

Mobil csomagkapcsolt hálózatok adatforgalmazását  
támogató új szabályozó mechanizmusok  
tervezése és elemzése

*Miklós György*

Nagysebességű Hálózatok Laboratóriuma  
Távközlési és Médiainformatikai Tanszék  
Budapesti Műszaki és Gazdaságtudományi Egyetem

**Tézisfüzet**

Tudományos vezető:

Dr. Molnár Sándor  
Nagysebességű Hálózatok Laboratóriuma  
Távközlési és Médiainformatikai Tanszék  
Budapesti Műszaki és Gazdaságtudományi Egyetem

Budapest  
2004

# 1. Bevezetés

A távközlési iparban jelentős paradigmaváltásnak vagyunk tanúi. A hagyományos telefon-szolgáltatás mellett egyre többet használunk adatkommunikációra épülő alkalmazásokat is (pl. E-mail, WWW). Ezek az alkalmazások legtöbbször a TCP/IP protokollcsaládra épülnek. Az általuk generált adatforgalom sokkal gyorsabban növekszik, mint a telefonszolgáltatásból származó forgalom. Sokhelyütt az adatforgalom már meghaladja a telefonforgalmat. Arra számíthatunk, hogy a közeljövőben általánosságban is az adatforgalom fogja uralni a kommunikációs hálózatokat.

Műszaki szempontból ez azzal a következménnyel jár, hogy a hagyományos szolgáltatásokra (pl. beszédátvitel) optimalizált kommunikációs eszközöket, architektúrákat és protokollokat egyre többször váltják fel olyanok, amelyeket az adatforgalom igényeihez igazodva terveznek. Ez a disszertáció is ehhez a technológiai váltoáshoz járul hozzá.

Az adat-alkalmazások forgalmi jellemzői lényegesen különböznek a telefonhálózatok forgalmi jellemzőitől. A hagyományos távközlő rendszerek beszédhívásai statikus forgalmi terhelést jelentenek a hívás időtartama alatt. Ezzel szemben az adatforgalom az alkalmazás jellegétől függően dinamikus. Emiatt a telefonhálózatok dimenzionálási technikái, melyeket Erlang fejlesztett ki (lásd pl. [1]) nem használhatók az adathálózatokban [2]. Megfigyelték, hogy az adatforgalom több időskálát átfogó börsztösödést mutat [3].

Az 1. tézis az adatforgalom dinamikus jellegét veszi szemügyre, ahol egy új módszert javasolok a forgalom változékonyságának jellemzésére. Ehhez a telefonhálózatok dimenzionálása során használt *csúcsossági* (peakedness) mértéket veszem alapul, melyet általánosítok és kiterjesztek úgy, hogy az adatforgalom jellemzésére is alkalmas legyen. Megmutatom, hogy hogyan használható a csúcsossági mérték diszkrét időben, és tárgyalok több gyakorlati kérdést is, majd alkalmazom a csúcsossági mértéket több, mérésből származó adatsorra. Ezen kívül mutatok egy módszert, amelynek segítségével egy Markov típusú forgalommodellt illeszthetünk a méréshez a forgalom csúcsossági mértéke segítségével. Példákat mutatok arra, amikor egy ilyen modell jól visszaadja a forgalom csúcsosságát több időskálán is. Bizonyos esetekben azonban csak egy időskálán illeszthetünk ilyen modellt. Ez a forgalomban jelenlevő hosszú távú összefüggőséget (long range dependence) jelzi.

A disszertáció további tézisei a mobil adathálózati architektúrák különféle tervezési szempontjaival foglalkoznak. A mobil hálózatok térnyerése szintén jelentős paradigmaváltást jelent a kommunikációs iparban. A 2-4. tézisek a vezeték nélküli átvitel által felvetett tervezési kérdések közül vizsgálnak meg néhányat a cellás hálózati architektúra keretei közt. Ilyen cellás architektúrát használnak például az első, második és harmadik generációs mobil rendszerek éppúgy, mint a drótnélküli helyi hálózatok (Wireless Local Area Networks, WLANs). A disszertáció a WLAN architektúrát használja, de a bemutatott eredmények kiterjeszthetők más magán illetve nyilvános mobil hálózatra is. A vizsgált kérdések a vezeték nélküli csatorna kapcsán felmerülő adaptációs mechanizmusokra fókuszálnak.

A 2. tézis az adatkapcsolati rétegben elhelyezkedő újraküldési protokoll (automatic retransmission request, ARQ) problematikáját tárgyalja vezeték nélküli átvitel esetén. A dinamikus forgalom minta és a gyorsan változó hibák megkövetelik azt, hogy ennek a protokollnak is dinamikusnak kell lennie. A HIPERLAN/2 drótnélküli LAN architektúrában javasolok egy új megoldást. A javaslat dinamikus természete miatt új: a forgalmi- és hibamintákon alapulva adaptívan állapítja meg az ARQ visszacsatolás mennyiségét és az ARQ üzenetek tartalmát is. Szimulációs eredmények jelzik, hogy a dinamikus megoldás nagyobb teljesítményt nyújt, mint a statikus megoldások.

A 3. tézis egy bázisállomás erőforrás-elosztási döntései során felmerülő teljesítménymutatók közötti összefüggéseket elemzi. Mivel a forgalmi mintázatokat nem lehet előre megjósolni, a bázisállomásnak valós idejű erőforrás-elosztási döntéseket kell hoznia. A vezeték

nélküli csatorna miatt azonban az erőforrás-elosztási probléma alapvetően különbözik a vezetékes esethez képest. Az egy felhasználó számára lefoglalt erőforrás különbözhet attól az erőforrástól, amit a felhasználó ténylegesen kap, mivel rádiós közegre jellemző helyfüggő csatornahibák léphetnek fel. A hibákat ugyan javíthatjuk, de ez befolyásolja a csatorna teljes kihasználtságát és az erőforrás-elosztás igazságosságát (fairness). Bevezetek egy egyszerű kompenzációs mechanizmust a bázisállomásnál, és elemzem az egyes teljesítmény-mutatók közötti összefüggéseket, figyelembe véve a szállítási rétegben (TCP) és az adatkapcsolati rétegben használt újraküldési mechanizmusok hatásait is.

A 4. tézis kiegészíti a 3. tézis elemzését és egy konkrét erőforrás-elosztási módszer-családot javasol. Ebben a hibák kompenzálása mellett a rádiócsatorna hibáinak tulajdonságait is figyelembe vesszük és megpróbáljuk elkerülni a hibákat. Egy elosztott moduláris architektúrát mutatok be, amelynek jellemzője, hogy a csatornát az egyes terminálok követik nyomon és önállóan hoznak döntést a csatorna használatára vonatkozóan. A bázisállomásban levő ütemező a felhasználókat az erőforrások hatékony felhasználására ösztönzi. Javaslatot teszek egy ilyen központi ütemező algoritmusra és egy lehetséges felhasználói működésre, majd szimulációs és analitikus eszközökkel megmutatom, hogy a javasolt megoldás javíthatja mind a rendszer kihasználtságát, mind az erőforrás-elosztás igazságosságát.

Végül az 5. tézis egy újabb típusú dinamikus környezetet vizsgát: ad hoc típusú hálózatot, amelyben mindegyik csomópont mobil lehet. Egy új mechanizmust javasolok, amely lehetővé teszi a frekvenciaugrások szűrt spektrumú rádiók használatát ad hoc hálózati környezetben. Ehhez az eszközöknek dinamikusan kapcsolnak a különböző frekvenciaugrások csatornák között. A különféle csatorna-konfigurációk elemzését analitikus módszerrel és szimuláció segítségével is elvégzem.

## 2. Kutatási célkitűzések

A hagyományos távközlési hálózatokat a legtöbbször úgy tervezték, hogy jól meghatározott alkalmazási követelményeket elégítsenek ki. A modern drótnélküli csomagkapcsolt adathálózatok esetében azonban az alkalmazások igényeit igen nehéz pontosan előre jelezni, a forgalmi mintázat pedig borsztös és szintén megjósolhatatlan. Emiatt egyre többször dinamikus adaptív megoldások használatára van szükség. A disszertáció célja az, hogy olyan új szabályozó mechanizmusokat tervezzünk, amelyek ezt a technológiai trendet segítik.

A disszertáció célja

- olyan eljárások kutatása, amelyek a forgalom dinamikus (borsztös) természetét jellemzik;
- olyan megoldások kifejlesztése, amelyek a dinamikus (borsztös) forgalom drótnélküli csatorna feletti hatékony továbbítását teszik lehetővé;
- a bázisállomás erőforrás-elosztását jellemző teljesítmény-mutatók közötti összefüggések elemzése a rádiós közegen fellépő hibákra való tekintettel;
- új erőforrás-elosztó algoritmusok tervezése és elemzése, amelyek az erőforrás-elosztás igazságossága és a rendszer kihasználtsága közti kompromisszumot kontrollálni tudják, és amelyek ugyanakkor a praktikus megvalósítást is jól támogatják;
- új mechanizmusok tervezése és elemzése, amelyek lehetővé teszik mobil ad hoc hálózatok kialakítását a meglévő rádió technológia segítségével.

### 3. Módszertan

A disszertáció több új jellemzési eljárást, szabályozó mechanizmust és architektúrát javasol. Amikor a javaslatokat vizsgáltam, éppúgy használtam matematikai analízist és szimuláción alapuló teljesítményvizsgálatot, mint mérési sorozatok elemzését. Amikor csak lehetett, kombináltam ezeket a technikákat, elősegítve ezzel a probléma mélyebb megértését és egyben az eredmények ellenőrzését is.

Az 1. tézis matematikai analízisre épül, melynek segítségével új jellemzési és modellezési eszközöket fejlesztettem ki. Az elméletet más kutatók által készített mérésekből származó mintákon is kipróbáltam. Ezek a próbák számítógépes szimulációkat is magukba foglaltak.

A 2. tézisben szimulációs eredményekkel ellenőriztem a javasolt új mechanizmusokat. A 3-4. tézisekben az analízis és szimuláció kombinációját használtam. Úgy találtam, hogy a szimulációs munka alapvető jelentőségű az analízisben. Ennek fő oka az, hogy több szabályozó hurok, köztük a TCP egymásra hatását vizsgáltam. Ezen protokollok bonyolultsága és az adaptív mechanizmusok száma olyan nagy, hogy a teljes és pontos matematikai modellezést nem teszi lehetővé. Ehelyett a pontos csomag-szintű szimulációk segítségével támaszkodva végeztem a teljesítményvizsgálatot. Emellett a mélyebb megértés érdekében absztrakt eseteket vezettem be, melyeket matematikai módszerekkel is vizsgáltam. Megmutattam, hogy a matematikai modellek hogyan közelítik a szimulációs eredményeket. Bár ezen matematikai modellek a javasolt eljárásokból több tényezőt figyelmen kívül hagytak, mégis nagyon hasznosnak bizonyultak a valós esetek teljesítményének közelítésében.

Ugyanezt az utat követtem az 5. tézisben. Először egy egyszerű teljesítménymodellt adtam a versengési algoritmusra, melyet szimulációval támasztottam alá. Ezt néhány lehetséges FHC konfigurációra adott analitikus teljesítménymodell követ. Ezen közelítő formulák nem esnek pontosan egybe a tényleges protokoll teljesítményével, de tisztán mutatják a trendeket és magyarázatot is adnak rájuk. Ezután csomag szintű szimulációk segítségével ellenőriztem a protokollt, melynek segítségével pontos teljesítmény-elemzést is adtam. Az eredmények összhangban vannak a matematikai analízissel. A szimulációs eszközt a továbbiakban arra használtam, hogy olyan új eseteket vizsgáljak, amelyeket a matematikai modellek nem fednek le.

Úgy találtam, hogy a matematikai és szimulációs módszerek együttes használata nagyon konstruktív és hasznosnak bizonyul a kommunikációs architektúrák és mechanizmusok vizsgálata során. A gyakorlatban a munka a javaslatok, matematikai modellek és szimulációk közötti iterációkból állt. Minden ilyen iteráció mélyebb megértéshez vezetett, és megerősített az alkalmazott kutatási módszer erősségében.

### 4. Új eredmények

#### 1. Tézis. A börsztös forgalom csúcossági jellemzése [C1], [C2], [T1], [T2], [D2]<sup>1</sup>, [DA]

Az adatforgalom természetét vizsgáló méréseken alapuló tanulmányok [2, 3, 4] egyik fontos tanulsága az, hogy a forgalom több időskálán is börsztös tulajdonságú.

