervező megoldásom alapötlete, hogy azon cellák kerüljenek egy LA-ba, amelyek között a cellahatárátlépési ráta magas, így csökkenthető a LA-k közötti váltások száma, ezúttal a helyzetfrissítő üzenetek mennyisége is. Viszont az a feladat, amikor a cellák egy adott halmazát úgy kell felosztani diszjunkt részhalmazokra, hogy a részhalmazok kardinalitása ne haladjon meg egy korlátot illetve a részhalmazon belüli cellaváltások száma maximális legyen, egy NP-teljes probléma [15].

Ezért egy lehetséges megoldása a problémának az optimális LA felosztás közelítése heurisztikus algoritmussal, majd a heurisztikus megoldással kapott LA struktúra további finomítása különböző technikákkal. A probléma komplexitása miatt a LA tervező algoritmusok analitikus vizsgálata csak egyedi esetekben lehetséges, általános megállapításokat csak kiterjedt szimulációs vizsgálatokkal lehet kimondani.

A heterogén környezetben alkalmazható LA tervezési eljárásom két fázisra bontható: először a helyzetfrissítési költség csökkentése egy heurisztikus algoritmussal, majd az így kialakuló LA struktúra lesz a kiindulási alapja az újracsoportosító algoritmusnak, amely az aggregált költségfüggvényt redukálja.

Ehhez definiáltam a paging és helyzetfrissítési költségfüggvényt, majd az aggregált költségfüggvényt egy heterogén környezetre.

Hívás érkezése esetén a hálózat paging üzenetet küld minden cellájába annak a LA-nak, ahol az adott mobil terminál be van jelentve, hogy megtalálja a pontos tartózkodási helyét a LA-n belül. Így a LA minden celláját terheli az összes a LA-ban tartózkodó MT paging forgalma. Így definiálható az l -ik LA-ra a paging költség egységnyi időintervallumra:

$$C_{p_l} = \sum_{i=1}^{K_l} N_l \cdot \lambda_{il} \cdot B_p, \quad (1)$$

ahol N_l a cellák száma az l -ik LA-ban, λ_{il} az i -ik MT-ba az egységnyi időintervallum alatt beérkező hívások rátája, B_p a paging üzenet átviteli költsége, és K_l a MT-ok száma az l -ik LA-ban.

A teljes paging költség az összes LA-ra a rendszerben az egységnyi időintervallum alatt:

$$C_p = \sum_{l=1}^M C_{p_l} = \sum_{l=1}^M \sum_{i=1}^{K_l} N_l \cdot \lambda_{il} \cdot B_p = B_p \cdot \sum_{l=1}^M N_l \cdot \sum_{i=1}^{K_l} \lambda_{il}, \quad (2)$$

ahol M a LA-k száma a rendszerben.

Definiáltam a helyzetfrissítési költségfüggvényt is (Location Update Cost) az l -ik LA-ra egységnyi időintervallumra:

$$C_{lu_l} = B_{lu} \cdot \sum_{j=1}^{D_l} q_{jl}, \quad (3)$$

ahol B_{lu} a helyzetfrissítési üzenet átviteli költsége, q_{jl} a j -ik cellahatáron történő átlépés intenzitása az egységnyi időintervallum alatt, míg D_l az l -ik LA külső cellahatárainak száma

A teljes helyzetfrissítési költségfüggvény az összes LA-ra a rendszerben:

$$C_{lu} = \sum_{l=1}^M C_{lu_l} = B_{lu} \cdot \sum_{l=1}^M \sum_{j=1}^{D_l} q_{jl} \quad (4)$$

A cél a súlyozott aggregált költségfüggvény minimalizálása, a w_1 és w_2 változtatható súlyokkal lehet figyelembe venni a két költségfüggvény fontosságát a hálózat tervezője szempontjából, jellemzően $w_2 = H_{pref} \cdot w_1$ és $H_{pref} \in [2,10]$:

$$\min C_{total} = \min \{ w_1 \cdot C_p + H_{pref} \cdot w_1 \cdot C_{lu} \}. \quad (5)$$

Tézis 1.1 Kidolgoztam egy új heurisztikus Location Area tervező algoritmust (Lafa, Location Area Forming Algorithm) heterogén mobilitási környezetre, mely megadja a $G(V, E)$ gráf maximális súlyú feszítőerdőjét.

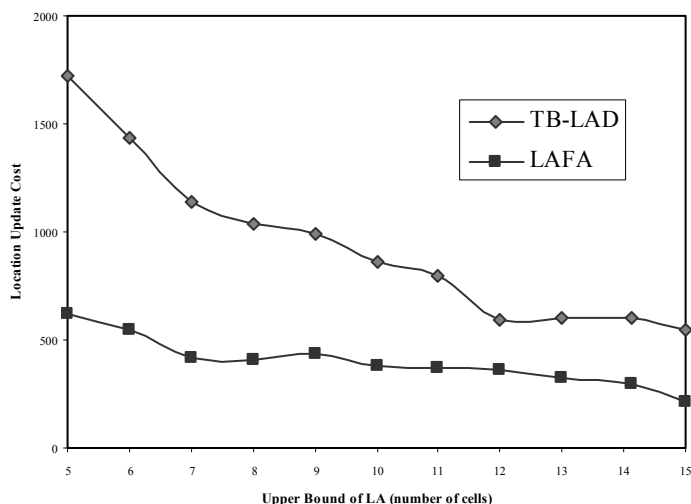
Megmutattam egyedi esetekre analitikusan és általánosságban szimulációs vizsgálatokkal, hogy a Lafa algoritmus hatékonyabban csökkenti a helyzetfrissítési költséget mint a létező megoldások, egyidejűleg nem növelve a cellák átlagos számát Location Area-nként, alacsony szinten tartva a paging költséget. Megmutattam, hogy azokban az esetekben amikor a LA-k átlagosan 5-10 cellából állnak a helyzetfrissítési költség csökkenése eléri a 40-60%-ot.

A $G(V, E)$ gráffal modelleztem a hálózatot, ahol a cellák a gráf $v \in V$ csomópontjainak, a cellahatár átlépesi irányok pedig a gráf $e \in E$ éleinek feleltethetőek meg. A gráf éleihez nem negatív súlyokat rendeltem a $[0,1]$ tartományból, e súlyok egyenlőek azokkal a valószínűségekkel, amelyekkel a mobil terminálok használják ezeket a határátlépesi irányokat. Kidolgoztam egy új LA tervező algoritmust (LAFA, Location Area Forming Algorithm), amely arra a fenn vázolt problémára ad konkrét megoldást, hogy a cellák halmazát hogyan lehet diszjunkt részhalmazokra bontani úgy, hogy minden részhalmaz számossága felülről korlátos legyen és hogy a részhalmazokon belüli cellaváltások száma maximális legyen. Mivel ez egy NP-teljes probléma [15], a LAFA egy olyan heurisztikus algoritmus, amely úgy szolgáltat egy hatékony LA felosztást, hogy a $G(V, E)$ gráf maximális súlyú feszítőerdőjét keresi meg. A LAFA algoritmus a $G(V, E)$ gráf $G_i(V, E)$ részgráfokra való felosztását adja meg, úgy hogy a $G(V, E)$ gráf maximális súlyú feszítőfájának komponensei a megfelelő $G_i(V, E)$ részgráfok maximális súlyú feszítőfái, és minden részgráf számossága kisebb egy korlátnál illetve egy speciális LA bővítés megállási szabályt is használ.

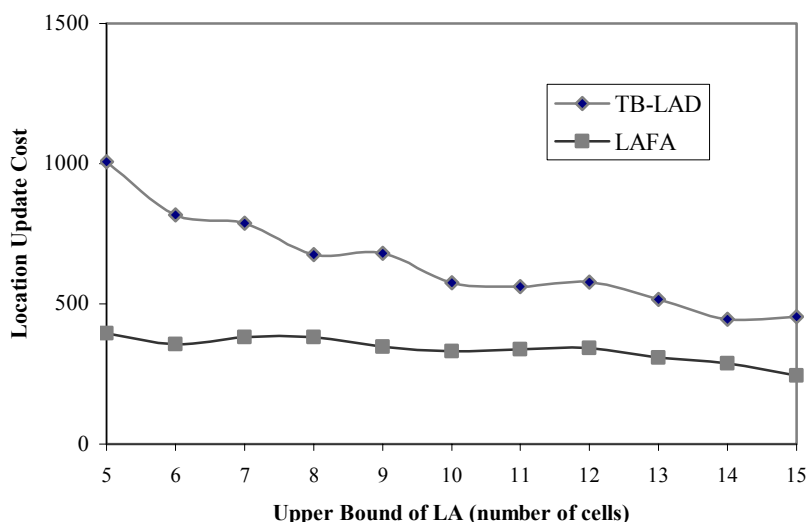
Megmutattam analitikusan, hogy a LAFA hatékonyabb a szakirodalomban gyakran hivatkozott Traffic-Based Static Location Area Design (TB-LAD) [15] megoldásnál, az elosztott LA formálási jellegének köszönhetően.

