



*Budapesti Műszaki és Gazdaságtudományi Egyetem
Távközlési és Média Informatikai Tanszék*

Szélessávú hálózatokbeli hívások jelzésrendszerének teljesítményelemzése, modellezése és optimalizálása

Székely Sándor

Ph.D. értekezés tézisei

Tudományos témavezetők:

Dr. Molnár Sándor
Dr. Frajka Béla

*Nagysebességű Hálózatok Laboratóriuma
Távközlési és Média Informatikai Tanszék
Budapesti Műszaki és Gazdaságtudományi Egyetem*

Budapest
2005

1. Az értekezés előzményei

Az 1990-es évek kezdetén új technológiák jelentek meg a távközlési hálózatokban, amelyek közül az aszinkron átviteli módot, az ATM-t fogadta el az akkori távközlési világ a szélessávú hálózatok átviteli technológiájaként. Az ezredforduló előtt azonban fordulat következett be, az internet protokoll (IP) kinőtte magát a legdinamikusabban fejlődő technológiaként, amely tulajdonképpen bármely adatátviteli réteg fölött alkalmazható. Ezen két hálózati struktúra konvergenciája az ezredfordulóra vált realitássá, amelynek természetes elterjedését azonban a kezdődő világszintű távközlési piaci válság nagy mértékben megfékezte, lelassította. A jelenlegi szűkös piaci helyzetben a szolgáltatók a működési költségek csökkentésére kényszerülnek. Amíg az elmúlt években csak állandó virtuális áramköröket (PVC-t) építettek ki a végfelhasználók felé, *napjainkra egyre nagyobb az érdeklődés a kapcsolt virtuális áramkörök (SVC) iránt, amely jelzési protokollok használatára épül*, hiszen ez utóbbi jobb hálózati kihasználtságot biztosít, anélkül, hogy ez extra anyagi beruházásba kerülne a szolgáltatóknak.

Az ATM hálózatok megnövekedett komplexitása és járulékos szolgáltatásai oda vezetnek, hogy a hívások feldolgozása, de különösképpen a jelzési procedúrák egyre bonyolultabbá válnak, amely végső soron a jelzési hálózatok teljesítményének romlását idézi elő. Továbbá, a jelzési üzenetek hossza nem azonos arányban növekszik meg, aminek következtében az SS7-es jelzésrendszert leíró modellek, lásd. [Baf93], már nem tudják hitelesen jellemezni a szélessávú hálózatok jelzésfeldolgozási folyamatát. *Emiatt nyilvánvalóan új jelzés-modellekre van szükség.* Az ATM jelzésrendszerre alkalmazható alapvető *jelzés-mértékeket* az ATM Forum dolgozta ki többszöri módosítás után 2000-ben az [ATMF00] dokumentumban. Továbbá egy jó összehasonlító tanulmányt találunk négy különböző gyártó ATM jelzés-szoftverének teljesítmény-vizsgálatáról a [Kaus97] és [Nie97] cikkekben, ahol a berendezéseket különböző hálózati konfigurációkban is megvizsgálták. A [Gel97] cikk már korán beazonosítja az ATM hálózatok jelzés teljesítményét, mint potenciális szűk hálózati keresztmetszetet, ennek ellenére nagyon kevés cikk foglalkozik a továbbiakban e témakörrel (értsd., a jelzési üzenetek torlódásával az ATM kapcsolókban).

Az ATM kapcsolók hívás-teljesítménye nagy mértékben függ attól, ahogyan a kapcsoló (belső architektúrájától és processzorának kapacitásától függően) kezelni tudja a jelzési üzeneteket. Doktori tanulmányaim során megvizsgáltam különböző architektúrákat és különféle jelzési protokollok tulajdonságait, és ezzel párhuzamosan áttanulmányoztam az ipari és akadémiai szférában megjelenő publikációkat, amelyek e témát érintették. Megállapítottam, hogy nagyon kevés tudományos kutatás jutott el a szélessávú hálózatok hívás-felépítésével, -elbontásával kapcsolatos gyakorlati megfigyelésig, és ezen kevés vizsgálat is felületes, csak néhány globális paraméter megfigyelésére szorítkozik, ebből kifolyólag néhány téves következtetést is levonnak (lásd. [Pil99], [Mau01]). Fenti indítatásból kiindulva disszertációm első részében részletes méréseket végeztem ATM kapcsolókon és kellő pontossággal, rengeteg paraméter figyelembevételével elemeztem ki az eredményeket, valamint vontam le újabb következtetéseket.

Disszertációm tárgya a szélessávú hálózatok híváskezelésének teljesítményelemzése, modellezése, optimalizálása, valamint a meglévő jelzési protokollok kiegészítése új procedúrákkal. Kiemelt figyelmet szenteltem az ATM, a VoDSL, a VoIP és az UMTS hálózatok jelzésének.