A forgalom börsztös jellegét leírhatjuk az ön hasonlóság elméletének segítségével. Az ön hasonlóság megfigyelése igen aktív kutatást eredményezett a forgalom fraktálok segítségével történő modellezésének irányába [3, 4]. Egyenlőre még nem világos, hogy milyen sikeresen használhatjuk az ön hasonlóság jellegét a gyakorlati mérnöki problémák megoldásának

<sup>1</sup>A [D $x$ ] jelölés a disszertáció  $x$ . fejezetére utal. [DA] a disszertáció A függelékére utal.

szempontjából, de egy dolog biztos: a nagysebességű hálózatokban a börsztösség tűnik az egyik legfontosabb forgalomjellemzőnek, mégis a börsztösség matematikai tulajdonságait még nem értjük eléggé. Ez a hiány motiválta ezt a tézist. Az egyik ígéretes börsztösségi mértéket, a csúcossági (peakedness) mértéket vizsgálom.

A legegyszerűbb börsztösségi mértékek csak a forgalom elsőrendű jellemzőit veszik figyelembe. A gyakorlatban a csúcs-átlag arány és a négyzetes szórási együttható a leggyakrabban használt elsőrendű mértékek [5, C1].

A forgalom másodrendű tulajdonságait kifejező mértékek bonyolultabbak. Az autokorrelációs függvény, a diszperziós indexek [6, 7] és az általánosított csúcossági mérték [8, 9] a legismertebbek ebben a kategóriában.

Ezen felül több börsztösségi mérték létezik, melyek különféle forgalmi koncepciókat használnak, pl. használhatunk börszt-hosszúsági mértékeket [5, 10] vagy leaky bucket paramétereket [11] a börsztösség jellemzésére. Az ön hasonlóság elméletéből ismert Hurst paramétert és más fraktál-paramétereket szintén használhatunk.

A forgalom *csúcossága* hasznos jellemzőnek bizonyult a blokkolási közelítésekben és a trónkölési elméletben [12]. Definíció szerint a forgalmat felajánljuk egy hipotetikus szervercsoportnak, ahol a foglalt szerverek szórásnégyzetének és várható értékének hányadosa adja a csúcossági mértéket, ahol a szerverek kiszolgálási ideje független és exponenciális eloszlású egy közös paraméter szerint.

Eckberg [8] kiterjesztette ezt a definíciót azzal, hogy tetszőleges kiszolgálási idő eloszlást megengedett és definiálta az *általánosított csúcosság* mértéket mint egy funkcionált, amely a tartásidő-eloszlást csúcossági értékre képzeli le. Egy adott kiegészítő tartásidő-eloszlásfüggvényhez ( $F^c(x) = P\{\text{tartásidő} > x\}$ ) Eckberg úgy definiálja a  $z\{F^c\}$  csúcossági funkcionált, mint a hipotetikus végtelen szervercsoport foglalt szervereinek szórásnégyzete osztva várható értékével, ahol a szerverek függetlenek és tartásidejük eloszlása  $F^c$  szerinti. Az általános definíció lehetőséget nyújt arra, hogy az érkezési folyamat változékonyságát egy adott kiszolgáló rendszerre vonatkoztatva jellemezzük.

### 1.1. Tézis. A csúcossági mérték diszkrét idejű általánosítása [C1], [C2], [T1], [T2], [D2.2.2-2.2.4], [DA]

Az általánosított csúcossági mérték elméletét kiterjesztettem diszkrét időre, és meghatároztam több forgalommodell csúcossági függvényét.

- Először jelöléseket és definíciókat vezetek be.  $w[i]$  jelöli az  $i$ . időszelvény alatti érkezők számát, ahol  $i = \dots -1, 0, 1, \dots$ . Feltételezzük, hogy  $w[i]$  stacionárius.  $w[t]$  első és második momentumát (melyek  $t$ -től függetlenek)  $m_1$ -el és  $m_2$ -vel jelölöm, az autokovariancia függvény pedig  $k[s] = \text{Cov}\{w[i], w[i+s]\} = k[-s]$ .

A kiszolgálási idő valószínűségi változó,  $T$ , szintén diszkrét, és a pozitív egészekben a  $t[1], t[2], \dots$  eloszlást veszi fel (nulla nem lehet). A kiszolgálási ráta  $\mu = 1/E\{T\}$ , a tartásidő kiegészítő eloszlásfüggvénye  $F^c[x]$ , az autokorrelációs függvény pedig  $\rho_{F^c}[x]$ .

A forgalmat felajánljuk egy végtelen szerver csoportnak, mely  $F^c[x]$  által meghatározott egymástól független és azonos eloszlású kiszolgálási idővel dolgozik. Minden egyes beérkezés egy új szervert foglal le. Az érkezési folyamat csúcossága a foglalt szerverek számának szórásnégyzete osztva a várható értékével:

$$z\{F^c\} = \frac{\text{Var}\{L[t]\}}{E\{L[t]\}} \quad (1)$$

ahol  $L[t]$  a foglalt szerverek száma a  $t$ . időszelvényben.

A definíció egy fontos módosítása az az eset, amikor a kiszolgálási idő csak az érkezési időtől függ, azaz a kiszolgálási idő azonos a  $t$ . időszelvény alatti összes,  $w[t]$  számú érkezésre. Ekkor ([13]-al összhangban) a csúcossági értéket *módosított* csúcosságnak nevezzük és  $\tilde{z}\{F^c\}$ -vel jelöljük.

- Megmutattam [DA], hogy

$$\tilde{z}\{F^c\} - z\{F^c\} = \left( \frac{m_2}{m_1} - 1 \right) (1 - \rho_{F^c}[0]\mu) \quad (2)$$

vagyis különbségük állandó (vö. [13], (35)). Ezen módosított definíció jelentősége abban áll, hogy megmutatja, hogyan kezeljük együtt a többszörös (batch) érkezéseket. Ezzel sok számítási kapacitást takarítunk meg abban esetben, amikor a csúcossági mértéket egy általános tartásidő esetén mérjük.

Az alábbiakban a csúcossági mérték eredeti definícióját fogjuk tovább használni (1).

- A legfontosabb eset diszkrét időben a geometriai eloszlású tartásidő:  $t[i] = \mu(1 - \mu)^{i-1}$ ,  $0 < \mu < 1$ .

Vezessük be a következő jelölést:

$$K[s] = \begin{cases} \frac{2}{m_1} k[s] & \text{ha } s > 0 \\ \frac{1}{m_1} k[0] & \text{ha } s = 0 \end{cases}$$

ennek z-transzformáltja pedig legyen  $K^*(\omega) = \sum_{s=0}^{\infty} K[s]\omega^s$ .

- Levezettem [DA], hogy az érkezési folyamat geometriai tartásidőre vonatkozó csúcossági függvénye

$$z_{\text{geo}}(\mu) = 1 + \frac{K^*(1 - \mu) - 1}{2 - \mu} \quad (3)$$

- Kimutattam, hogy az érkezések számára vonatkozó diszperziós index (IDC, index of dispersion for counts) és az érkezési folyamat geometriai tartásidőre vonatkozó csúcossága között kapcsolat van. Az IDC mértéket az érkezési folyamat különböző időskálákon vett változékonyságának jellemzésére használjuk, definíciója

$$I[t] = \frac{V[t]}{E[t]} = \frac{V[t]}{m_1 t} \quad (4)$$

ahol  $E[t]$  és  $V[t]$  az érkezések számának várható értéke és szórásnégyzete  $t$  egymást követő időszelvény alatt ( $t = 1, 2, \dots$ ).

Meghatároztam a kapcsolatot a csúcossági függvény és az IDC között [DA]:

$$z_{\text{geo}}(\mu) = 1 - \frac{\mu^2 \frac{dI^*(1-\mu)}{d\mu} + 1}{2 - \mu} \quad (5)$$

ahol  $I^*(\omega)$  az  $I[t]$  sorozat z-transzformáltja.

A következő aszimptotikus eredményeket vezettem le (5) alapján. Ha  $\lim_{s \rightarrow \infty} I[s]$  létezik és nem nulla, akkor:

$$z_{\text{geo}}(0) = \frac{\lim_{s \rightarrow \infty} I[s] + 1}{2} \quad (6)$$

$$z_{\text{geo}}(1) = I[1] = \frac{\text{Var}\{w[i]\}}{\text{E}\{w[i]\}}. \quad (7)$$

- Meghatároztam a csúcossági mértéket több forgalommodell esetére is [DA],[C2]. A Markov-modulált többszörös Bernoulli folyamat (Markov Modulated Batch Bernoulli Process, MMBBP) esetén a csúcossági mértéket a következőképpen számítjuk. Legyen  $\mathbf{P}$  illetve  $\mathbf{D}$  a moduláló Markov folyamat állapotátmeneti valószínűségeinek mátrixa és a stabil állapot eloszlásvektora ( $\mathbf{D}\mathbf{P}=\mathbf{D}$ ). Legyenek  $\mathbf{M}_1$  és  $\mathbf{M}_2$  diagonál mátrixok, amelyek a megfelelő állapotokbeli érkezések számának első és második momentumait adják meg. Legyen  $\mathbf{e}$  a csupa egyesekből álló vektor és jelölje  $\mathbf{I}$  az egységmátrixot.

Az érkezések átlagos száma  $m_1 = \mathbf{D}\mathbf{M}_1\mathbf{e}$ , második momentuma pedig  $m_2 = \mathbf{D}\mathbf{M}_2\mathbf{e}$ , az érkezési folyamat autokovariancia függvénye pedig  $k(i) = \mathbf{D}\mathbf{M}_1\mathbf{P}^i\mathbf{M}_1\mathbf{e} - m_1^2$ .

- Kimutattam [DA], hogy a csúcossági függvény

$$z_{\text{geo}}(\mu) = 1 + \frac{1}{2-\mu} \left( \frac{2(1-\mu)\mathbf{D}\mathbf{M}_1\mathbf{P}(\mathbf{I} - (1-\mu)\mathbf{P})^{-1}\mathbf{M}_1\mathbf{e} + m_2}{m_1} - 1 \right) - \frac{m_1}{\mu}. \quad (8)$$

- Levezettem a csúcossági mértéket néhány más fontos speciális esetben is: a többszörös érkezésű Bernoulli folyamatra (Batch Bernoulli Process, BBP), a Markov-modulált többszörös érkezésű folyamatra (Markov Modulated Bernoulli Process, MMBP) és a kapcsolt többszörös érkezésű Bernoulli folyamatra (Switched Batched Bernoulli Process, SBBP) [D2.2.4].
- Egy másik fontos forgalommodell a többszörös érkezésű felújítási folyamat (batch renewal process). Azért fontos ezt a modellt vizsgálni, mert képes a forgalom korrelációs struktúrájának leírására [14]. A diszkrét idejű többszörös érkezésű felújítási folyamat esetén a többszörös érkezések közti intervallumok független és azonos eloszlású valószínűségi változók, a többszörös érkezések száma pedig szintén független és azonos eloszlású, és független a köztes intervallumok hosszától is.

Használjuk a következő jelöléseket:  $a$  és  $b$  jelöli az intervallumok átlagos hosszát és a többszörös érkezések átlagos méretét. Az egy időszelvény alatti érkezések első és második momentuma  $m_1 = b/a$  és  $m_2 = m_1 b(C_b^2 + 1)$ , ahol  $C_b^2$  a többszörös érkezések méretének négyzetes szórás együtthatója (szórásnégyzet és várható érték aránya). Az érkezések közötti idő eloszlásának generátorfüggvénye  $A^*(\omega)$ . Levezettem [DA], hogy a geometriai tartásidő-eloszlás esetén a csúcossági mérték

$$z_{\text{geo}}(\mu) = 1 + \frac{1}{2-\mu} \left( \frac{1 + A^*(1-\mu)}{1 - A^*(1-\mu)} - b + \frac{m_2}{m_1} - 1 \right) - \frac{m_1}{\mu}. \quad (9)$$

## 1.2. Tézis. A csúcossági mértéken alapuló új modellillesztési eljárás [C2], [T1], [T2], [D2.2.5-2.3]

A forgalom csúcossági mértékére alapozva kidolgoztam egy új modellillesztési eljárást, és sikeresen alkalmaztam mért forgalmi mintákra.