Nagyszámú szimulációs vizsgálatot végeztem el a LAFA és a TB-LAD algoritmus teljesítményének összehasonlítása érdekében egy rurális és városi mobilitási környezetben. A vizsgálatok legfontosabb következtetései:

- rurális mobilitási környezetben a LA-kénti cellák számának felső korlátját növelve (N_{\max}) a LAFA hatékonyabban csökkenti a helyzetfrissítési költséget a TB-LAD megoldásnál, alacsony szinten tartva a paging költséget. Ha $N_{\max} \in [5,10]$, a LAFA átlagban akár 40-60%-al is képes csökkenteni a helyzetfrissítési költséget (1. ábra)
- városi környezetben a LAFA szintén képes jelentősen csökkenteni a helyzetfrissítési költséget, hatékonyabban mint a TB-LAD megoldás, a paging terhelés növekedése nélkül. A költség redukció az $N_{\max} \in [5,11]$ intervallumban a legnagyobb mértékű. A 12 és 15 cella közötti intervallumban a LAFA bővítés leállási feltétele teljesül, így a korlát növekedése ellenére a LA-k mérete nem növekszik már. A TB-LAD viszont minden esetben a felső korlátig növeli a LA-kénti cellaszámot, így nagyobb helyzetfrissítési költséget produkál nagyobb átlagos cellaszám mellett, vagyis a paging költsége is magasabb (2. ábra).



1. ábra A helyzetfrissítési költség rurális környezetben a TB-LAD és LAFA partíciókra



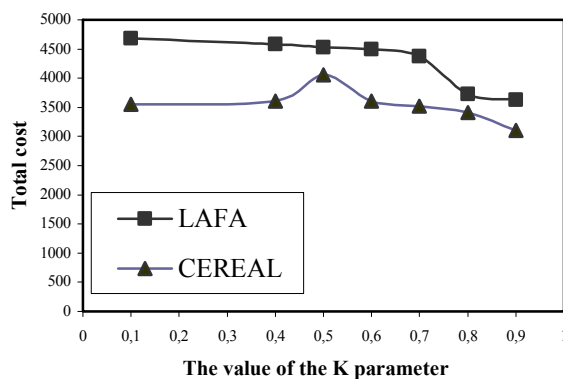
2. ábra A helyzetfrissítési költség városi környezetben a TB-LAD és LAFA partíciókra

Tézis 1.2 *Kidolgoztam egy Location Area újracsoportosító algoritmust (CEREAL, Cell Regrouping Algorithm), mely a LAFA algoritmus által nyújtott Location Area struktúrát teszi hatékonyabbá az aggregált költség szempontjából. Megmutattam szimulációs vizsgálatokkal, hogy a CEREAL algoritmus nagyobb mértékben csökkenti az aggregált költséget mint a LAFA algoritmus.*

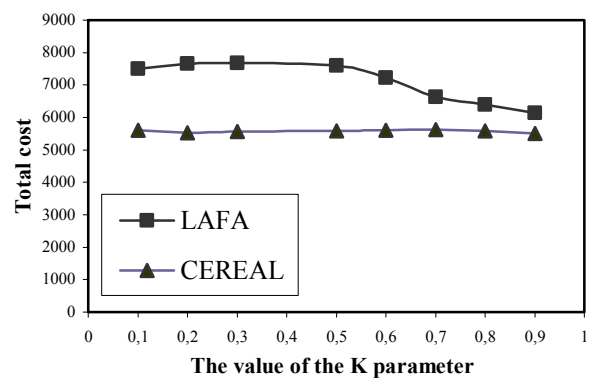
A CEREAL algoritmus a LAFA algoritmus által nyújtott LA struktúrából kiindulva, a cellák szomszédos LA-kba történő áthelyezésével csökkenti az aggregált költséget. A CEREAL aggregált költség csökkentő képességét a K paraméter függvényében vizsgáltam. A K paraméter dinamikusan módosítható, a használt modell függvényében. Értékét a $(0,1)$ intervallumban veheti fel, 1-hez konvergálva a LA-k mérete csökkenni fog, ami a helyzetfrissítő költség növekedését vonja maga után, míg a paging költség ezzel ellentétesen csökkenni fog.

Szimulációs vizsgálatokat végeztem arra az esetre, amikor a CEREAL algoritmust a LAFA által kialakított LA struktúrára alkalmazzuk, figyelve az aggregált költség alakulását (C_{total}) a K paraméter függvényében. A vizsgálatok legfontosabb következtetései:

- rurális környezetben a K paraméter növelésével az aggregált költség csökken, ellentétben a helyzetfrissítési költséggel. Ez azzal van összefüggésben, hogy a K paraméter növelésével a LA-k mérete csökken, így a paging költség is jelentősen csökken. Ebben az esetben a CEREAL algoritmus jobban teljesít mint a LAFA, a $K \in [0.1, 0.4]$ és $K \in [0.6, 0.7]$ intervallumokban akár 40%-al hatékonyabb csökkenti az aggregált költséget (3.ábra)
- városi környezetben a CEREAL algoritmus átlagban 30-40%-al hatékonyabban csökkenti az aggregált költséget a LAFA algoritmushoz képest (4-ábra).



3.ábra Az aggregált költség rurális környezetben



4.ábra Az aggregált költség városi környezetben

Attól függően, hogy melyik költségfüggvény csökkentése a cél, lehet alkalmazni a két LA tervező algoritmust: ha csak a helyzetfrissítési költség redukálása a cél, akkor a LAFA a megoldás, amennyiben viszont az aggregált költség a fontosabb, akkor a LAFA után a CEREAL algoritmust is alkalmazni kell.

4.1.2 Location Area tervező algoritmusok homogén mobilitási környezetre

A LA tervezés témájában újdonságnak számít, hogy a LA tervezési módszereim, azon felül, hogy jobb eredményeket adnak a meglévő megoldásoknál, heterogén és homogén mobilitási környezetre is alkalmazhatóak (nem egyenletes és közel egyenletes eloszlású cellaváltási ráta). Míg heterogén környezetnél a cél a helyzetfrissítési és aggregált költség minimalizálása volt, addig a homogén esetben úgy kell minimalizálni a helyzetfrissítési költséget, hogy fennáll egy paging költség korlát. Azért van szükség e paging korlátra, mivel ha a homogén esetre is a LAFA algoritmus LA bővítés leállási feltételét használnánk, akkor minden cellát egy LA-ba csoportosítana, figyelmen kívül hagyva a cellás hálózatok paging kapacitását.

Ezért homogén esetben úgy kell meghatározni a cellák optimális számát LA-nként, hogy a paging korlátot ne haladjuk meg, miközben a helyzetfrissítési költségfüggvény erre az értékre minimális, ráadásul a LA struktúra is a lehető leghatékonyabb legyen.

Az általam a homogén esetre kidolgozott LA tervezési módszer két fázisból áll: először egy mohó algoritmus (GREAL) megalkotja a kiindulási LA felosztást, majd egy szimulált lehűtésen alapuló algoritmus (SABLAF) teszi még hatékonyabbá ezt a felosztást.

Tézis 1.3 Egy folyadék áramláson alapuló analitikus modellt vezettem be a mobil terminálok Location Area-k közötti mozgásának leírására homogén mobilitási környezetben. A modell legfontosabb eredményei a Location Area kerület és a határcellák számának meghatározása, illetve a helyzetfrissítési költségfüggvény és a paging költség korlát bevezetése.

A modell alapján módszert adtam az optimális cellaszám meghatározására LA-nként, amellyel a helyzetfrissítési költségfüggvény a minimumát éri el a paging korlát betartása mellett.

Minden alkalommal amikor a mobil terminál olyan cellahatárt lép át, amely egyben LA határ is, helyzetfrissítő üzenet kerül kiküldésre, így a LA-n belüli mozgás nem számít bele a helyzetfrissítési költségbe. Ezért definiáltam a k -ik LA határcelláinak számát (N_{pk} , az N_k részhalmaza) és e határcellák kerületének azon átlagos hányadát, amely hozzájárul a k -ik LA kerületéhez ($\delta_{pk}(N_k)$). E két egyenlet segítségével megadható a k -ik LA kerülete:

$$P_k = \kappa \cdot \sqrt{N_k} \cdot P_c \cdot (a + b \cdot N_k^{\eta-1}) \quad (6)$$

Behelyettesítve a mobil terminálok sűrűségét (ρ) és P_k -át a folyadék áramlási modell kiáramlási rátájába (a kiáramlási ráta megegyezik a k -ik LA határátlépési számával):

$$R_{out} = \left(\frac{v \cdot \frac{K}{N_k \cdot S} \cdot \kappa \cdot \sqrt{N_k} \cdot P_c \cdot (0.333 + 0.309 \cdot N_k^{-0.425})}{\pi} \right) \quad (7)$$

Ezek alapján definiáltam a helyzetfrissítési költségfüggvényt egységnyi időintervallumra:

$$C_{LU_k} = B_{LU} \cdot v \cdot K \cdot \kappa \cdot P_c \cdot \left(\frac{0.333 \cdot N_k^{-0.5} + 0.309 \cdot N_k^{-0.925}}{\pi \cdot S} \right) \quad (8)$$

Figyelembe véve a cellás hálózat paging kapacitását bevezettem a k -ik LA paging korlátját:

$$C_{P_k} = B_P \cdot N_k \cdot K_k \cdot \lambda < C_k \quad (9)$$

Módszert adtam az optimális LA-nkénti cella számra, amire a helyzetfrissítési költségfüggvény minimális lesz a paging korlát betartása mellett (C_k nem léphető át), a maximális értéke (N_{opt}) az N_k -nak a következő egyenlőtlenségből számolható és bemenetként szolgál a GREAL és SABLAF algoritmusok számára:

$$P(\lambda < \frac{C_k}{B_P \cdot N_k \cdot K}) = 1 - e^{-\lambda} \quad (10)$$

Tézis 1.4 *Kidolgoztam egy szimulált lehűtésen alapuló új Location Area tervező algoritmust homogén mobilitási környezetre (SABLAF, Simulated Annealing based Algorithm). Megmutattam szimulációs vizsgálatokkal, hogy rurális és városi környezetre a SABLAF algoritmus hatékonyabban redukálja a helyzetfrissítési költséget, mint a létező megoldások.*

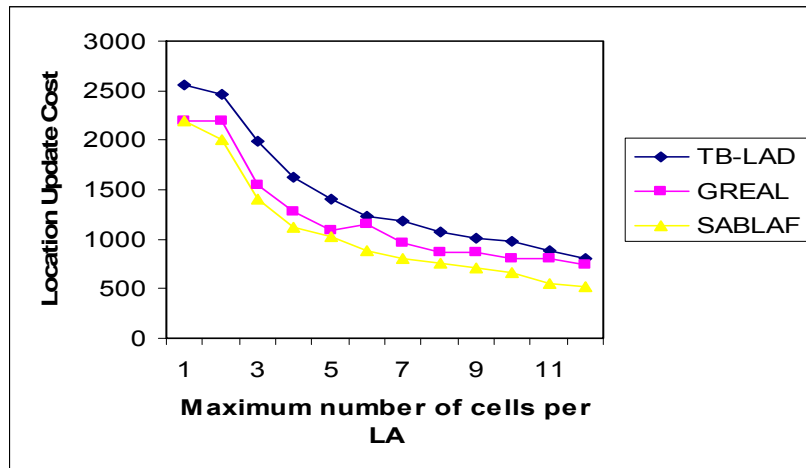
Módosítottam a LAFA algoritmust, hogy használható legyen a közel egyenletes eloszlású cellaváltási rátákra is, e módosított LA tervező algoritmus a GREAL (Greedy Algorithm), amely a szimulált lehűtésen alapuló SABLAF algoritmus kiindulási LA struktúráját állítja elő. A GREAL a $G(V, E)$ gráf felosztását adja meg $G_i(V, E)$ részgráfokra, a maximális feszítőerdő előállításával a $G(V, E)$ gráfban. E maximális feszítőerdő komponensei a megfelelő $G_i(V, E)$ részgráfok maximális feszítőfái, miközben a csomópontok száma részgráfonként kisebb vagy egyenlő mint az N_{opt} , így a GREAL ezt a LA bővítés leállási feltételt használja a LAFA algoritmusban használt helyett.

A LA tervezési megoldásom első kiinduló lépését adja meg a GREAL algoritmus a szimulált lehűtésen alapuló SABLAF algoritmusnak, a cellaváltási rátákkal a szomszédos határcellák között. Teljesítményhatékonysági vizsgálatot folytattam a GREAL, SABLAF és TB-LAD algoritmus összehasonlítására, két különböző mobilitási környezetet használva (rurális és városi).

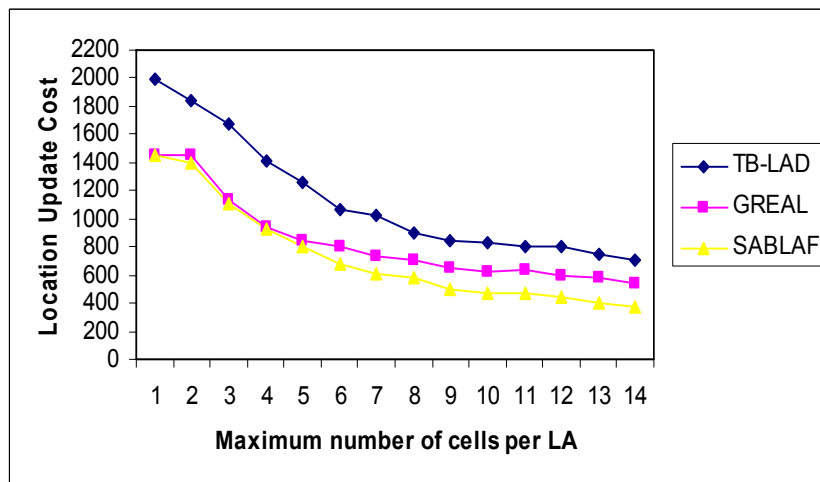
A vizsgálatok legfontosabb következtetései:

- rurális környezetben a GREAL algoritmus képes csökkenteni a helyzetfrissítési költséget a TB-LAD megoldáshoz képest, viszont a SABLAF algoritmust találja meg minden maximális cellaszám értékre a leghatékonyabb LA struktúrát. A számolt $N_{opt} = 12$ esetre éri el a szimulált lehűtésen alapuló SABLAF a helyzetfrissítési költség minimális értékét (5.ábra)
- városi környezetben a maximális cellaszám kisebb értékeire (ahol ez a korlát kisebb a Location Area-nkénti öt cella értéknél) a SABLAF elfogadja módosítás nélkül a GREAL által nyújtott LA partíciókat, de ahogy a cellák maximális száma növekszik, úgy a SABLAF felülmúlja a két másik algoritmus által nyújtott megoldásokat. Az $N_{opt} = 14$ számolt értékre (amely nagyobb a rurális környezetben számolt értéknél a cella méretek miatt) a SABLAF ismét a helyzetfrissítési költség legalacsonyabb értékét produkálja (6.ábra).

Összefoglalva: mindkét mobilitási környezetre a SABLAF algoritmus nyújtja a legjobb megoldást, átlagosan 50%-al hatékonyabban csökkentve a helyzetfrissítési költséget, mint a TB-LAD algoritmus, illetve átlagban 20%-al hatékonyabban, mint a GREAL algoritmus.



5.ábra A helyzetfrissítési költség rurális környezetben



6.ábra A helyzetfrissítési költség városi környezetben

4.2 Hierarchikus hálózati struktúrát optimalizáló algoritmusok

2. Téziscsoport ([D1], 4.1-4.6; [J3], [C11]) *Levezettem egy költségfüggvény struktúrát a helyzetfrissítési és csomagkézbesítési költség modellezésére Proxy Ügynök Architektúrában (Proxy Agent Architecture). Egy valószínűségi fán alapuló modellt adtam a Proxy Ügynök Architektúra hierarchikus hálózatára, bevezetve a hozzáférési útvonalválasztó (Access Router) átlagos távolságát a hierarchia csúcsán található útvonalválasztótól. Kidolgoztam egy hierarchikus hálózat tervező algoritmust (HIENDA, Hierarchical Network Design Algorithm), amely olyan valószínűségi fát generál, ahol a hozzáférési útvonalválasztó átlagos távolsága a hierarchia csúcsán található útvonalválasztótól minimális. Megmutattam szimulációs vizsgálatokkal, hogy a HIENDA hatékonyabban csökkenti a helyzetfrissítési költséget a létező megoldásokhoz képest, miközben alacsony szinten tartja a csomagkézbesítési költséget.*

A Mobile IPv6 [16] az IP protokoll kiterjesztése a mobil csomópontok mozgásának kezelésére, amely viszont nem képes a cellaváltások valós időben történő kezelésére. E hiányossága miatt csak a makro-mobilitás kezelésére használják, míg a mikro-mobilitási

tartományokon belüli mobilitás kezelésével külön protokollok foglalkoznak. E mikro-mobilitási megoldások egyik legjelentősebbikje a jelzési üzenetek és késleltetés csökkentés szempontjából a Hierarchikus Mobile IPv6 [17]. A megoldás alapötlete, hogy a mikro-mobilitási tartományok forgalmát egy hierarchikus architektúra szolgálja ki, egy mobilitási ügynökkel a hierarchia csúcsán. A mobilitási ügynök egy új típusú Mobile IPv6 csomópont, amelyet mobilitási horgonypontnak (Mobility Anchor Point, MAP) neveznek és a hierarchiába szervezett útvonalválasztók bármelyik szintjén működhet, beleértve a hozzáférési útvonalválasztókat is (Access Router, AR). A hozzáférési útvonalválasztók a mobil csomópontok alapértelmezett útválasztói, amelyek összegyűjtik az adott tartományban mozgó terminálok kimenő forgalmát. A MAP csomópontok alkalmazása csökkenti a Mobile IPv6 által generált jelzésmennyiséget a rádiós interfészen, mivel a Mobile IPv6 jelzések a helyi mikromobilitási tartományon belül maradnak, korlátozva az innen kimenő jelzésmennyiséget.