A módszer a beérkező forgalom átlagos rátáját,  $m_1$ -et használja, valamint a csúcossági értéket  $\mu = 1$ -nél és három másik pontban:  $\mu_1, \mu_2$  és  $\mu_3$ -nál. A csúcossági görbére megszakított többszörös érkezésű Bernoulli folyamatot illeszttek (Interrupted Batch Bernoulli Process, IBBP): a moduláló Markov folyamat egyik állapotában az érkezéseknek általános (tetszőleges) eloszlása van, a másik állapotban nincsenek érkezések.

- A  $z(1) = m_2/m_1 - m_1$  összefüggésből megkapjuk  $m_2$ -t. Vezessük be  $\omega = 1 - \mu$ -t,  $\omega_i = 1 - \mu_i$ . Innen kiszámíthatjuk a következő értékeket (felhasználva azt, hogy

$$K^*(\omega_i) = (z_{\text{geo}}(\mu_i) - 1)(\omega_i + 1) + 1)$$

$$Y_i = Y(\omega_i) = m_1 \frac{1 - \omega_i}{2\omega_i} \left( K^*(\omega_i) + m_1 \frac{1 + \omega_i}{1 - \omega_i} - \frac{m_2}{m_1} \right) \quad (10)$$

Alkalmazva (8)-t az IBBP esetre, azt kapjuk, hogy

$$Y(\omega) = m_* - \frac{(m_* - m_1^2)(1 - \gamma)}{1 - \gamma\omega}. \quad (11)$$

Vezessük be a következő jelölést

$$\tilde{Y} = \frac{Y_1 - Y_2}{Y_2 - Y_3} \quad (12)$$

amelyből

$$\tilde{Y} = \left( \frac{\omega_2 - \omega_1}{\omega_3 - \omega_2} \right) \left( \frac{1 - \gamma\omega_3}{1 - \gamma\omega_1} \right) \quad (13)$$

és azt kapjuk, hogy

$$\gamma = \frac{\tilde{Y} \frac{\omega_3 - \omega_2}{\omega_2 - \omega_1} - 1}{\tilde{Y} \frac{\omega_3 - \omega_2}{\omega_2 - \omega_1} \omega_1 - \omega_3}. \quad (14)$$

Ha meghatároztuk  $\gamma$  értékét, kaphatunk egy becslést  $m_*$ -ra:

$$m_* = \frac{1}{3} \sum_{i=1}^3 \frac{Y_i - \frac{m_1^2(1-\gamma)}{1-\gamma\omega_i}}{1 - \frac{1-\gamma}{1-\gamma\omega_i}} \quad (15)$$

ahol a jobb oldalon az ismert  $\omega_i, Y_i$  értékek átlaga szerepel.

- Az IBBP-t a következők szerint illeszthetünk (emlékezzünk, hogy nincs érkezés a 2-es állapotban):

$$m_{1,(1)} = \frac{m_*}{m_1}, \alpha_2 = \frac{m_1(1-\gamma)}{m_{1,(1)}}, \alpha_1 = 1 - \gamma - \alpha_2, m_{1,(2)} = m_2 \frac{\alpha_1 + \alpha_2}{\alpha_2}. \quad (16)$$

Ha már megvan az 1-es állapotbeli érkezések számának első és második momentuma, használhatunk egy általánosított geometriai eloszlást a többszörös érkezések méretének eloszlására. Ebben az esetben nincs érkezés  $1 - \varphi$  valószínűséggel, és többszörös érkezés van  $\psi$  paraméterű geometriai eloszlás szerint. A momentumok ekkor

$$m_{1,(1)} = \varphi/\psi, \quad (17)$$

$$m_{1,(2)} = \varphi/\psi^2 \quad (18)$$

amelyekből megkaphatjuk  $\varphi$  és  $\psi$  értékét.

Ha lehetséges pontosan illeszteni egy IBBP modellt a  $\mu_i, z_{\text{geo}}(\mu_i)$  párokra, akkor a  $m_*$ -re vonatkozó egyenletben összeadott értékek azonosak. Ha nincs olyan IBBP, amelyik pontosan illeszkedne az adott csúcossági értékekhez, akkor  $m_*$  egy becslést ad és az illesztett IBBP modell csúcossági görbéje közelíti a  $\mu_i, z_{\text{geo}}(\mu_i)$  párokat.



- Több empirikus csúcossági mérést is végeztem valós forgalmi mintákon, számítógéppel szimulálva a hipotetikus kiszolgálást. Megmutattam, hogy a csúcossági görbe (geometriai kiszolgálási idő eloszlást használva és változtatva a kiszolgálási rátát) jellemzi a forgalom különböző időskálákon vett változékonyságát. Ez látható az MPEG videó mintákból számított csúcossági görbéken, ahol a keret-szekvencia és a képcsoport-szekvencia (Group of Pictures, GOP) elkülöníthető, és a film jellege láthatóan befolyásolja a csúcossági görbét. Sikeresen alkalmaztam a modellillesztési eljárást ezekre a mintákra.
- Adatforgalom mintákra is alkalmaztam a modellillesztési technikát. Az ATM és Ethernet minták esetében kimutattam, hogy a Markovi modell csak egy időskálán illeszthető. Ez azt jelzi, hogy a csúcossági mérték görbéje alkalmazható a hosszú távú összefüggőség (long range dependence) kimutatására is.

## 2. Tézis. Új dinamikus újraküldési protokoll tervezése [C5], [S1]-[S6], [P1], [P2], [P3], [D3]

A HIPERLAN/2 drótnélküli LAN rendszerben az ARQ (újraküldési) protokollnak dinamikusan változó mennyiségű adatra kell visszacsatolást nyújtania oly módon, hogy az ARQ információ az adatoktól függetlenül ütemezett kontrol PDU csomagokban kerül elküldésre. Ehhez új mechanizmusokra van szükség, melyek segítségével az ARQ üzeneteket ütemezni és azok tartalmát optimalizálni lehet. Ezen mechanizmusoknak akkor is működni kell, hogyha a végberendezésekben implementáció-függő késleltetések vannak.

Ezen igények kielégítésére a szelektív ismétléses ARQ egy újabb verzióját definiáltuk, az ún. részleges bittérképes protokollt (Selective Repeat ARQ with Partial Bitmaps, SRPB) [C5, 15]. A protokoll lehetővé teszi a csomagok (protocol data units, PDU-k) mennyiségének és az újraküldés időpontjának rugalmas megválasztását a küldési illetve vételi ablakon belül.

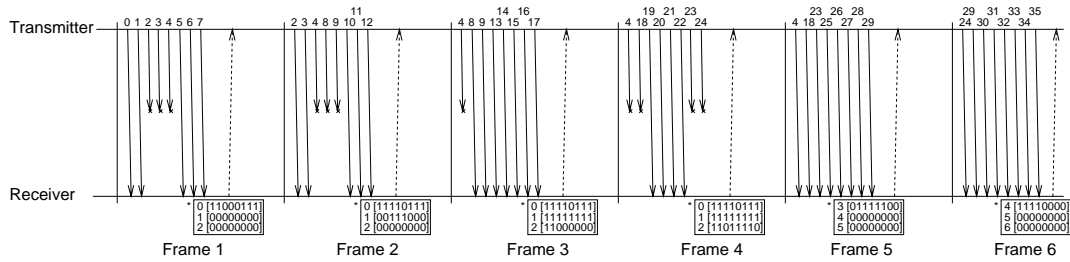
- Javasoltam egy rugalmasan kezelhető üzenet-formátumot a szelektív ARQ protokollhoz. A vevő egy bittérképet tart fenn, ami a vevő ablakba tartozó egyes PDU-k vételi állapotának felel meg. A vevő ezen bittérképnek segítségével ad visszacsatolást a küldő számára. A visszajelzés alapján a küldő újraküldi a hiányzó PDU-kat. Az ARQ üzenetben használt információs mezőket a 1. táblázat összegzi. Egy ARQ kontrol PDU-n belül a vevő három darab nyolc bites bittérkép részletet helyezhet el. Ezen kívül a vevő összegzően (kumulatív) is nyugtázhhatja a korábbi PDU-k vételét a CAI bit (Cumulative Ack Indication) beállításával. Amikor ez be van kapcsolva, akkor az első bittérkép blokk előtti összes PDU helyesen megérkezett.

Mező	Bitszám	Funkció
CAI	1	Kumulatív nyugta jelző
BMN 1	7	1. bittérkép száma
BMB 1	8	1. bittérkép blokk
BMN 2	5	2. bittérkép száma
BMB 2	8	2. bittérkép blokk
BMN 3	5	3. bittérkép száma
BMB 3	8	3. bittérkép blokk
ABIR	1	ARQ sávszélesség növelés kérés

1. táblázat. ARQ üzenet mezők

- Három egyszerű sémát javasoltam arra vonatkozólag, hogy a vevő hogyan válassza ki az ARQ üzenetben szereplő három bittérkép blokkot.

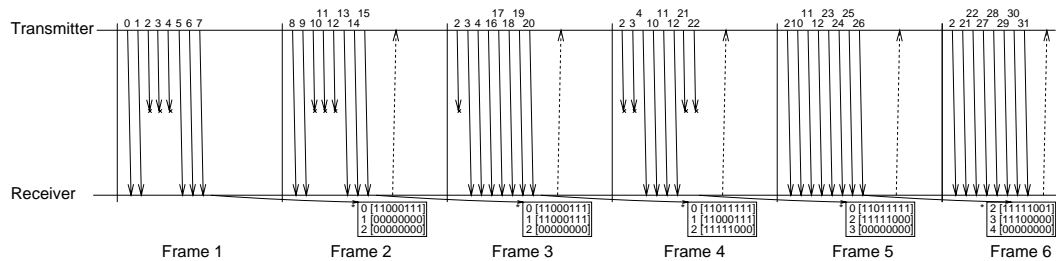
A. Az első és legegyszerűbb stratégia az, hogy a vevő az ablak aljától kezdődően folyamatosan kiválasztja az első három blokkot. Ez a legegyszerűbb stratégia, de nem használja ki az ARQ üzenet formátumban rejlő rugalmasságot. A stratégiát az 1. ábra illusztrálja egy példán keresztül.



1. ábra.

Az A. vevő ARQ stratégia. A vevő az első három bittérkép blokkot választja. (A bittérkép blokkokat szögletes zárójelben tüntettük fel, előttük a bittérkép száma áll.)

B. A második stratégia az, hogy a vevő az ablak aljától kezdődően választja ki a bittérkép blokkokat, de csak azokat, amelyekben van hiba is. Ezáltal a hibás PDU-k újraküldése gyorsabbá válik. A stratégiát a 2. ábra illusztrálja.

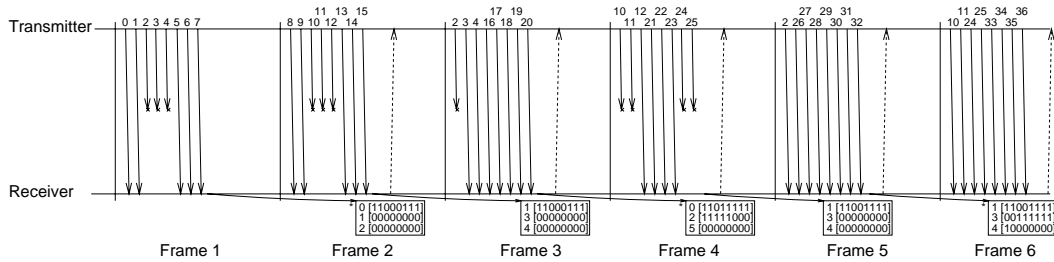


2. ábra.

A B. vevő ARQ stratégia. A vevő az első három, hibát tartalmazó bittérkép blokkot választja. A példában a vevőnél egy keretnyi késleltetés van az ARQ üzenet elküldéséig.

C. A harmadik stratégia arra az esetre vonatkozik, amikor az ARQ protokoll körülfordulási ideje nagyobb, mint egy keret. A körülfordulási idő az a minimális idő, ami szükséges az újraküldött csomag megérkezéséhez attól számítva, hogy a vevő elküldi a negatív nyugtát. Az ARQ protokoll körülfordulási ideje nagyobb lehet, mint egy keret, mert a vevőben vagy a küldőben a feldolgozás plusz időt vehet igénybe. Ebben a stratégiában a vevő az ablak aljáról kezdi jelezni a blokkokat, de csak azokat, amelyekben van hiba, azokat viszont nem, amelyeket az előző körülfordulási időn belül (beleértve az épp aktuális keretet is) már jelzett és azóta nem változtak. Ezen túl a vevő az első bittérkép blokkot választja legalább egyszer egy körülfordulási időn belül, hogy képes legyen összegző (kumulatív) módon nyugtázni az ablak előtti összes adatot.

Ezt a szabályt az motiválja, hogy elkerüljük ugyanazon blokkok többszöri jelzését egy körülfordulási időn belül, hiszen az újraküldés nem érkezhets meg hamarabb. Ehelyett az ARQ jelzési kapacitást arra használjuk fel, hogy újabb információt csatoljunk vissza a küldő felé. A stratégiát a 3. ábra illusztrálja.



3. ábra.

A C. vevő ARQ stratégia. A vevő nem használja azokat a bittérkép blokkokat, amelyek a megelőző körülfordulási időben már szerepeltek, és azóta nem változtak. A CAI (kumulatív nyugta) bit bekapcsolását csillag jelzi; minden körülfordulási időben legalább egyszer küldünk kumulatív nyugtát.

- A B stratégiát megvalósítottuk egy csomagszintű szimulátorban és összehasonlítottuk a PRIME ARQ protokollal [C5]. Az SRPB és a PRIME ARQ protokollokat ugyanolyan feltételek mellett vizsgáltuk: egy cellát szimuláltunk borsztös Markovi hibamodell esetén. A felajánlott forgalom növelésével a telítődési átvitel megközelítőleg 20%-al magasabb az SRPB protokoll esetében.
- Javasoltam egy eljárást, amely lehetővé teszi a bázisállomásban az ARQ kapacitás dinamikus beállítását egy egybités visszacsatolás segítségével. Ha több ARQ kontrol üzenetet allokálunk, akkor az újraküldések gyorsabbá válhatnak, de ez több erőforrást foglal le. Az ARQ kapacitás dinamikus hozzárendelését segíti az ABIR bit (ARQ Bandwidth Increase Request) az ARQ üzenetben. Amikor ez be van kapcsolva a terminál által küldött ARQ üzenetben, akkor ez jelzi a bázisállomás ütemezőjének, hogy a terminál növelni szeretné az ARQ kontrol csomagok számát.

Az adaptációs mechanizmus során a vevő az ABIR bit beállításával jelzi, ha nem tudja jelenteni az összes hiányzó PDU-t az ARQ üzenetben. Ebben az esetben az allokált ARQ kontrol csomagok száma eggyel nő, ellenkező esetben pedig eggyel csökken, de legalább egy ARQ kontrol csomag mindig jelen van. Ez gyors adaptációt tesz lehetővé, amely hatékonyabb a statikus allokációnál. A szimulációs eredmények azt mutatták, hogy ez a megoldás a rendszer átvitelét akár 100%-al is növelheti a hibák eloszlásától függően.

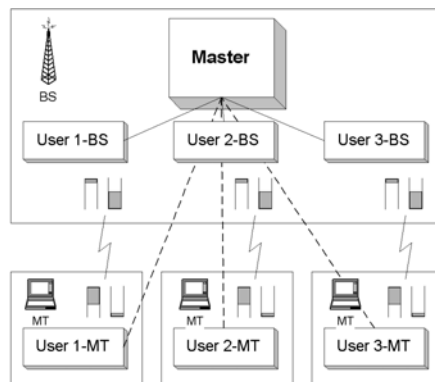
### 3. Tézis. A bázisállomás erőforrás-elosztási algoritmusának elemzése kihasználtság és igazságosság szempontjából [C3], [C4], [T3], [D4]

Az igazságos (fair) erőforrás-elosztás kérdését kimerítően tanulmányozták a vezetékes hálózatok esetében az úgynevezett folyadék fair sorbanállás segítségével. Ezen elv szerint definiálhatjuk az ideális fair erőforrás-elosztást több, közös összeköttetést használó folyam között [16, 17].

Ha a fair sorbanállási algoritmusokat és eredményeket megpróbáljuk a vezeték nélküli esetre alkalmazni, új problémákkal találkozunk. Mivel a csomagok elveszhetnek a rádiós interfészen, az egyes folyamatok számára lefoglalt erőforrás nem egyezik meg azzal, amint amit a folyamat ténylegesen kap. Ebből az is következik, hogy a folyamatok fair sorbanállás által nyújtott igazságosság megsérül. Mivel a csatornahiba-karakterisztikák helyfüggők lehetnek, az egyes folyamatok által kapott kiszolgálás nem lesz arányos a súlyokkal.

Ezeket a kérdéseket a 4. ábrán illusztrált erőforrás-elosztási architektúra keretei közt vizsgáljuk. Ez az architektúra összhangban van a HIPERLAN/2 drótnélküli LAN rendszerrel [18, 19, 20, 21] [D3.2], de maga az architektúra általános és más rendszerekben is alkalmazható.

A modellben felhasználónak nevezünk minden olyan egységet, amely adatokat küld illetve vesz. Az MT (mobil terminál) minden egyes kapcsolata egy-egy felhasználónak felel meg. A bázisállomás (BS, base station, melyet a HIPERLAN/2 terminológiában AP-nak, access point-nak is nevezünk) annyi felhasználóval van reprezentálva, amennyi kapcsolattal rendelkezik a terminálok felé. A BS tartalmazza a központi (master) ütemezőt, amely a kapacitás-elosztásért felelős. Az ütemező kimenete az egyes MAC keretek dinamikusan erőforrás-elosztása, ahol az elosztás egysége a fix hosszúságú rádió PDU.



4. ábra. Erőforrás-elosztási architektúra (user=felhasználó)

A központi (master) ütemező a felhasználók által küldött erőforrás-kérések alapján hozza meg az erőforrás-elosztási döntéseit. A bázisállomásbeli felhasználók esetében ezek a kérések csupán belső információátadást jelentenek. A terminálokbeli felhasználók az erőforrás-kéréseket kontrol információként elküldik a bázisállomásnak. Az erőforrás-kérések megadják azt az adatmennyiséget, ami a felhasználó bufferében elküldésre vár.

A központi ütemező minden keretre lefuttatja az ütemező algoritmust. Az algoritmus inputként használja fel a felhasználóktól származó erőforrás-kéréseket, és meghatározza, hány PDU kapacitást foglaljunk le az egyes felhasználóknak a következő keret során. Itt most nem foglalkozunk a foglalások egy kereten belüli sorrendjének meghatározásával. A hangsúly az egy kereten belüli foglalások mennyiségén van.

- Javasoltam egy egyszerű kompenzációs mechanizmust, melynek segítségével elemezhetjük a bázisállomás erőforrás-elosztását jellemző teljesítménymutatók közötti összefüggéseket. A központi kérdés itt az erőforrás-elosztás igazságossága és a rendszer kihasználtsága közötti kompromisszum. A központi (master) ütemező extra allokációt biztosít azon felhasználók részére, amelyeknél hibás küldések fordulnak elő. A hibák az ARQ protokoll segítségével kerülnek a központi ütemező tudtára azután, miután bekövetkeztek.

- A kompenzációs mechanizmust az SFQ (Start-time Fair Queueing) ütemező algoritmus kiegészítésével valósítottam meg. Egy új állapotváltozót vezetek be, a lemaradás (lag) változót minden egyes felhasználóra, melyet  $n_i$ -vel jelölök. Ez reprezentálja a normalizált kiszolgálás mennyiségét, amelyet a felhasználónak meg kell kapnia kompenzációként. Az elveszett erőforrásokért járó kompenzáció mennyiségét  $\beta_l \leq 1$  konstans jelöli.  $\beta_l = 1$  esetén az összes elveszett erőforrást kompenzáljuk,  $\beta_l = 0$  esetén egyáltalán nincs kompenzáció. Beszélhetünk negatív kompenzációról is, amikor  $\beta_l < 0$ .
- A javasolt kompenzációs mechanizmus a következő [D4.4-4.5]:

1. Minden felhasználóhoz hozzárendelünk egy kezdő és egy befejező címkét (start and finish tag),  $S_i$ -t és  $F_i$ -t, melyek a sor elején levő csomag virtuális kezdő és befejező idejét jelentik. Amikor egy új csomag belép a sor elejére  $t$  időben (ami azt jelenti, hogy egy csomagot kiszolgált a rendszer, vagy a felhasználó sora korábban üres volt), akkor az új értékeket a befejező címke régi értékéből,  $F_i'$ -ből számítjuk a következőképpen:

$$S_i \leftarrow \max\{F_i', v(t)\}, \quad (19)$$

ahol kezdetben  $F_i' = 0$ ,  $w_i$  jelenti az  $i$ . folyam súlyát,  $L$  az összes csomag hossza, és  $v(t)$  a virtuális idő  $t$  időpillanatban.

2. A virtuális idő kezdetben 0. Egy foglalt periódus alatt a virtuális idő  $t$  időpillanatban  $v(t)$ , amely nem más, mint a kiszolgálás alatt álló csomag kezdő címkéje. A foglalt periódus végén a virtuális idő a maximális befejező címke azon csomagok közül, amelyek kiszolgálása  $t$  időpillanatig megtörtént.
3. A csomagok kiszolgálása a kezdő címkék növekvő sorrendjében történik; egyező címkék esetén tetszőleges.
4. Amennyiben az ARQ protokoll jelentése szerint elveszett az  $i$ . felhasználó egy csomagja, akkor ennek lemaradása megnő az elveszett normalizált kiszolgálás  $\beta_l$ -szeresével:

$$n_i \leftarrow n_i + \beta_l \frac{L}{w_i} \quad (20)$$

$$n_i \leftarrow \min\{n_i, n_{max}\}, \quad n_i \leftarrow \max\{n_i, -n_{max}\}. \quad (21)$$

5. A befejező címkét így számítjuk:

$$l_c \leftarrow \min\{n_i, \beta_l \frac{L}{w_i}\}, \quad l_c \leftarrow \max\{l_c, -\beta_l \frac{L}{w_i}\}, \quad (22)$$

$$F_i \leftarrow S_i + \frac{L}{w_i} - l_c, \quad n_i \leftarrow n_i - l_c, \quad (23)$$

ahol  $l_c$  a normalizált kompenzáció egy csomag kiszolgálása alatt. Értéke  $\beta_l \frac{L}{w_i}$ , ha  $n_i$  pozitív, és  $-\beta_l \frac{L}{w_i}$ , ha  $n_i$  negatív.

- Megmutattam, hogy egy új folyam beengedésének kritériuma (a legrosszabb eset alapján, amikor minden folyamnak szüksége van kompenzációra) a következő:

$$\frac{1}{1 - \beta_l} \sum_{i \in \mathcal{U}} w_i \leq C. \quad (24)$$

- Szimulációk segítségével megvizsgáltam az erőforrás-elosztás igazságossága és kihasználtsága közti összefüggéseket a fenti kompenzációs mechanizmus alapján. A szimulációkban több folyam is szerepelt különböző súlyokkal. A folyamatok mindegyike egy mohó

TCP kapcsolatot jelentett. A szimulációhoz megvalósítottam egy egyszerűsített adatkapcsolati rétegbeli ARQ sémát, a 2. tézisben javasolt SRPB protokoll egyszerűsített változatát. Megvizsgáltam a rendszer kihasználtságát és az erőforrás-elosztás igazságosságát mind a lefoglalt sávszélesség, mind az adatkapcsolati, hálózati és alkalmazási réteg feletti átvitel esetében. Az eredmények megmutatták, hogy a kompenzációs mechanizmus valóban képes kontrollálni az igazságosság és a kihasználtság közötti kompromisszumot. A szimulációs eredmények ezen túlmenően a következő eredményeket hozták.

- Azt találtam, hogy az alkalmazási szint felett mért igazságosság jelentősen, akár egy nagyságrenddel is javulhat a kompenzációs mechanizmus hatására a rendszer kihasználtságának kis mértékű csökkenése árán [D4.6].
- A kihasználtság csökkenése csekély akkor, hogyha a csatornahibák függetlenek (1.8%), és növekszik, ahogy a veszteségmodell börsztössége nő (10.3%).
- Nagyobb kihasználtságot érhetünk el, ha negatív kompenzációt használunk (3.8%), vagy ha félig megbízható adatkapcsolati szintű ARQ mechanizmust alkalmazunk.
- Az allokáció igazságossága sokkal rosszabbá válhat és nem lehet a kompenzációs mechanizmussal javítani, ha félig-megbízható adatkapcsolati szintű ARQ mechanizmust használunk.