A szabványok (a 4140-es RFC sem) nem térnek ki a hierarchia megvalósítási részleteire a hálózattervezés folyamán. Több lényeges kérdést vet ez fel: milyen elvek alapján történjen a hierarchia szintek meghatározása, a cellák hozzáférési útvonalválasztókhoz történő csoportosítása, illetve melyik hierarchia szinten javasolt a MAP funkciót implementálni.

Az első téziscsoportomban olyan Location Area tervezési megoldásokat adok, amelyek segítségével csökkenthető a cellaváltások által okozott járulékos jelzésmennyiség. Ezáltal a LA tervező eljárások megadják hogyan csoportosítsuk a cellákat tartományokba, amelyeknek a forgalmáért az adott AR a felelős. A kulcskérdés hogyan vonjuk össze ezeket az AR-eket a hierarchia következő szintjén, illetve melyik szinten implementáljuk a MAP funkcionalitást, vagyis hány AR tartozzon egy adott MAP alá, amely meghatározó fontossággal bír a hálózat teljesítménye szempontjából. Olyan általános hierarchia tervező megoldást dolgoztam ki, amely nem csak HMIPv6-ban, hanem más a proxy ügynök architektúra (Proxy Agent Architecture) csoporthoz tartozó mikro-mobilitási protokollban is használható.

Az egyes hierarchiatervező megoldások hatékonyságának méréséhez viszont szükség van a helyzetfrissítési és csomagkézbesítési költség meghatározására. Megvizsgálva a helyzetfrissítő üzenet útját az otthoni hálózat és a MAP által lefedett mobil csomópont között egy proxy ügynök architektúrában, a mobil terminál minden egyes alhálózat váltása a következő helyzetfrissítési költséget generálja:

$$C_{LU} = 2 \cdot T_{HA-MAP} + 2 \cdot T_{MAP-AR} + 2 \cdot T_{AR-MN} + p_{HA} + 2 \cdot p_{MAP} + 2 \cdot p_{AR}, \quad (11)$$

ahol a T_{HA-MAP} , T_{MAP-AR} , és T_{AR-MN} a helyzetfrissítő üzenet átviteli költsége a Home Agent (HA), a MAP, az AR és a mobil csomópont között, míg p_{HA} , p_{MAP} és p_{AR} a helyzetfrissítési üzenet feldolgozási költsége a HA-ben, a MAP-ban és az AR-ben.

Ha a mobil csomópont Access Router-t vált, de marad ugyanazon MAP alatt, akkor az otthoni ügynökét nem kell értesíteni a helyi váltásról, így csak egy helyi helyzetfrissítési költség keletkezik:

$$C_{LUI} = 2 \cdot T_{MAP-AR} + 2 \cdot T_{AR-MN} + p_{MAP} + 2 \cdot p_{AR}. \quad (12)$$

Az átviteli költség arányos a forrás és célállomás távolságával, ami a csomag által megtett n_{S-D} hopszámmal és a K_T arányossági tényezővel fejezhető ki, míg a vezeték nélküli szakasz átviteli költsége μ -val nagyobb mint a vezetékese.

Mivel a mobil terminál AR váltásainak csak ε része ténylegesen MAP váltás is (a többi csak olyan AR váltás, ami a MAP lefedési területén belül zajlik le), így a teljes helyzetfrissítési költség a következőképpen írható fel:

$$C_{LUtotal} = (1 - \varepsilon) \cdot q \cdot C_{LUI} + \varepsilon \cdot q \cdot C_{LU} \quad (13)$$

$$C_{LUtotal} = q \cdot (2 \cdot \varepsilon \cdot n_{HA-MAP} \cdot K_T + 2 \cdot n_{MAP-AR} \cdot K_T + 2 \cdot \mu \cdot n_{AR-MN} \cdot K_T + \varepsilon \cdot p_{HA} + (1 + \varepsilon) \cdot p_{MAP} + 2 \cdot p_{AR}) \quad (14)$$

ahol q az AR váltások száma a hálózatban az egységnyi időintervallum alatt.

Mobile IP alkalmazása esetén minden csomagot amit a mobil csomópontnak küldenek először az otthoni ügynöke fogad, majd „alagúton” átküldi annak a MAP-nak, amelyiknél a mobil csomópont be van jelentkezve, aki ezt a mobil csomópont AR-ének küldi tovább. Ez egy járulékos átviteli és feldolgozási költséggel jár:

$$C_{PD} = \sum_{k=1}^N \theta_i (D_{HA-MAP} + D_{MAP-AR} + h_{HA} + h_{MAP}), \quad (15)$$

ahol N a mobil csomópontok száma a hálózatban, θ a mobil csomóponthoz beérkező csomagok száma az egységnyi időintervallum alatt, míg D_{HA-MAP} és D_{MAP-AR} a csomagkézbesítés átviteli költségei, h_{HA} és h_{MAP} pedig a csomagkézbesítés feldolgozási költségei.

Hasonlóan a helyzetfrissítési költséghez, az átviteli költség a csomagkézbesítési műveletnél is arányos a forrás és célállomás közötti hopszámmal és egy K_D arányossági tényezővel:

$$C_{PD} = \sum_{k=1}^N \theta_i (n_{HA-MAP} \cdot K_D + n_{MAP-AR} \cdot K_D + h_{HA} + h_{MAP}). \quad (16)$$

Tézis 2.1 Levezettem a következő költségfüggvényeket, amelyekkel modellezni lehet a jelzésmennyiséget Proxy Ügynök Architektúrában:

- Lokális helyzetfrissítési költség

$$C_{LUI} = 2 \cdot T_{MAP-AR} + 2 \cdot T_{AR-MN} + p_{MAP} + 2 \cdot p_{AR} \quad (12)$$

- Teljes helyzetfrissítési költség

$$C_{LUtotal} = q \cdot (2 \cdot \varepsilon \cdot n_{HA-MAP} \cdot K_T + 2 \cdot n_{MAP-AR} \cdot K_T + 2 \cdot \mu \cdot n_{AR-MN} \cdot K_T + \varepsilon \cdot p_{HA} + (1 + \varepsilon) \cdot p_{MAP} + 2 \cdot p_{AR}) \quad (14)$$

- Csomagkézbesítési költség

$$C_{PD} = \sum_{k=1}^N \theta_i (n_{HA-MAP} \cdot K_D + n_{MAP-AR} \cdot K_D + h_{HA} + h_{MAP}) \quad (16)$$

Tézis 2.2 Egy valószínűségi fán alapuló modellt adtam a Proxy Ügynök Architektúra hierarchikus hálózatára a hozzáférési útvonalválasztók közötti váltás rátákat alapul véve.

A proxy ügynök architektúra valószínűségi fája egy olyan véges fagraf, ahol minden csomóponthoz egy nem negatív súly van hozzárendelve a következő definíciók alapján:

- A valószínűségi fa minden közbenső csomópontja egy változóval, míg minden levél csomópontja egy valószínűséggel reprezentálható.
- A gyöker csomópont valószínűsége 1.

- Minden közbenső csomópont valószínűsége egyenlő a hozzá tartozó részfa (minden közbenső elem egy részfa gyökereként tekinthető, így a fa részfákra bontható) csomópontjai valószínűségeinek összegével
- A levél csomópontok valószínűsége a hozzáférési útvonalválasztók közötti váltási rátákkal arányos: az adott hozzáférési útvonalválasztók közötti váltási ráta és a hálózatban végbemenő teljes váltásmennyiség hányadosa

A Huffman kódolásból ismert úthossz lemmát alkalmaztam az AR-ek $E[L]$ átlagos távolságának számolására a csúcson lévő útvonalválasztótól, ahol L az út hosszának valószínűségi változója és egyenlő a közbenső csomópontok valószínűségeinek összegével, a közbenső csomópontok között érve a gyökér csomópontot is:

$$P_{root} + P_{I1} + P_{I2} + \dots + P_{IR} = \sum_{i=1}^T n_i \cdot P_H(h_i) = E(L) \quad (17)$$

ahol R a közbenső útvonalválasztók száma, T az AR-ek száma, n_i az adott AR és a csúcson lévő útvonalválasztó közötti hopszám, illetve $P_H(h_i)$ az AR váltási valószínűségek eloszlása.