#### 4. Tézis. Elosztott architektúra a bázisállomás erőforrásának elosztására [C6], [C7], [D5]

##### 4.1. Tézis. Új elosztott erőforrás-elosztási architektúra [C6], [D5.1-5.6]

Javasoltam egy elosztott megközelítést alkalmazó új erőforrás-elosztási sémát. A bázisállomásban levő központi (master) ütemező nem rendelkezik semmilyen információval a rádiós csatorna állapotára vonatkozólag, pusztán kompenzálja a hibákat, miután azok bekövetkeztek. Ezt az architektúrát kiegészítjük azzal, hogy *a felhasználók ideiglenesen elhalasztják a küldést akkor, amikor a csatorna éppen rossz állapotban van*. A megközelítés tehát elosztott abban az értelemben, hogy a központi ütemező egy egyszerű kompenzációs módszert használ anélkül, hogy a drótnélküli csatorna állapotát figyelembe venné. A csatorna összes felhasználója maga felelős azért, hogy a csatorna jellemzőit megbecsülje és annak használatát optimalizálja azáltal, hogy meghatározza a küldés idejét. Amikor arra számítunk, hogy a csatorna rossz lesz, akkor a felhasználó késleltetheti az adatküldést addig, ameddig a mért tulajdonságok alapján a csatorna már valószínűleg helyreállt. Mindezt a központi ütemező elősegíti – de nem kontrollálja – azzal, hogy a felhasználók számára lehetővé teszi, hogy a jövőben részben visszakapják az általuk fel nem használt kapacitást.

Ennek a megközelítésnek több fontos előnye is van:

- A csatorna hibakarakteristikájáról nem kell semmilyen feltételezést tennünk, a központi ütemezőnek nem kell megjósolnia a drótnélküli csatorna állapotát.
- A javasolt mechanizmus moduláris megvalósítást tesz lehetővé és így nagyban leegyszerűsíti a központi ütemező tervezését.
- Lehetővé teszi a funkciók szétválasztását: a felhasználók által használt becslési és előrejelzési módszereket megváltoztathatjuk anélkül, hogy a központi ütemezőt befolyásolnánk.

Írányelveket adtam arra nézve, hogy miként lehet a vezetékes fair ütemezési algoritmusokat alkalmazni a bázisállomás központi ütemezőjeként. Ezeket az irányelveket alapul véve az SFQ (Start-time Fair Queueing) algoritmus alkalmazását és kiterjesztését javasoltam. Ez a kiterjesztés a 3.1. altézisbeli elveszett erőforrásokra vonatkozó kompenzációs mechanizmuson túl egy másik típusú kompenzációt is tartalmaz: a fel nem használt erőforrások után járó kompenzációt.

A központi ütemezőben használom a 3. tézisben bevezetett  $\beta_l$  konstanst, amellyel az elveszett erőforrásuk után járó kompenzációt szabályozom. Emlékezzünk, hogy  $\beta_l = 1$  esetén az elveszett erőforrás teljes mennyisége kompenzálásra kerül, míg  $\beta_l = 0$  esetén az elveszett erőforrások után nincs kompenzáció (most  $0 \leq \beta_l \leq 1$  értékeket használunk). Bevezetek egy új konstanst:  $\beta_u$  jellemzi a fel nem használt erőforrások után járó kompenzációt, ahol  $0 \leq \beta_u \leq 1$ . A  $\beta_l$  és  $\beta_u$  konstansokon kívül a kompenzáció sebességének hatását is vizsgálom, melyet a  $\gamma$  konstans jellemez,  $0 \leq \gamma \leq 1$ . Ez a konstans határozza meg az allokációk mennyiségének növelését akkor, amikor egy felhasználót éppen kompenzálunk.  $\gamma = 0$  esetén nincs kompenzáció, míg  $\gamma = 1$  esetén azonnali a kompenzáció, amikor is egy felhasználót kompenzálunk még mielőtt bárki más további erőforrásokat kaphatna. Mindegyik felhasználó mérések segítségével maga figyelheti saját csatornájának minőségét és ennek alapján dönthet, mennyi kapacitást kérjen.

A javasolt központi ütemező a 3.1. altézisbeli ütemező módosítása a következők szerint.

1. Minden felhasználóhoz hozzárendelünk egy kezdő és egy befejező címkét (start and finish tag),  $S_i$ -t és  $F_i$ -t, melyek a sor elején levő csomag virtuális kezdő és befejező idejét jelentik. Amikor egy új csomag belép a sor elejére  $t$  időben (ami azt jelenti, hogy egy csomagot kiszolgált a rendszer, vagy a felhasználó sora korábban üres volt), akkor az új értékeket a befejező címke régi értékéből,  $F_i'$ -ből számítjuk a következőképpen:

$$S_i \leftarrow \max\{F_i', v(t)\} \quad (25)$$

ahol kezdetben  $F_i' = 0$ ,  $w_i$  jelenti az  $i$ . folyam súlyát,  $L$  az összes csomag hossza, és  $v(t)$  a virtuális idő  $t$  időben.

2. A virtuális idő kezdetben 0. Egy foglalt periódus alatt a virtuális idő  $t$  időpillanatban  $v(t)$ , amely nem más, mint a kiszolgálás alatt álló csomag kezdő címkéje. A foglalt periódus végén a virtuális idő a maximális befejező címke értéke azon csomagok közül, amelyek kiszolgálása  $t$  időpillanatig megtörtént.
3. A csomagok kiszolgálása a kezdő címkék növekvő sorrendjében történik; egyező címkék esetén tetszőleges.
4. Amennyiben az ARQ protokoll jelentése szerint elveszett az  $i$ . felhasználó egy csomagja, akkor ennek lemaradása megnő az elveszett normalizált kiszolgálás  $\beta_l$ -szeresével:

$$n_i \leftarrow \min\left\{n_i + \beta_l \frac{L}{w_i}, n_{max}\right\}. \quad (26)$$

5. Amikor az  $i$ . felhasználó egy keret elején  $t$  időpontban új, ki nem elégített kéréssel jelenik meg, akkor a lemaradását az általa kihagyott normalizált kiszolgálás  $\beta_u$ -szorosára növeljük meg:

$$n_i \leftarrow \min\{n_i + \beta_u \max\{v(t) - F_i', 0\}, n_{max}\} \quad (27)$$

ahol  $v(t)$  a virtuális idő  $t$  időpillanatban, és  $F_i'$  az  $i$ . felhasználó befejező címkéje mielőtt megjelent a kérés. Mivel  $v(t)$ -t úgy is értelmezhetjük, mint az a normalizált fair kiszolgálás-mennyiség, amit  $t$  időpillanatig kaphatott, ezért a felhasználó  $v(t) - F_i'$  mennyiségű kiszolgálást mulasztott el, amíg nem kért erőforrást.

6. A befejező címkét így számítjuk:

$$l_c \leftarrow \min\{n_i, \gamma \frac{L}{w_i}\}, \quad (28)$$

$$n_i \leftarrow n_i - l_c, \quad (29)$$

$$F_i \leftarrow S_i + \frac{L}{w_i} - l_c, \quad (30)$$

ahol  $l_c$  a normalizált kompenzáció egy csomag kiszolgálása alatt.

Javasoltam egy olyan felhasználói viselkedési modellt, amely kiegészíti a bázisállomás ütemezőjét. A felhasználó vagy küld egy erőforrás-kérést a bufferben várakozó csomagok száma szerint, vagy nulla erőforrást kér, ezáltal a kiszolgálást egy későbbi keretre halasztja. A javasolt megoldás egy olyan módszerből áll, amelynek segítségével a felhasználó eldöntheti, mikor halassa el a kiszolgálást (vagyis mikor kérjen nulla erőforrást). Feltételezzük, hogy a felhasználó tud az ütemezési algoritmusról és a paraméter-beállításokról.

A kiszolgálás elhalasztása hasznos lehet egy mobil számára, mert a csatorna viselkedése tipikusan pozitívan korrelált, így amikor a felhasználó rossz csatornát érzékel, valószínű, hogy a csatorna továbbra is rossz marad bizonyos ideig.

A javasolt megoldásban [D5.5] a felhasználó egy csatornamodellt épít fel és megbecsüli ennek paramétereit. Ezáltal lehetővé válik a csatorna jövőbeli állapotának előrejelzése, és eldönthető, mikor és meddig halasszuk el az adást. A felhasznált modell AR(1) típusú, amely képes a csatorna korrelációs tulajdonságait is megőrizni. A javasolt megoldás küszöbököt használ a mért sikerarány értékére, és ennek alapján dönt a küldés elhalasztásáról, amikor a jövőbeli várható kompenzáció több, mint az azonnal felhasználható erőforrás.

Javasoltam egy egy-bites visszacsatolási mechanizmust is [D5.5.3]. A központi ütemező által szolgáltatott információ lehetővé teszi a felhasználók számára, hogy adaptívan állítsák be a küszöbököt. Ennek az adaptív érzékenység-beállításnak az az előnye, hogy a felhasználók maximális átvitelt érhetnek el azáltal, hogy az algoritmus érzékenységét pont akkorára állítják be, hogy épp a lehető legtöbb kompenzációt kérjenek.

Elvégeztem a javasolt architektúra szimuláción alapuló teljesítményvizsgálatát. Több szimulációt is végeztem, amelyek azt mutatják, hogy a kompenzációs mechanizmus hatására a rendszer összesített átvitele 10%-20%-al nő. A szimulációs eredmények alátámasztják az algoritmus működésének helyességét, és mutatják a paramétereiktől való függést is.

- Megmutattam, hogy az egy felhasználóra vonatkozó átviteli teljesítménynek van egy maximuma az algoritmus érzékenységének függvényében. Ha a felhasználó túl ritkán halasztja el az erőforrás-kérést, akkor nem elég hatékonyan kerül el a rossz csatorna-állapotokat. Ha a felhasználó túl gyakran halasztja el az erőforrás-kérést, akkor a kompenzációs mechanizmus telítődik és a felhasználó nem kapja meg a szükséges kompenzációt. Ebből az következik, hogy van egy optimális érzékenysége a felhasználói viselkedés algoritmusának. (Az egyik példa esetében ez akkor van, amikor a kérések elhalasztásának aránya 0.25.) Ezt az optimális pontot helyi mohó felhasználói viselkedésnek nevezzük, mivel a felhasználó saját szempontjából ez optimalizálja az átvitelt.
- Megmutattam, hogy az adaptív érzékenység-beállítás esetén gyakrabban lehet a kéréseket elhalasztani (ennek aránya 0.05-ről 0.35-re nő egy bizonyos paraméter-beállítás mellett).
- Megmutattam, hogy a javasolt architektúra mind a börsztös, mind a független hibájú csatornával rendelkező felhasználók átvitelét növelheti. (A szimulációkban 10%-20%



közötti javulást értünk el.) A börsztös csatornahibák esetében a felhasználói algoritmus kihasználhatja a rossz állapotok előrejelzésének lehetőségét. A független hibák esetében a javulás abból származik, hogy más felhasználók hatékonyabban használják a csatornát, valamint abból, hogy az elveszett erőforrások után kompenzáció jár.

#### 4.2. Tézis. A rendszer teljesítményének analitikus közelítése [C6], [C7], [D5.7]

Javasoltam egy analitikusan kezelhető approximációt, amellyel a rendszer teljesítménye leírható.

- Javasoltam két absztrakt felhasználói viselkedésmódot, az „algoritmus nélküli” és az „ideális algoritmus” eseteket. Az „algoritmus nélküli” eset egyszerűen az, amikor a felhasználó egyáltalán nem halasztja el az erőforrás-kéréseket. Az „ideális algoritmus” eset olyan, hogy anélkül, hogy ténylegesen el kellene halasztani a kihasználást, a sikerarány nem kisebb, mint amit el lehet érni bármilyen késleltetett kiszolgálással. Egy ilyen algoritmust természetesen lehetetlen megvalósítani vagy akár megközelíteni, de értékes absztrakcióval szolgál a rendszer analíziséhez.
- Analitikusan levezettem a lefoglalt és alkalmazás-szintű felhasználónkénti átviteli értékeket, a rendszer-szintű kihasználtságot és az igazságossági mértéket a következők szerint.