Tézis 2.3 *Kidolgoztam egy hierarchikus hálózat tervező algoritmust (HIENDA, Hierarchical Network Design Algorithm), amely olyan valószínűségi fát generál, ahol a hozzáférési útvonalválasztó $E(L)$ átlagos távolsága a hierarchia csúcsán található útvonalválasztótól minimális. Megmutattam szimulációs vizsgálatokkal, hogy a HIENDA hatékonyabban csökkent a helyzetfrissítési költséget a Multi-Level HMIPv6 megoldáshoz képest, miközben alacsony szinten tartja a csomagkézbesítési költséget.*

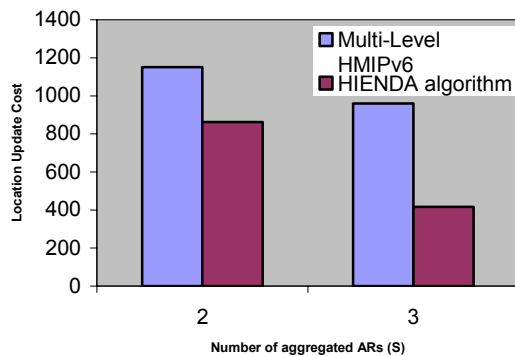
A jelzéseköltség arányos a különböző hierarchikus entitások közötti váltások számával, ezért úgy célszerű tervezni a proxy ügynök architektúra hierarchiáját, hogy az egy hierarchikus entitáson belül lévő AR-ek között maximális legyen a váltások valószínűsége. Így a jelzésüzenetek csak a hierarchia következő szintjéig jutnak fel a struktúrában, és nem a hierarchia csúcsáig. Ezzel összhangban a HIENDA algoritmus célja, hogy csökkentse a jelzéseköltséget azáltal, hogy csökkenteni azon AR váltások számát, amelyek egyidőben MAP váltások is. Azon AR-eket csoportosítja egy MAP tartományba, amelyek között a váltási ráta magas, így jelentősen csökkenti a MAP váltásokat. Erre a célra a Huffman algoritmus lett módosítva, amely minimalizálva a csomagok átlagos hopszámát csökkenteni tudja a helyzetfrissítő költséget.

A HIENDA algoritmus egy olyan valószínűségi fát állít elő, ahol az $E(L)$ átlagos távolság a levélcsomópontok és a gyökér csomópont között minimális, egy adott $P_H(h_i)$ AR váltási ráta eloszlásra és T számú levél csomópontra. Ezzel a csomagok átlagos hopszáma minimális lesz, ami minimális helyzetfrissítő költséget eredményez, egyidejűleg alacsony szinten tartva a csomagkézbesítési költséget a HIENDA algoritmus által generált MAP allokációkkal.

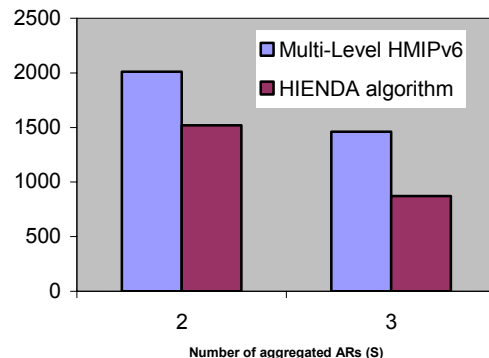
Nagyszámú szimulációs vizsgálatot végeztem el a HIENDA és a Multi-Level HMIPv6 [18] algoritmus hatékonyságának összehasonlítására, az 1. Téziscsoportban használt mobilitási szimulátor szolgáltatva a mobilitási adatbázist. A hierarchia első szintjének kialakítását (a cellák AR-ek alá történő csoportosítását) a LAFA és CEREAL algoritmussal végeztem el, szemléltetve, hogy a két tézisrendszer eredményei együtt alkalmazhatóak a hálózattervezés folyamán. Az így kapott AR cellacsoportok közötti váltási ráták szolgáltatták a HIENDA algoritmus bemenetét.

A vizsgálatok legfontosabb következtetései:

- $S = 2$ esetében (amikor az algoritmus minden lépésben két csomópontot vonunk össze) a HIENDA algoritmus helyzetfrissítési költség tekintetében felülmúlja a Multi-Level HMIPv6 megoldást (7.ábra).
- abban az esetben amikor három csomópont kerül összevonásra lépésenként, a HIENDA algoritmus feleannyi helyzetfrissítő üzenetet generál mint a Multi-Level HMIPv6 megoldás.
- az aggregált költségfüggvény esetében is a HIENDA túlszárnyalja a Multi-Level HMIPv6 teljesítményét az S függvényében vizsgálva (8.ábra).



7.ábra A helyzetfrissítési költség a két hierarchikus struktúrára



8.ábra Az aggregált költségfüggvény a két hierarchikus struktúrára

4.3 Információ terjesztési algoritmusok állandó kapcsolat nélküli mobil hálózatokban

3. Téziscsoport ([D1], 5.1-5.7; [B3], [C6], [C8], [C9], [C10], [C12]) *Kidolgoztam két új információ terjesztési algoritmust (MIOBIO, Multi-Message SBA) állandó kapcsolat nélküli mobil hálózatokra. Megmutattam, hogy mindkét új információterjesztési algoritmus felülmúlja a létező algoritmusokat robusztusság és járulékos forgalom tekintetében. Kidolgoztam egy természetes szelekción alapuló új megoldást a különböző információ terjesztési algoritmusok versenyztetésére és kooperációjára.*

Az 1. és 2. téziscsoportban hagyományos, állandó kapcsolatot fenntartó cellás mobil hálózatokra dolgoztam ki jelzésekcsökkentő tervezési algoritmusokat. A mobil hálózatoknak egy erőteljesen feltörekvő, forradalmian új vonulata az autonóm, önszerveződő, állandó kapcsolat nélküli hálózatok csoportja, amire szintén szerettem volna járulékos jelzémennyiség csökkentő megoldásokat adni. Így ebben a téziscsoportban egy merőben eltérő szemléletre és megoldásra volt szükségem, ugyanis teljesen más technikákkal lehet költségsökkentést elérni egy állandó kapcsolat nélküli ad hoc hálózatban, ahol központi infrastruktúra híján a mobil terminálok alkotják a hálózatot. A járulékos jelzémennyiséget ilyen hálózatokban a duplikált (a mobil csomópontoknak új információt nem hordozó) adatüzenetek és az információ terjesztési algoritmusok működéséhez szükséges jelzésüzenetek összesége adja.

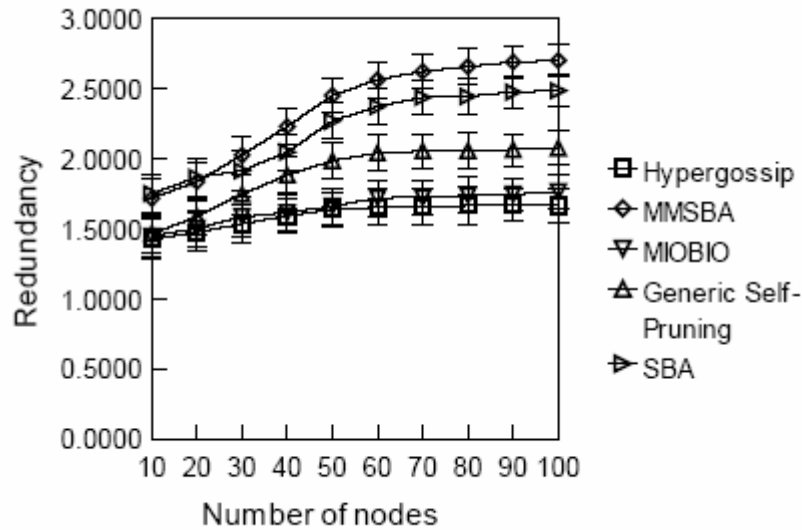
Tézis 3.1 *Kidolgoztam egy új információ terjesztési algoritmust (MIOBIO, Modified InfOrMation Dissemination Protocol for BIologically Inspired autonomic Networks and Services) állandó kapcsolat nélküli mobil hálózatokra. Megmutattam szimulációs*

vizsgálatokkal, hogy az MIOBIO produkálja a legkedvezőbb járulékos/hasznos üzenet arányt magas üzenetkérési ráta mellett a létező megoldások között.

A létező információterjesztési megoldások hiányosságainak kiküszöbölésére dolgoztam ki a MIOBIO algoritmust, amely három-utas kézfogást használ, hogy kiderítse, hogy a szomszédos mobil terminálok között melyik lehet érdekelt az általa birtokolt üzenetekben. Az algoritmus elsődleges célja, hogy csökkentse a szomszédos mobil terminálokra küldött duplikált vagy járulékos üzenetek számát, a legfontosabb előnye, hogy nem generál szükségtelen adatforgalmat, mint más megoldások. A három-utas kézfogás előnye, hogy nem kell minden alkalommal üzenetszórást indítani a mobil terminálnak, amikor új üzenetet kap. Az első két lépésben kis méretű jelzésüzeneteket használ a küldő és az általa hirdetett üzenetben érdekelt szomszédos terminálok (hirdető és kérés jelzésüzenetek), az adatüzenet kiküldése csak a harmadik lépésben történik meg, ha valamelyik szomszédos mobil terminál ténylegesen érdekelt az üzenetben. Így a kommunikáció által felhasznált energia jelentősen csökkenthető, mivel csak rövid jelzésüzeneteket terjesztenek a mobil terminálok, nagyobb méretű adatüzeneteket pedig csak abban az esetben, amikor ténylegesen szüksége van rá valamelyik terminálnak, ami nem jellemző a létező elárasztásos megoldásokra. A MIOBIO az adaptív periódikus elárasztást (Adaptive Periodic Flood, APF) használja a hirdetési jelzésüzenetek kiküldésére. Az APF egy ellenőrzött elárasztáson alapuló algoritmus, amely periódikusan sugározza ki az üzeneteket, viszont amennyiben duplikációt észlel, akkor növeli a periódusidőt. Így a MIOBIO ötvözi a két algoritmus előnyeit (APF és IOBIO), az APF-et használja a hirdetési jelzés üzenetek mennyiségének csökkentésére, míg az IOBIO-t az adatüzenetek redukálására.

A BIONETS szimulációs platform [19] segítségével nagyszámú szimulációs vizsgálatot végeztem a MIOBIO és a létező algoritmusok teljesítményének összehasonlítására, mely utóbbiak a szakirodalom összehasonlító vizsgálatai alapján a létező legjobb megoldások információterjesztésre egy állandó kapcsolat nélküli mobil környezetben: a Generic Self-Pruning (általános önmetsző algoritmus) [20], Scalable Broadcast Algorithm (skalázható önmetsző algoritmus, SBA) [21] és a Hypergossiping [22]. Nagyon fontos teljesítményleíró paraméter a járulékos és hasznos üzenetek aránya, ami tulajdonképpen azt mutatja meg, hogy átlagban hány duplikált üzenet szükséges egy hasznos üzenet célbajuttatásához.

A vizsgálatokból azt a következtetést lehet levonni, hogy a MIOBIO produkálja a legkedvezőbb járulékos/hasznos üzenet arányt a Hypergossiping mellett, viszont a Hypergossip messze a legrosszabbul teljesít az üzenetek elterjesztése területén, így velem szemben a MIOBIO eléri a mobil csomópontok jóval nagyobb hányadát alacsony járulékos üzenetmennyiség mellett (9.ábra).



9.ábra A járulékos és hasznos üzenetek aránya a mobil terminálok számának függvényében

Tézis 3.2 Kidolgoztam egy új információ terjesztési algoritmust (MMSBA, Multi-Message Scalable Broadcast), amely az SBA algoritmus módosított változata, több üzenetes működésre általánosítva. Megmutattam szimulációs vizsgálatokkal, hogy az MMSBA terjeszti el a leggyorsabban az információt a létező algoritmusok között, alacsony járulékos forgalommal.

Az MMSBA algoritmus egy önmetsző információ terjesztési megoldás, az önmetszéshez szüksége van a két ugrásnyira (hop) lévő szomszédok információira illetve az üzenetet utolsónak továbbító mobil terminál azonosítójára az adott üzenetben. Amikor a v mobil terminál egy szórt üzenetet kap az u mobil termináltól, kizárja az u szomszédait ($N(u)$) a saját $N(v)$ szomszédossági listájából és az így létrejövő $B = N(v) \setminus N(u)$ halmaz tartalmazza majd az adott üzenetben lehetségesen érdekelt mobil terminálokat. Ezen kívül használja még a véletlenszerűen megállapított késleltetés (Random Assessment Delay, RAD) mechanizmusát.

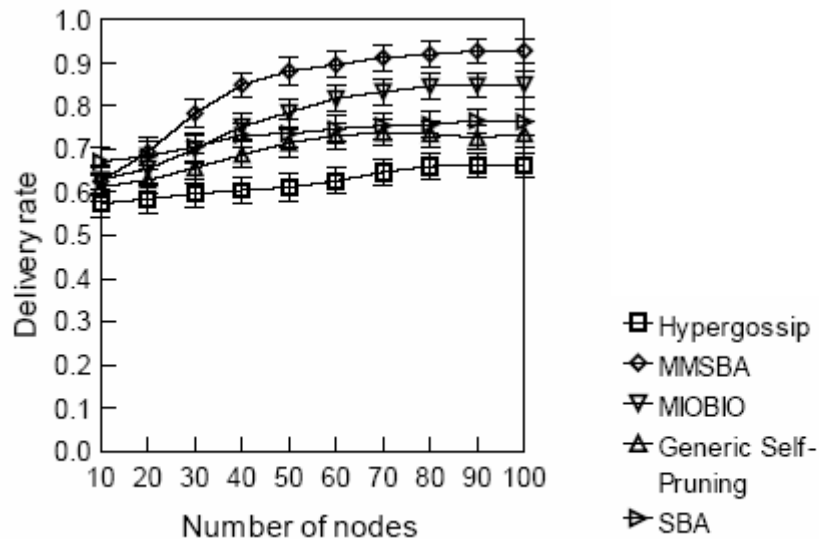
A RAD késleltetés maximális értéke a $\left(\frac{d_v}{d_{\max}}\right) \cdot T_{\max}$ képlettel számolható, ahol $d_v = |N(v)|$ és

d_{\max} pedig az $N(v)$ halmazban a legmagasabb fokszámmal rendelkező mobil csomópont fokszáma. Az adni készülő csomópontok ebből az intervallumból választhatnak, így biztosítva, hogy a nagyobb fokszámmal rendelkező (több szomszéddal körülvett) csomópontok előbb kezdjenek szórt adásba mint a kisebb fokszámúak.

Az elkülönülő csomópont csoportokkal nem számoló megoldások mint az eredeti SBA és a Generic Self-Pruning gyengén teljesítenek egy állandó kapcsolat nélküli mobil környezetben. Ezekben az algoritmusokban csak akkor kerül kiküldésre az adott környezetben az üzenet, amikor újonnan megérkezik. Ha az adóközvetben nincs több mobil csomópont amikor az utolsó üzenet kiküldésre kerül, akkor az üzenetterjesztés megáll. Viszont ha új mobil csomópont érkezik az adási körzetbe, nincs olyan esemény ami egy újabb üzenetkiküldést kezdeményezhetne. Az általam kidolgozott új algoritmus a Multi-Message SBA (MMSBA) nem csak egy új üzenet első vételénél indíthat RAD periódust, hanem minden olyan esemény folytán, ami megváltoztatja a helyi környezet információit. Minden alkalommal amikor HELLO üzenet érkezik az MMSBA frissíti a szomszédlistáját és ha az érdekelt csomópontok száma nagyobb nullánál, akkor RAD periódust indít. Így minden olyan csomópont amelyik HELLO üzenetet küld, felkerül az érdekelt szomszédok listájára, annak ellenére, hogy a szórt üzenet már eljutott hozzá korábban. Ez a jelenség nem volt jelen az eredeti SBA-ban, mivel a

csomópontok csak egyszer sugározzák ki az üzenetet, az érkezése után közvetlenül. Így az új megoldásban a csomópontok azon üzenetek listáját is csatolják a HELLO-hoz, amelyeket már más forrásból korábban megkapták. Ez egy visszacsatolást is jelent az MMSBA-nak, hogy a sugárzott üzenetei elvesztek-e az átvitel során, és így több üzenetet lehet lehet függetlenül kezelni, amire az előző változatban nem volt lehetőség.

Az MMSBA teljesítményét is a 3.1 tézisben már bemutatott algoritmusokéval hasonlítottam össze. Megmutattam, hogy az MMSBA adja a legjobb szolgáltatási lefedettséget, vagyis ő terjeszti el a legrövidebb időn belül az adott üzenetet a hálózatban, alacsony járulékos forgalommal (10.ábra).



10.ábra Az algoritmusok által nyújtott lefedettség

Tézis 3.3 Kidolgoztam egy természetes szelekción alapuló új megoldást a különböző információ terjesztési algoritmusok versenyztetésére és kooperációjára. Megmutattam, hogy a mobil csomópontok közötti kooperáció ösztönzésével javul a rendszer hasznos üzenet kézbesítési teljesítménye és csökken a hasznos üzenetek életkora.

Annak ellenére, hogy számos információterjesztési algoritmus létezik a szakirodalomban, a mobil terminálok által alkotott topológia olyan gyorsasággal változik, hogy nem lehetséges kiválasztani az elméletileg optimális algoritmust. Ezért egy olyan biológiai ihletésű megoldást dolgoztam ki, amely a természetes kiválasztódás jelenségét használja fel egy ilyen folyamatos változásban lévő hálózatban. Így nincs egy előre meghatározott algoritmus amelyik a hálózat egészére a legjobb teljesítményt nyújtja, hanem több, előzetesen hatékonynak bizonyult algoritmus versenyez folyamatosan egymással, így a legéletképesebb algoritmusok fognak elterjedni a hálózat különböző részein, annak függvényében, hogy mennyire felelnek meg az adott hálózati részen kialakult topológia jellemzőinek. Azon algoritmusok amelyek alkalmasabbak az adott környezetben való működésre, gyorsabban „szaporodnak” majd, mint a kevésbé alkalmasak. Így úgy választottam meg a jósági függvényt, hogy azon algoritmusok kapjanak nagyobb jósági pontszámot, amelyek jobb teljesítménnyel működnek az adott környezetben. Egy nagy előnye ennek a megoldásnak, hogy nincs központi jósági függvény kiértékelés, minden döntés helyi szinten születik meg az egyes mobil terminálokban, így megszűnik a teljesítmény visszajelzésekből származó járulékos forgalom.