Jelöljük  $\Pi_i$ -vel az  $i$ . felhasználó sikerarányát. A kompenzációt az  $i$ . felhasználó súlyának módosításával modellezhetjük a következőképpen [C7]:

$$w'_i = \frac{1}{1 - \min\{\gamma, \beta_i(1 - \Pi_i)\}} w_i. \quad (31)$$

A rendszer teljes kapacitása,  $C$  a (31). egyenlet szerint oszlik el az egyes felhasználók között. Vezessük be a következő jelölést:  $W' = \sum_i w'_i$ , ezután (mohó felhasználókat feltételezve) az allokáció és átvitel az  $i$ . felhasználó esetében

$$a_i = \frac{w'_i}{W'} C \text{ és } g_i = \Pi_i a_i. \quad (32)$$

A következő metrikák segítségével jellemzem a rendszer teljesítményét: a rendszer kihasználtsága  $U = \sum_i g_i / C$ , az alkalmazás szintű átvitel eloszlásának igazságossága  $V_a = \sigma\{a_i/w_i\} / E\{a_i/w_i\}$ , vagyis a normalizált átvitel és lefoglalt értékek négyzetes szórás együtthatója.

Egy adott csatorna esetén a következő sikerarányokat használhatjuk:  $\Pi_{i,avg}$  a csatorna átlagos sikeraránya, és  $\Pi_{i,max}$  a maximális sikerarány, amit elérhetünk bármifajta felhasználói viselkedésmóddal. Egy ilyen felső korlátot értelmezhetünk úgy például egy Markovi csatorna esetében, mint a sikerarányt a legjobb állapotban. A  $\Pi_{i,avg}$  és  $\Pi_{i,max}$  sikerarányok használatával megkaphatjuk a kihasználtsági és igazságossági metrikákat az „algoritmus nélküli” és az „ideális algoritmus” esetekben.

- Összehasonlítottam az eredményeket a szimulációkkal és megmutattam, hogy az absztrakt analízis közelíti a rendszer teljesítményét. Ez a fenti analízis érvényességét támasztja alá.

Ezen teljesítménymetrikák segítségével megmutattam, hogy a javasolt architektúra javíthatja a rendszer kihasználtságot és ugyanakkor javíthatja az erőforrás-elosztás igazságosságát is. Ez azért lehetséges, mert a javasolt architektúra az erőforrások hatékony kihasználtságát ösztönzi.

## 5. Tézis. Új módszer több frekvenciaugrásos csatorna összekötésére ad hoc hálózatokban [J2], [P6], [P7], [D6]

A fenti tézisekben olyan csomagkapcsolt hálózatokat vizsgáltam, ahol jelen volt egy vezeték nélküli gerinchálózat. A drótnélküli technológiák segítségével lehetséges olyan hálózatokat is tervezni, ahol nincs ilyen előre installált infrastruktúra. Ezeket a hálózatokat ad hoc hálózatoknak nevezzük. Az IETF MANET (Mobile Ad hoc Networking) munkacsoportja ezen hálózatok protokoll kérdéseinek megoldására szerveződött. A MANET munkacsoport elképzelése [22] az, hogy az útvonalválasztási funkciót beépítjük a mobil eszközökbe, és ezáltal robusztus és hatékony mobil hálózatok jönnek létre. Ezek dinamikus, néha gyorsan változó hálózatok, amelyek több csomóponton keresztüli adattovábbítást biztosítanak, és tipikusan kis sávzélességű rádiós kapcsolatokból állnak.

### 5.1. Tézis. Kommunikáció több frekvenciaugrásos csatorna esetén [J2], [D6.3]

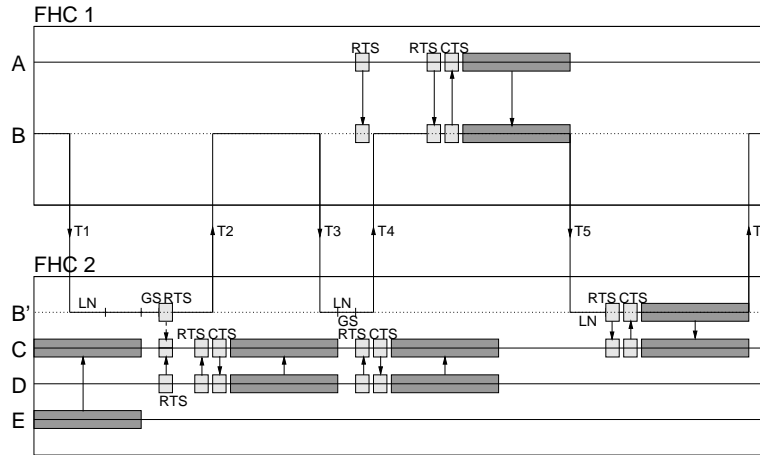
Javasoltam egy új eljárást, a több frekvenciaugrásos csatorna kommunikációt (MFHC, Multiple Frequency Hopping Channel), melynek segítségével több frekvenciaugrásos csatorna esetén is összekötött ad hoc hálózatot hozhatunk létre. Az eljárás az otthoni FHC (frekvenciaugrásos csatorna, frequency hopping channel) definícióját használja. Minden egyes eszköznek, amely az ad hoc hálózatban részt vesz, van egy otthoni FHC-ja, amely meghatározza azt, hogy melyik frekvenciaugrásos csatornát követi akkor, amikor nem küld egy másik FHC-n. Ahhoz, hogy valakinek adatot küldjön egy eszköz, szükséges, hogy átkapcsoljon a cél eszköz otthoni FHC-jára, figyelje a csatornát és feloldja a versengést a CSMA/CA egy adaptált változatát használva. Más rendszerekhez képest ennek az architektúrának az az előnye, hogy használhatjuk a Bluetooth-hoz [23] hasonló olcsó frekvenciaugrásos rádió technológiákat egy egyszerű kapcsolatmentes igény szerinti erőforrás foglalásra épülő rendszerben, mint amilyen például a IEEE 802.11 [24] szabvány. Lehetővé teszi az összes eszköz egy ad hoc hálózatba való összekötését hasonlóan a HRMA (Hop-Reservation Multiple Access, [25]) és a HFITF protokollokhoz (High Frequency Intra Task Force Communication Network, [26]), mégis szükség van hálózat-szintű szinkronizációs mechanizmusra. Egyik korábbi rendszer sem rendelkezik mindezekkel az előnyökkel.

Egy FHC-n belül a csatornahozzáférés az IEEE 802.11-ben is használt CSMA/CA megközelítésen [24] alapul. Ez azt jelenti, hogy az a csomópont, amelynek van az FHC-n egy elküldendő csomagja, először vár, amíg a csatorna szabaddá válik legalább egy minimális időre (GS, guard space). A kommunikáció fix időrehatóknál kezdődhet. (Itt nem határozzuk meg az időreható hosszát, hanem ezt használjuk időegységként.) Amikor több, mint egy állomás ad egyszerre egy csatornán, ütközés lép fel, melyet a versengési mechanizmus old fel a következőképpen. Minden állomás nyilvántart egy versengési ablakot,  $CW$ -t, és választ egy véletlen backoff értéket,  $B$ -t a  $[0, CW - 1]$  intervallumból. Minden egyes időrehatóban, amikor a csatornát szabadnak érzékeljük,  $B$  értékét eggyel csökkentjük, amennyiben pozitív. Egy csomópont akkor küldhet, amikor  $B$  értéke eléri a nullát. Ha a küldés sikeres, a  $CW$  értéke  $CW_{min}$ -re inicializálódik. Ha a küldés sikertelen,  $CW$  értékét duplázzuk, amíg el nem éri  $CW_{max}$ -ot. Ez a módszer biztosítja, hogy az ütközéseket egy vagy több lépcsőben biztosan feloldjuk.

Minden egyes csomagküldést megelőz egy RTS-CTS üzenetváltás, mint a 802.11 protokoll esetében. Ez kezeli a rejtett terminál problémát (a célállomás csomagokat kap egy olyan állomástól, amelyet a forrás nem hall), és emellett csökkenti a versengés költségét hosszú csomagok esetében. Az RTS-CTS üzenetváltás lehetőséget biztosít arra is, hogy a két állomás megállapodjon az adatküldés paramétereiben.

Ezt a sémát kiegészítettem több FHC esetére, amint azt a 5. ábrán látható példa mutatja. Bár egy csomópont átkapcsolhat egy másik FHC-ra, minden eszköznek van egy

saját otthoni FHC csatornája is. Az ábra két FHC-t mutat, ahol az 1. FHC az A és B eszközök otthoni FHC-ja, míg a 2. FHC a C, D és E eszközök otthoni FHC-ja. Egy csomópont időlegesen elhagyhatja az otthoni FHC-t, amint azt B is teszi, hogy átkerüljön FHC 2-be (B'), de visszatér az eredeti csatornához, amint befejezte a versengést vagy adatküldést. Ahhoz, hogy egy eszköz adatküldést kezdeményezzen, át kell kapcsolnia a célállomás csatornájára és várnia kell, amíg a célállomás és a csatorna szabad lesz.



5. ábra. Példa a több frekvenciaugrásos csatornán való kommunikációra

Amikor a cél csomópont FHC-ja különbözik a forrásétól, akkor a versengés során a forrásnak időszakosan kapcsolnia kell a forrás és a cél FHC között. A 5. ábrán a B csomópont csomagot akar küldeni C-nek a 2. FHC-ban. Először átkapcsol a 2. FHC-ba (B' csomópont a T1 jelű átmenet után) és egy rögzített ideig (ezt az ábrán az LN, listen jelzi) figyel a csatornán. Erre azért van szükség, hogy a csatornára szinkronizálódjon és meghatározza, hogy van-e éppen csomagküldés. Ha éppen adatküldés van folyamatban, mint a példában is, akkor B-nek várnia kell, amíg a küldésnek vége lesz (figyelembe véve a GS védőidőt is), mielőtt elküldi az RTS-t. Az ábrán D is küldeni akar C-nek, és miután összeütközik B-vel az első RTS küldés során, megnyeri a versengést a második lépésben. B ezt akkor veszi észre, amikor hallja az RTS-t a D csomóponttól, és vár addig, amíg a csomagküldésnek vége nem lesz. Erre az időre visszakapcsol az otthoni FHC-jára (T2 átmenet). Ahhoz, hogy meghatározza, mikor próbálhatja meg az új RTS küldést, B az RTS-beli csomaghossz információból megbecsüli az adatküldés hosszát (ez a CTS csomagban is szerepel). B csomópont visszakapcsol a 2. FHC-ra (T3 átmenet) úgy, hogy még LN időt figyel azelőtt, hogy a backoff számláló értéke elérné a nullát. Az ábrán D csomópont ismét megnyeri a versengést, és B visszakapcsol az 1. FHC-ra (T4 átmenet). Közben A csomópont adatküldést kezdeményez B csomópont felé, amely sikertelen, mert B nem volt ott abban az időben. Az RTS újraküldésre kerül, majd adatküldés követi. Ez késlelteti B csomópont átkapcsolását a 2. FHC-ra. Azonban amikor az 1. FHC-beli adatküldésnek vége van, B csomópont azonnal átkapcsol a 2. FHC-ra (T5 átmenet). Miután LN periódus eltelt és a 2. FHC-t üresnek érzékeli a csomópontot, B elküldi az RTS-t, amelyet a vevő ezúttal sikeresen venni tud, lehetővé téve az ezt követő csomagküldést. Amikor ennek vége, B visszakapcsol az otthoni FHC-jára (T6 átmenet).

A szomszéd csomópontok címét és otthoni FHC-ját a szomszédság-felfedező mechanizmus segítségével tudjuk meg. Ez alapulhat statikus konfiguráción, vagy használhat a csomópontok által küldött beacon csomagokat [27]. Ilyen csomagokat küldhetünk egy dedikált

frekvencián, vagy egy speciális frekvenciaugrási sorozaton. Ezen kívül a beacon csomagokat arra is használhatjuk, hogy az egyes FHC-kat időben szinkronizáljuk [24]. Ugyan szükség van arra, hogy az egymással kommunikáló eszközöket időben jól szinkronizáljuk, az MFHC nem igényel hálózati szintű szinkronizációs mechanizmust.

A javasolt MFHC kommunikációs sémát megvalósítottam egy csomagszintű szimulátorban, és ezzel ellenőriztem is azt. Az implementáció bebizonyította, hogy a megközelítés megvalósítható, és a szimuláció ugyanakkor a következő altézisbeli elemzéshez is hasznos eszközként szolgált.

## 5.2. Tézis. Az MFHC kommunikáció teljesítményanalízise [J2], [D6.4-6.5]

Az MFHC teljesítmény-jellemzőit analitikus és szimulációs módszerekkel vizsgáltam. Ennek keretében összehasonlítottam az MFHC egyik szélsőséges esetét, ahol minden eszköznek külön FHC-ja van, és a referencia esetet, ahol az egész ad hoc hálózat ugyanazt az FHC-t használja. Vizsgáltam továbbá azt az esetet is, amikor az ad hoc hálózat egyes részei egy részben zárt kommunikációs csoportot alkotnak. Ez azt jelenti, hogy egy csoport tagjai nagyrészt a csoport többi tagjával kommunikálnak, és csak ritkán az ad hoc hálózat más csomópontjaival. Ez az eset tipikus lehet több valóságos ad hoc hálózatban.

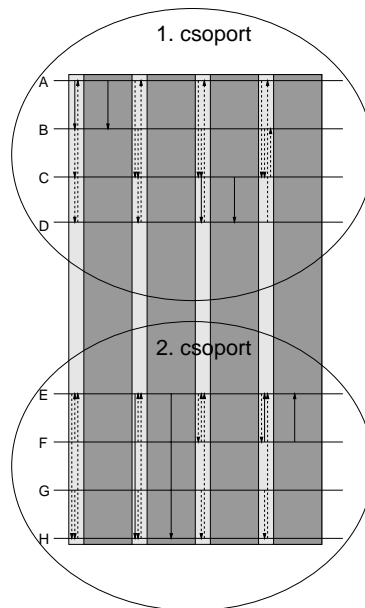
A teljesítményanalízishez bevezettem egy hálózati- és forgalommodellt, melyek segítségével összehasonlítottam a különböző FHC konfigurációkat. Az elsődleges metrika a rendszer összesített átvitele. Az egyazon lefedettségű területen használatos különböző alkalmazások modellezésére csoport alapú forgalommodellt használtam: az eszközök adataik nagy részét ugyanazon csoport tagjainak küldik. A numerikus analízis során ennek azt a szélsőséges esetét tekintem, amikor a csoport tagjai csak egymásnak küldenek csomagot. Összesen  $N$  csomópont van, amelyek  $G$  méretű csoportokba vannak osztva. A források mohón viselkednek, ami azt jelenti, hogy mindig van elküldendő csomagjuk. Minden csomagküldés előtt a célállomást véletlenül és függetlenül választják ki egyenletes eloszlás szerint a csoport többi tagja közül. Az összes  $N$  csomópont egymás küldési hatósugarán belül van, emiatt az egyes csoportokon belüli azonos frekvencián történő csomagküldések összeütköznek.

Az analízis során a fent említett három FHC konfigurációt különböztettem meg annak alapján, hogy mely csomópontok használnak közös FHC-t a versengés feloldására és kommunikációra. A *közös* (common) FHC esetben ugyanazt a csatornát használja mind az  $N$  csomópont. Ez szolgál a referencia esetként, ahol az eszközöknek nincs szükségük arra, hogy különböző FHC-kba kapcsoljanak. Az *eszköz* (device) FHC esetben külön FHC van versengésre és adatátvitelre minden egyes eszköz esetében. Ekkor minden cél csomópont esetén az eszközöknek egy új FHC-ra kell kapcsolniuk. A harmadik megvizsgált FHC konfiguráció a kettő közötti kompromisszumot képviseli. A *csoporthoz* FHC esetben minden  $G$  csomópontból álló csoport saját FHC-val rendelkezik mind versengés, mind adatküldés szempontjából. Mivel a forgalommodellünkben most a csomagokat csak a csoporton belül lehet elküldeni, ezért nincs szükség az egyes FHC-k közötti átkapcsolásra ebben az esetben sem.

A 6. - 8. ábrák mutatják a három esetet. A sötét téglalapok jelentik az adatcsomagokat, míg a világos téglalapok jelentik a versengést a csatornán. A nyilak mutatják az adatküldés és versengés irányát.

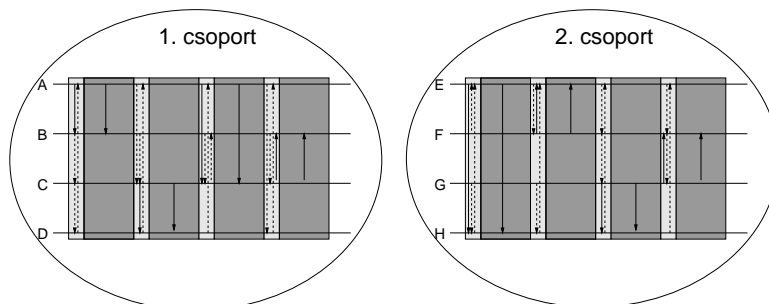
- A három FHC konfigurációra analitikus teljesítménybecslést adtam. Az eredményeket a 2. táblázat foglalja össze.

Itt  $\lambda$  jelöli az egy csomópont által felajánlott forgalmat, vagy másképpen megfogalmazva a küldéssel töltött idő arányát, beleértve az újraküldéseket is;  $\Lambda$  jelöli az FHC terhelését;  $p$  jelöli a csomagvesztési valószínűséget;  $\Theta$  jelöli az összes átvitelt;  $L_0$  jelöli a csomaghosszt;  $K$  jelöli a frekvenciák számát.

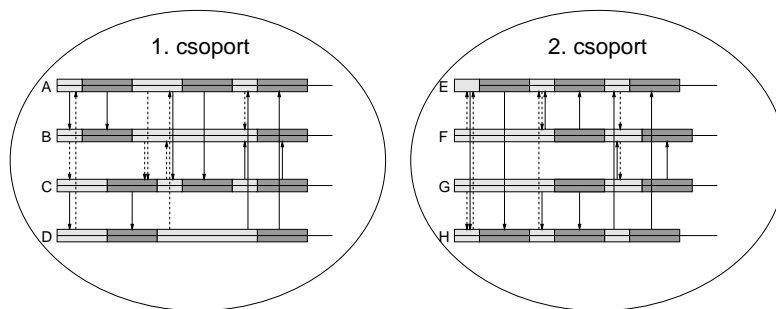


6. ábra. *Közös FHC*: az összes csomópont ugyanazt a csatornát használja.

- Elvégeztem a három FHC konfiguráció teljesítmény-összehasonlítását tipikus paraméter beállítások mellett. Több megfigyelést is tettem. Az alábbiakban numerikus eredményeket is közlök adott paraméterhalmaz [J2] mellett.
  - A *csoport* és *eszköz* FHC konfigurációk esetében a csomópontokbeli felajánlott forgalom nem függ a csoportok számától.
  - A teljes átvitel állandó a *közös* FHC esetben, és a csoportok számával nő az *eszköz* és *csoport* esetekben. Az *eszköz* FHC konfiguráció jelentősen magasabb összes átvitelt eredményezhet (háromszor akkora), mint a *csoport* eset a multiplexálási nyereség miatt.
  - A *közös* FHC adja a legjobb spektrális kihasználtságot akkor, ha úgy valósítjuk meg, hogy csak egy csatorna van a hálózatban.



7. ábra. *Csoport FHC*: minden csoport saját csatornával rendelkezik.



8. ábra. *Eszköz* FHC: minden csomópont külön csatornával rendelkezik.

Közös	Csoport	Eszköz
$\lambda_c = \frac{1}{N} \left( \frac{L_0}{L_0+e} \right)$	$\lambda_g = \frac{1}{G} \left( \frac{L_0}{L_0+e} \right)$	$\lambda_d = \frac{L_0}{2L_0+\beta e}$
$\Lambda_c = \frac{L_0}{L_0+e}$	$\Lambda_g = \frac{L_0}{L_0+e}$	$\Lambda_d = \lambda_d$
$p_c = 0$	$p_g = 1 - \left( 1 - \Lambda_g \frac{2}{K} \right)^{N/G-1}$	$p_d = 1 - \left( 1 - \lambda_d \frac{2}{K} \right)^{N-1}$
$\Theta_c = \Lambda_c \left( 1 - \frac{1}{T_b} \right)$	$\Theta_g = \frac{N}{G} \Lambda_g \left( 1 - p_g \right) \left( 1 - \frac{1}{T_b} \right)$	$\Theta_d = N \lambda_d \left( 1 - p_d \right) \left( 1 - \frac{G}{T_b} \right)$

2. táblázat. Analitikus teljesítmény eredmények a három konfigurációra

- Ha a csoport mérete kettő, a *csoport* FHC hatékonyabb, mint a megfelelő *eszköz* FHC konfiguráció (pl. 15%-al, a paraméterektől függően).
- Nagyobb csoportméretek esetében az *eszköz* FHC nagyobb összesített átvitelt biztosíthat a csoporton belül, mint a *csoport* FHC (pl. ha a csoport mérete 20, akkor az *eszköz* FHC hétszer akkora átvitelt érhet el, mint a *csoport* FHC).
- A rövid csomagok jelentősen csökkenthetik az átvitel hatékonyságát a viszonylagosan nagyobb versengési költség miatt (pl. a 12 időegység hosszú csomagok küldése kétszer olyan hatékony lehet, mint a 2 időegység hosszú csomagoké).
- Az *eszköz* FHC konfiguráció a legérzékenyebb a szinkronizáció miatti többletküldésekre, különösen nagy csoportméret esetén. Pl. ha a csomagméret 10, az átvitel csökkenése elérheti vagy meghaladhatja az 50%-ot a beacon periódustól függően.
- Megvalósítottuk az MFHC mechanizmust csomagszintű szimulációs környezetben és ennek segítségével teljesítményvizsgálatot végeztünk. A szimulátor figyelembe veszi a különböző frekvenciaugrás mintákat és ennek alapján modellezi az interferenciát a fizikai rétegben. A sérült csomagok újraküldésre kerülnek egy egy-bites ARQ visszacsatolás segítségével. Azt találtam, hogy a szimulációs eredmények megerősítik az analitikus teljesítmény fent megfogalmazott észrevételeit. A szimulációk használatával a versengés hatékonyságát mennyiségileg is vizsgálni tudtam az *eszköz* FHC esetben. Összehasonlítva a *csoport* FHC esettel, azt találtam, hogy a versengés hatékonyságának csökkenése amiatt, hogy az *eszköz* FHC esetben a csomópontnak kapcsolnia kell a különböző csatornák között, akár 8 vagy 16-szoros is lehet, a paraméterektől függően. Ennek ellenére az *eszköz* FHC konfiguráció így is jelentősen magasabb teljes átvitelt érhet el, amint azt fentebb is említettem, jobb multiplexálási képessége miatt.

- Kiterjesztettem a forgalommodellt, hogy megvizsgáljam a csoportok közötti forgalmat is. A kiterjesztett modellben  $p_{ng}$  valószínűséggel a csomópont az összes eszköz közül választ célállomást, nem csak a csoporton belüli eszközök közül. Megmutattam, hogy az *eszköz* FHC nem érzékeny erre a változásra, mivel nem függ a csomagok kialakításától. A *csoport* FHC eset csökkenő teljesítményt mutat mind kis, mind nagy csomagméretre, de a csökkenés jóval nagyobb, ha a csomagméret kicsi (40%), mint akkor, amikor nagy (25%). Mindkét esetben az eredmények azt mutatják, hogy az *eszköz* FHC konfiguráció nagyobb teljesítményt ad heterogén forgalom esetén, vagyis akkor, amikor a csoportok között is jelentős forgalom van.
- Megvizsgáltam annak a hatását is, hogy a csoporton belüli forgalom kliens-szerver alkalmazást modellez (a csoportok között most nincs forgalom). Ebben az esetben minden csoportban egy csomópontot kinevezünk szervernek, és a csoporton belül a többi eszköz játssza a kliensek szerepét. Mint korábban, most is minden csomópont mohó marad annyiban, hogy mindig van elküldendő csomagjuk, de egy konstans  $p_s$  valószínűséggel a kliensek a szervert választják célként. Ebben a kísérletben a *csoport* FHC konfiguráció átvitele állandó marad, mivel ezt a csatorna kapacitása határozza meg. Másrészt viszont az *eszköz* FHC konfiguráció csökkenő teljesítményt mutat és még a *csoport* esetén is rosszabb. Ezt az magyarázza, hogy szerver-kliens forgalom esetén az *eszköz* FHC eset nem tud multiplexálási nyereséget elérni, és sokkal kevésbé hatékony versengési sémát használ, mint a *csoport* FHC eset.

## 5. Az eredmények alkalmazása

Az 1. tézis az adathálózatok forgalmának jellemzése során használható. Jelenleg kevés tudásunk van az internet adatforgalmának természetéről és arról, hogyan függ az új alkalmazásoktól és változó felhasználói igényektől. A tézis eredményei alkalmazhatóak kereskedelmi hálózatokon végzett mérések kiértékelése során. Az ilyen mérések és analízisek hatékonyabb hálózattervezéshez vezethetnek.