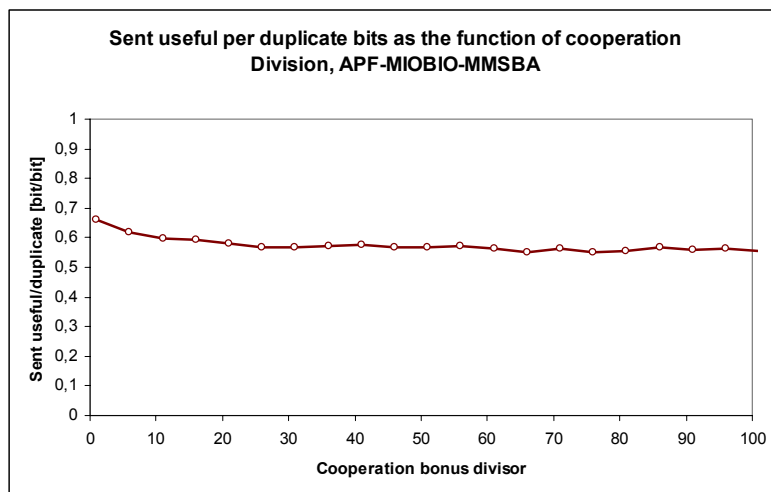
A 3.1 és 3.2 tézis eredményei alapján a leghatékonyabb információterjesztési algoritmusokat választottam ki (MIOBIO és MMSBA), hogy versenyezzenek egymással a természetes szelekción alapuló keretrendszerben. Az APF is szerepel mint versenyző algoritmus a MIOBIO és az MMSBA mellett, hogy meg lehessen vizsgálni egy ilyen „agresszív” algoritmus hatását a természetes szelekcióra, amikor más komplex és visszafogottabban információt terjesztő algoritmusokkal szembeül.

A kooperáció hatását vizsgáltam az algoritmusok teljesítményére szimulációs vizsgálatokkal a természetes szelekciós keretrendszerben. A vizsgálatok során a kooperáció foka lett változtatva, vagyis az ún. kooperációs bónusz arány, amely a küldött illetve a fogadott hasznos üzenetért kapott bónusz pontok aránya.

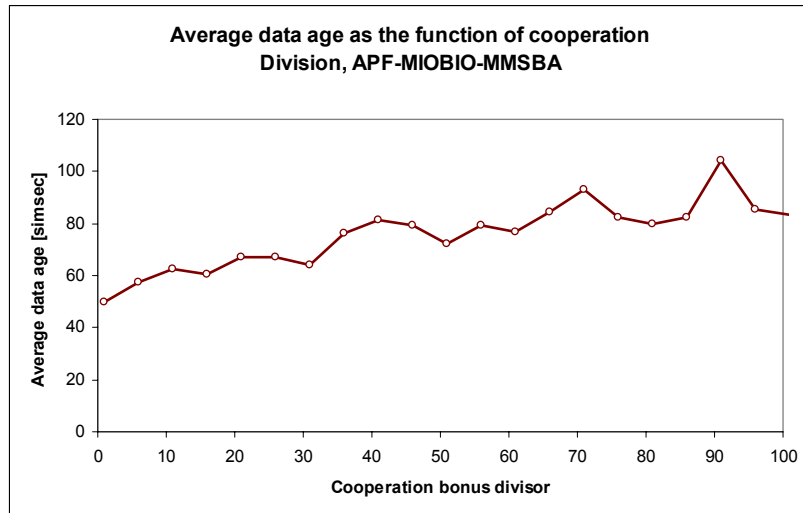
A vizsgálatok legfontosabb következtetései:

- a kooperáció fokának növelésével az egész rendszer teljesítőképessége nő (11.ábra, a vízszintes tengelyen a kooperációs bónusz arány növekedésével csökken a kooperációs hajlandóság)
- a kooperáció ösztönzésével az adatüzenetek átlagos “életkora” csökken, ami azt jelenti, hogy a felhasználók friss információkat kapnak, vagyis gyorsabban terjednek el a hálózatba újonnan kerülő információk (12.ábra).

Így lehet visszafogni a mohó, spammelő algoritmusokat, ugyanis egy nagy visszatartással rendelkező elárasztásos módszer még túlélhet egy ilyen szelekciós versenyben, de egy mohó megoldás nem képes rá.



11.ábra A küldött hasznos/duplikált adatmennyiség a kooperációs bónusz arány függvényében (APF/MIOBIO/MMSBA)



12.ábra Az adatüzenetek átlagos életkora a kooperációs bónusz arány függvényében (APF/MIOBIO/MMSBA)

5 Az új eredmények alkalmazása

Az 1.téziscsoportban ismertetett Location Area tervező megoldásaim jobb teljesítményt nyújtanak mint a létező Location Area tervezési algoritmusok. Mivel a manapság elterjedt cellás mobil hálózatok mindegyike körzet-alapú helyzetfrissítési mechanizmust használ, így fontos optimalizációs kérdést érintenek a megoldásaim. Az általam kidolgozott tervezési elvek használatával a mobil szolgáltatók úgy tudják konfigurálni a Location Area-kat, hogy alacsony szinten legyen tartható a jelzésforgalom. Ez különösen fontos a harmadik generációs és Long Term Evolution mobil rendszerekben, ugyanis a felhasználók számának növekedésével egyre kritikusabb lesz a helyzetfrissítési és paging eljárások által okozott jelzsmennyiség és késleltetés.

A körzet-alapú megoldások hiányossága, hogy nem foglalkoznak a felső hálózati hierarchiaszintek optimalizálásával, hogy milyen módon kerüljenek csoportosításra az útvonalválasztók a következő hierarchiaszinten illetve melyik szinten legyenek implementálva a mobilitási ügynökök. A 2.téziscsoportban pótlom ezt a hiányosságot, így a körzetek mellett lehetőség van a hierarchiaszintek megtervezésére is, így a teljes hálózati szegmens optimalizálható a mobilitáskezelés szempontjából.

A járulékos jelzések csökkentésének módját megvizsgáltam teljesen más típusú, forradalmian új hálózatokban is, egy olyan állandó kapcsolat nélküli hálózatban, ahol a mobil terminálok alkotják a hálózatot, bármilyen központi infrastruktúra nélkül. A 3.téziscsoportban bemutatott információ terjesztési algoritmusaim és a természetes szelekciós keretrendszer egy ilyen típusú hálózatban, a BIONETS-ben (BIologically inspired NETwork and Services environment) [19] kerül majd felhasználásra.