A 2. tézis javaslata része lett a HIPERLAN/2 drótnélküli LAN szabványnak, amit az ETSI (European Telecommunications Standards Institute) BRAN (Broadband Radio Access Networks) projekt [28] keretében szabványosítanak. A szabványt 1999-ben fogadták el [18]. A rendszer az 5GHz-es tartományban OFDM moduláció segítségével nagysebességű összeköttetést biztosít akár 54 Mbps sebességgel [19, 20, 21]. Az ARQ javaslat több szabványosítási hozzájárulásban került ismertetésre [S1]-[S6]. Mivel az SRPB vált a végső szabvány részévé, a HIPERLAN/2 szabvány követelményei szerint készülő összes termék kötelezően ezt használja. Más gyártók mellett az Ericsson is készített ilyen szoftver és hardver prototípusokat.

A protokoll több szabadalmaztatott eljárást használ [P1, P2, P3]. Ezen felül a 3. és 4. tézisek eredményét a HIPERLAN/2 prototípus implementációja során is felhasználták.

A 5. tézis architektúrája szintén szabadalmaztatásra került [P6, P7]. Hasznosítható olyan kis költségű frekvenciaugrásos rendszerekben, mint a Bluetooth. Más megoldások, melyek a 5. tézis kutatási témájához kapcsolódnak, szintén szabadalmaztatásra kerültek [P4, P5, P8, P9].

## Hivatkozások

- [1] Leonard Kleinrock. *Queueing Systems. Vol I: Theory*. John Wiley & Sons, 1975.
- [2] Vern Paxson and Sally Floyd. Wide area traffic: The failure of poisson modeling. *IEEE/ACM Transactions on Networking*, 3(3):226–244, 1995.
- [3] Will E. Leland, Murad S. Taqqu, Walter Willinger, and Daniel Wilson. On the self-similar nature of Ethernet traffic (extended version). *IEEE/ACM Transactions on Networking*, 2(1), February 1994.
- [4] Sándor Molnár and Attila Vidács. On modeling and shaping self-similar ATM traffic. In *Proceedings of ITC-15, Washington D.C., USA*, June 1997.
- [5] Raif O. Onvural. *Asynchronous Transfer Mode Networks, Performance Issues*. Artech House, Boston, London, 1994.
- [6] Kotikalapudi Sriram and Ward Whitt. Characterizing superposition arrival processes in packet multiplexers for voice and data. *IEEE Journal on Selected Areas in Communications*, 4(6), September 1986.
- [7] R. Gusella. Characterizing the variability of arrival processes with indexes of dispersion. *IEEE Journal on Selected Areas in Communications*, 9(2), February 1991.
- [8] A. E. Eckberg, Jr. Generalized peakedness of teletraffic processes. In *ITC-10*, Montreal, 1983.
- [9] A. E. Eckberg, Jr. Approximations for bursty (and smoothed) arrival queueing delays based on generalized peakedness. In *ITC-11*, Kyoto, Japan, 1985.
- [10] G. D. Stamoulis, M. E. Anagnostou, and A. D. Georgantas. Traffic source models for ATM networks: a survey. *Computer Communications*, 17(6), 1994.
- [11] Sándor Molnár, István Cselényi, and Nils Björkman. ATM traffic characterization and modeling based on the leaky bucket algorithm. In *IEEE Singapore International Conference on Communication Systems*, Singapore, November 1996.
- [12] H. Heffes and J. M. Holtzman. Peakedness of traffic carried by a finite trunk group with renewal input. *The Bell System Technical Journal*, 52(9):1617–1642, November 1973.
- [13] Brian L. Mark, David L. Jagerman, and G. Ramamurthy. Peakedness measures for traffic characterization in high-speed networks. In *Proceedings of IEEE INFOCOM'97*, 1997.
- [14] Demetres Kouvatsos and Rod Fretwell. Batch renewal process: Exact model of traffic correlation. In *High Speed Networking for Multimedia Application*, pages 285–304. Kluwer Academic Press, 1996.
- [15] Broadband radio access networks (bran) HIPERLAN type 2 data link control (dlc) layer part 1: Basic data transport functions, April 2000.
- [16] Hui Zhang. Service disciplines for guaranteed performance service in packet-switching networks. *Proceedings of the IEEE*, 83(10), October 1995.
- [17] Jon C. R. Bennett and Hui Zhang. Hierarchical packet fair queueing algorithms. *IEEE/ACM Transactions on Networking*, 5(5):675–689.
- [18] HiperLAN/2 Global Forum. <http://www.hiperlan2.com/>.
- [19] Martin Johnsson. HiperLAN/2 — the broadband radio transmission technology operating in the 5ghz frequency band, 1999. White Paper of the HIPERLAN/2 Global Forum.



- [20] Jamshid Khun-Jush, Göran Malmgren, Peter Schramm, and Johan Torsner. HIPERLAN type 2 for broadband wireless communication. *Ericsson Review*, 2000.
- [21] Jamshid Khun-Jush, Göran Malmgren, Peter Schramm, and Johan Torsner. Overview and performance of HIPERLAN type 2 - a standard for broadband wireless communications. In *Proceedings of VTC2000 Spring*, Tokyo, Japan.
- [22] S. Corson and J. Macker. Mobile ad hoc networking (MANET): Routing protocol performance issues and evaluation considerations, 1999. RFC2501.
- [23] Bluetooth specification 1.1.
- [24] IEEE. IEEE 802.11 family of standards.
- [25] Zhenyu Tang and J. J. Garcia-Luna-Aceves. Hop-reservation multiple access (HRMA) for ad-hoc networks. In *IEEE infocom*, Mar 1999.
- [26] A. Ephremides, J. Wieselthier, and D. Baker. A design concept for reliable mobile radio networks with frequency hopping signaling. *Proceedings of the IEEE*, 1987.
- [27] Miklós Aurél Rónai and Eszter Kail. A simple neighbour discovery procedure for Bluetooth ad hoc networks. In *Proceedings of Globecom*, December 2003.
- [28] ETSI Broadband Radio Access Networks project. <http://www.etsi.fr/bran/>.

## Publikációk

### [J] Folyóirat publikációk

- [J1] Gy. Miklós, „Bluetooth: Olcsó, drótnélküli helyi összeköttetés,” Magyar Távközlés, vol. 11, No. 9, pp. 11-14, September 2000.
- [J2] Gy. Miklós, F. Kubinszky, A. Rácz, Z. R. Turányi, A. G. Valkó, M. A. Rónai, S. Molnár, „A Novel Scheme to Interconnect Multiple Frequency Hopping Channels into an Ad Hoc Network,” ACM Mobile Computing and Communications Review (MC2R), Special Issue on Wireless PAN & Sensor Networks, vol. 8, No. 1, pp. 109-124, January 2004.

### [C] Konferencia publikációk

- [C1] S. Molnár, Gy. Miklós, „On Burst and Correlation Structure of Teletraffic Models,” 5th IFIP Workshop on Performance Modelling and Evaluation of ATM Networks, 21-23 July 1997, Ilkley, U.K.
- [C2] S. Molnár, Gy. Miklós, „Peakedness Characterization in Teletraffic,” IFIP TC6, WG6.3 conference, Performance of Information and Communications Systems, PICS’98 Lund, Sweden, May 25-28, 1998.
- [C3] Gy. Miklós, S. Molnár, „Fair Bandwidth Allocation of a Wireless Base Station,” First Workshop on IP Quality of Service for Wireless and Mobile Networks (IQWiM99), April 1999, Aachen, Germany.
- [C4] Gy. Miklós, S. Molnár, „Fair Allocation of Elastic Traffic for a Wireless Base Station,” IEEE GLOBECOM’99, December, 1999 Rio de Janeiro, Brazil.
- [C5] H. Li, J. Lindskog, G. Malmgren, Gy. Miklós, F. Nilsson, G. Rydneil, „Automatic Repeat Request (ARQ) Mechanism in HIPERLAN/2,” VTC2000 Spring, May 2000, Tokyo, Japan.
- [C6] Gy. Miklós, S. Molnár, „Distributed Fair Bandwidth Allocation of a Wireless Base Station,” Networking 2000, May 14-19, 2000, Paris, France. In: G. Pujolle et al. (Eds.), „Lecture Notes in Computer Science 1815,” Springer-Verlag Berlin Heidelberg 2000, pp 689-701.
- [C7] Gy. Miklós, S. Molnár, „Analysis of a Distributed Wireless Fair Scheduling Scheme,” ITC Specialist Seminar on Mobile Systems and Mobility, pp 301-312, March, 2000, Lillehammer, Norway.
- [C8] Gy. Miklós, A. Rácz, Z. Turányi, A. Valkó, P. Johansson, „Performance Aspects of Bluetooth Scatternet Formation,” Proceedings of MobiHoc 2000, pp. 147-148, Boston, USA.
- [C9] A. Rácz, Gy. Miklós, F. Kubinszky, A. Valkó, „A Pseudo Random Coordinated Scheduling Algorithm for Bluetooth Scatternets,” Proceedings of MobiHoc 2001, pp. 193-203, Long Beach, USA.

### [T] Tanulmányok

- [T1] Gy. Miklós, „Peakedness Measures,” technical report, High Speed Networks Laboratory, Technical University of Budapest, 1997.
- [T2] S. Molnár, Gy. Miklós, „Generalized Peakedness in Discrete Time,” COST 257 TD(98)052, Granada, Spain, September 24-25, 1998.
- [T3] S. Molnár, Gy. Miklós, „A Simple Compensation Mechanism for Wireless Resource Allocation,” COST 257 TD(99)037, September 30 - October 1, 1999, Larnaca, Cyprus.

### [S] Szabványosítási hozzájárulások

- [S1] Ericsson, „Selective repeat ARQ mechanism in HiperLAN2,” HL12.5ERI7a, March 1999, Stockholm, Sweden, contribution to ETSI BRAN standardisation.
- [S2] Ericsson, „ARQ message formats,” HL13ERI8a, April 1999, Stockholm, Sweden, contribution to ETSI BRAN standardisation.
- [S3] Ericsson, „Flow control in HiperLAN 2 ARQ protocol,” HL14ERI11a, June 1999, Sophia Antipolis, France, contribution to ETSI BRAN standardisation.
- [S4] Ericsson, „Polling bit in FCCH RG for RR,” HL14ERI20a, June 1999, Sophia Antipolis, France, contribution to ETSI BRAN standardisation.
- [S5] Ericsson, „Discard mechanism in the HiperLAN2 ARQ protocol,” HL14ERI22b, June 1999, Sophia Antipolis, France, contribution to ETSI BRAN standardisation.
- [S6] Ericsson, „Error handling in the HiperLAN2 ARQ protocol,” HL14ERI24a, June 1999, Sophia Antipolis, France, contribution to ETSI BRAN standardisation.
- [P] **Szabadalmak**
- [P1] Gy. Miklós, „Discard Mechanism for Selective Repeat Automatic Repeat Request,” US patent application 09/273259, March 1999.
- [P2] J. Lindskog, J. Ollson, G. Malmgren, Gy. Miklós, „Ordering of LCH&SCH,” US patent application 09/544219, April 2000.
- [P3] G. Rydneil, J. Lindskog, G. Malmgren, Gy. Miklós, H. Li, F. Nilsson, „ARQ C-PDU Format,” US patent application 60/128041, April 2000.
- [P4] Gy. Miklós, M. Félegyházi, „Traffic Dependent Bluetooth Scatternet Optimization Procedure,” US patent application No. 09/666529, September 2000.
- [P5] Gy. Miklós, Z. Turányi, A. G. Valkó, „Flexible Neighbour Discovery in Bluetooth,” UK patent application No. 0110397.7, April 2001.
- [P6] Gy. Miklós, Z. Turányi, A. G. Valkó, „Lightweight Piconet Concept in Bluetooth,” UK patent application No. 0110396.9, April 2001.
- [P7] Gy. Miklós, Z. Turányi, A. G. Valkó, F. Kubinszky, A. Rác, „Random Access Communication in Bluetooth,” UK patent application No. 0110399.3, April 2001.
- [P8] A. Rác, Gy. Miklós, A. G. Valkó, F. Kubinszky, „Signaling Free, Self Learning Scatternet Scheduling Using Checkpoints,” US patent application No. 09/840241, April 2001.
- [P9] Gy. Miklós, A. Rác, „Predictable Communication Establishment in Ad-hoc Wireless Network” US patent application No. 10/039606, October 2001.
- [P10] Gy. Miklós, A. Rác, „Fast Hard Handover” WIPO patent application No. PCT/SE2004/000192, February 2004.