Irodalmi hivatkozások

- [1] B. L. Mark and Z. R. Zaidi, "Robust mobility tracking for cellular networks," in Proc. IEEE Int. Conf. Communications, vol. 1, New York, Apr.–May 2002, pp. 445–449.
- [2] A.Bar-Noy, I.Kessler, M.Sidi, "Mobile users: To update or not to update? ", Wireless Networks, vol.1., no.2, pp 175-185, July 1995
- [3] M. Mouly and M. B. Pautet, "The GSM system for mobile communication", Telecom Publishing, France, June 1992
- [4] I.F. Akyildiz, J. McNair, J. Ho, H. Uzunalioglu, and W. Wang, "Mobility Management in Current and Future Communications Networks", IEEE Network, July/August 1999
- [5] 3GPP, Boston, MA, USA and London, UK, "3GPP TS 23.012 V5.0.0, 3rd Generation Partnership Project, Technical Specification Group Core Network, Location Management Procedures (Release 5), March 2001
- [6] Zhang X., Castellanos J., Campbell A., "Design and performance of Mobile IP paging", ACM Mobile Networks and Applications, special issue on modeling analysis and simulation of wireless and mobile systems, vol.7, no.2, Mar. 2002
- [7] R. Thomas, H. Gilbert, and G. Mazziotto, "Influence of the moving of the mobile stations on the performance of a radio mobile cellular network", In Proceedings 3rd Nordic Seminar, Copenhagen, Denmark, September 1998
- [8] Chic-Lin I, Gregory P. Pollini, and Richard D. Gitlin, "Pcs mobility management using the reverse virtual call setup algorithm", IEEE/ACM Transactions on Networking, ToN, 5(1):13-24, February 1997.
- [9] C.U. Saraydar and C. Rose, "Location Area design using population and traffic data", In Proceedings of Conference on Information Science and Systems, CISS 1998, pp 739-744, Princeton, NJ, USA, March, 1998
- [10] P. R. Gondim, "Genetic algorithm and the location area partition problem in cellular networks", In Proceeding of the IEEE 46th Vehicular Technology Conference, VTC'96, pages 1835-1838, 1996
- [11] I. G. Tollis, "Optimal partitioning of cellular networks", In IEEE International Conference on Communications, ICC'96, volume 3, pages 1377-1381, 1996
- [12] Saraydar C. U., Kelly O., Rose C., "One-dimensional location area design, IEEE Trans. Veh. Technol.", vol. 49, pp. 1626-1633, Sept. 2000
- [13] Li J., Kameda H., Li K., "Optimal Dynamic Mobility Management for PCS Networks", IEEE/ACM Trans.Networking, vol.8, no.3, June 2000 , pp. 319-327
- [14] Bhattacharje P. S., Saha D., & Mukherjee A., "An Approach for Location Area Planning in a Personal Communication Services Network (PCSN)", IEEE Transactions on Wireless Communications, Vol. 3, No. 4, pp. 1176-1187, July, 2004
- [15] E. Cayirci And I.F. Akyildiz, "Optimal Location Area Design to Minimize Registration Signalling Traffic in Wireless Systems", IEEE Transactions on Mobile Computing, Vol.2, No.1, January-March 2003
- [16] S. Deering, R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC 2460, December 1998
- [17] H. Soliman, C. Castelluccia, K. El Malki, L. Bellier, Hierarchical Mobile IPv6 Mobility Management (HMIPv6), RFC 4140, August 2005
- [18] S. Pack, M. Nam, Y. Choi, "A Study On Optimal Hierarchy in Multi-Level Hierarchical Mobile IPv6 Networks", Globecom 2004, pp 1290-1294, Nov. 2004.
- [19] BIOlogically inspired NETwork and Services: www.bionets.eu
- [20] Jie Wu, Fei Dai, "Broadcasting in ad hoc networks based on self-pruning", In Proceedings of INFOCOM 2003, San Francisco, USA, April, 2003

- [21] Williams, B., and Camp, T., "Comparison of broadcasting techniques for mobile ad hoc networks", Proceedings of the 3rd ACM international symposium on Mobile ad hoc networking & computing, pp. 194 – 205, 2002.
- [22] Khelil, A., Marrón, P. J., Becker, C., and Rothermel, K., "Hypergossiping: A generalized broadcast strategy for mobile ad hoc networks", In Proceedings of the 2005 Conference on Communication in Distributed Systems (KiVS), February 2005.

Publikációk

Disszertáció:

- [D1] Vilmos Simon: „Mobility Management Algorithms for Reducing Signaling Overhead in Next Generation Mobile Networks”, Ph.D. Dissertation, Budapest University of Technology and Economics, September, 2008

Idegen nyelvű könyvfejezetek:

- [B1] V. Simon, Á. Huszák, S. Szabó, S. Imre: „QoS Provision in IP Based Mobile Networks”, Lectures Notes in Computer Science, Published at Springer, Vol. 2790, ISBN 978-3-540-40788-1, pp. 1137-1140, 2004
- [B2] Vilmos Simon, Sándor Imre: „Network Planning Algorithms for Optimizing Signalling Load in Mobile Networks”, Breakthrough Perspectives in Network and Data Communications Security, Design and Applications, Volume 1 of the Advances in Business Data Communications and Networking Series (ABDCN), IGI Global, edited by Indranil Bose, Chapter IX, pp. 149-163, 2008
- [B3] Daniele Miorandi, Iacopo Carreras, Francesco De Pellegrini, Imrich Chlamtac, Vilmos Simon, Endre Varga: „Chemical Relaying Protocols”, in Bio-inspired Computing and Communication Networks, Eds. Y. Xiao and F. Hu, published by Auerbach Publications, Taylor & Francis Group, CRC, ISBN-10: 1420080326, ISBN-13: 978-1420080322, 2008 (to appear)

Idegen nyelvű folyóirat cikkek:

- [J1] Vilmos Simon, Sándor Imre: "Location Area Design Algorithms for Minimizing Signalling Costs in Mobile Networks", International Journal of Business Data Communications and Networking (IJBDCN), IGI Global, Vol.3, No. 2, pp. 36-50, April-June 2007
- [J2] Vilmos Simon, Sándor Imre: „A Simulated Annealing Based Location Area Optimization in Next Generation Mobile Networks”, Journal of Mobile Information Systems, IOS Press, Vol. 3, No. 3,4, pp 221-232, ISSN 1574-017X, 2007
- [J3] Vilmos Simon, László Bokor, Sándor Imre: „A Hierarchical Network Design Solution for Mobile IPv6”, Journal of Mobile Multimedia (JMM), 2008

Magyar nyelvű folyóirat cikkek:

- [J4] Á. Huszák, T. Kiefer, V. Simon, G. L. Tilk, S. Imre, S. Szabó: "Handling of Mobility in IP based Networks", *Telecommunications*, Vol. LVIII, No.4., 2003. pp. 4-13.
- [J5] G. Péterfalvi, B. Pozsár, V. Simon, Á. Huszák, S. Imre: „Evolution of the Next Generation Mobile Networks”, *Telecommunications*, Vol. LX, No. 11., 2005. pp. 10-16.

Idegen nyelvű konferenci cikkek:

- [C1] V. Simon, Á. Huszák, S. Szabó: "Hierarchical Mobile IPv6 and Regional Registration Optimization", 45th International Symposium Electronics in Marine, ELMAR2003, 16-18 June 2003, Zadar, Croatia, ISSN 1334-2630, pp. 292-296
- [C2] V. Simon, S. Imre: "A Domain Forming Algorithm for Next Generation, IP Based Mobile Networks", SoftCOM2004, October 10-13, 2004, Split, Dubrovnik (Croatia), Venice (Italy), Published at FESB-Split, ISBN 953-6114-69-0, pp. 289-292
- [C3] V. Simon, S. Imre: „Location Area Design Algorithms for Reducing Signalling Overhead in Mobile Networks, The 3rd International Conference on Advances in Mobile Multimedia- MoMM 2005, 19-21 September, Kuala Lumpur, Malaysia
- [C4] Vilmos Simon, Tom Van Leeuwen, Gábor Péterfalvi, Sándor Imre, Ingrid Moerman: " A Network Planning Tool for Location Area Forming in Next Generation Mobile Access Networks", The 15th IST Mobile & Wireless Communications Summit 2006, 4-8 June 2006, Myconos, Greece
- [C5] Vilmos Simon, Sándor Imre: " A Paging Cost Constrained Location Area Planning in Next Generation Mobile Networks", The 4th International Conference on Advances in Mobile Multimedia- MoMM 2006, ISBN 3-85403-215-3, pp. 59-70, Yogyakarta, Indonesia, December 4-6. 2006.
- [C6] E. Varga, T. Csvórics, L. Bacsárdi, M. Bérces, V. Simon, S. Szabó: „Novel Information Dissemination Solutions in Biologically Inspired Networks”, The 9th International Conference on Telecommunications- ConTEL 2007, Zagreb, Croatia, June 13-15, 2007
- [C7] László Bokor, Vilmos Simon, István Dudás, Sándor Imre: „Anycast Subnet Optimization for Efficient IPv6 Mobility Management”, IEEE GIIS 2007, ISBN 1-4244-1375-3, pp. 187-190, Marrakesh, Morocco, 2-6 July, 2007
- [C8] L. Bacsárdi, M. Bérces, E. Varga, T. Csvórics, V. Simon, S. Szabó: „Strategies for Reducing Information Dissemination Overhead in Disconnected Networks”, The 16th IST Mobile and Wireless Communications Summit, ISBN 978-963- 8111-66-1, Budapest, Hungary, 1-5 July, 2007

- [C9] V. Simon, L. Baczárdi, M. Bérces, E. Varga, T. Csvorics, S. Szabó, S. Imre: „Overhead Reducing Information Dissemination Strategies for Opportunistic Communications”, IFIP/IEEE MWCN 2007, ISBN 978-1-4244-1719-3, pp.171- 75, Cork, Ireland, 19-21 September, 2007
- [C10] Vilmos Simon, László Baczárdi, Sándor Szabó, Daniele Miorandi: “BIONETS: a new vision of opportunistic networks”, IEEE WRECOM 2007, Rome, Italy, 1-2 October, 2007
- [C11] Vilmos Simon, László Bokor, Sándor Imre: “Novel Network Design Algorithm for Optimizing Hierarchical Mobile IPv6”, The 5th International Conference on Advances in Mobile Multimedia- MoMM 2007, ISBN 978-3-85403-230-4, pp. 123-133, Jakarta, Indonesia, 3-5 December, 2007
- [C12] Ahmad Al Hanbali, Mouhamad Ibrahim, Vilmos Simon, Endre Varga, Iacopo Carreras: „A Survey of Message Diffusion Protocols in Mobile Ad Hoc Networks”, in Proc. of Inter-Perf 2008, Athens, Greece, October 20-24, 2008 (to appear)