Tézisfüzetem 2. fejezete bemutatja a kutatásom célkitűzéseit, majd a 3. fejezetben a kutatási módszertant, illetve néhány fontos definíciót találunk. A 4. fejezet tartalmazza az új tudományos eredményeket, öt különböző téziscsoportba sorolva. Az 1. tézis a híváskezelés teljesítményét méri izolált ATM kapcsolók esetén. A fő cél a hívás felépítési és -elbontási idők fő komponenseinek beazonosítása, és különböző hívásprofilok hatásának kvantitatív jellemzése egyszerű és többszörös pont-pont összeköttetésekre, valamint pont-többpont kapcsolatokra. Többek között egy új módszer („populációs diagram”) is bemutatásra kerül, amely segít megtalálni olyan rejtett rendszer-hibákat,

melyek hagyományos módszerekkel nem észlelhetőek. Felhasználva az 1. tézis eredményeit, egy új híváskezelési modellt fejlesztettem ki a 2. tézisben, majd hálózati szintű vizsgálatokat végeztem vele a 3. tézisben. Az esettanulmányaim kiterjednek egy 10 csomópontos kaszkádra, egy 4-es hálóra, egy 7, 30, illetve 35 csomópontos tetszőleges hálózatra. A 4. tézisben bemutatok egy optimalizáló algoritmust, amely az UMTS hálózatra illesztett 2. tézisbeli modell hívásfelépítési és -elbontási idejét minimalizálja. Az 5. tézis a hálózatban elveszett szélessávú hívások megismétlésével és sorbanállási analízisével foglalkozik.

A tézisfüzet 5. fejezete az új eredményeim jelenlegi, illetve lehetséges alkalmazásait sorolja fel, a 6. fejezetben a felhasznált szakirodalmat gyűjtöttem össze, míg a saját publikációim a 7. fejezetben kaptak helyet. Végül tézisfüzetemet a 8. fejezettel, egy köszönetnyilvánítással zártam.

2. Kutatási célkitűzések

A disszertációm célja a szélessávú jelzés-hálózatok teljesítményének elemzése, olyan új vizsgálati módszerek és modellek kidolgozása, melyek segítségével pontosabban leírható a szélessávú hívások bonyolult folyamata, továbbá a jelzésprocesszort optimalizáló új algoritmusok kidolgozása, valamint meglévő jelzési protokollok kiegészítése új funkciók, szolgáltatások betöltésére.

Kezdetben kutatásaim alapvetően az ATM hálózatokra irányultak, de hasonló problémák merülnek fel a 3G mobil hálózatok UTRAN hozzáférési hálózataiban, ahol az ATM alapú AAL2-es adaptációs megoldás lett az elfogadott szabvány (lásd. [Ene99], [C-7]). Továbbá az új technológiák megjelenése a digitális előfizetői hurokban (xDSL), az intelligens hálózati megoldások (IN), az IP-over-ATM, Voice-over-ATM és az intelligens WWW megoldások (pld. RSVP) mind-mind a jelzés-alapú kapcsolat-felépítést sürgetik a meglévő ATM hálózatokban, megalapozva az ilyen jellegű valós mérésekre irányuló erőfeszítéseket (lásd. [Mer00]).

Ennek érdekében az alábbi kutatásokat végeztem:

- négy kereskedelmi forgalomban található ATM kapcsoló jelzés-processzorának teljesítményét mértem, és beazonosítottam azon legfontosabb tényezőket, amelyek a hívásfelépítési és -elbontási időket lényegesen befolyásolják, miközben megvizsgáltam a jelzésüzenetek komplexitását, annak érdekében, hogy kiszűrjem az ATM specifikus információs elemeket;
- kidolgoztam egy új hívás modellt, amely tükrözi az összes ATM specifikus tulajdonságokat, majd ezen modellt úgy prioritásos, mint prioritás mentes (FIFO) sorokra megvizsgáltam;
- összehasonlítottam a kaszkádolt sorok híváskezelési teljesítményét tetszőleges hálózati topológiák teljesítményével, továbbá megvizsgáltam a hívás-terhelés eloszlását ezekben a hálózatokban;
- kidolgoztam egy szimuláción alapuló optimalizáló algoritmust a híváskezelési idők csökkentésére;
- kiegészítettem a jelenlegi standard jelzési protokoll állapotgráfját úgy, hogy alkalmas legyen a hálózatban blokkolt szélessávú hívások ismételt sorbanállítására, majd analitikus vizsgálatnak vettem alá.

3. Kutatási módszertan

A mérnöki világban széles körben elfogadott tény, hogy egy új jelzés-rendszert szigorú teljesítmény-tesztelésnek kell alávetni, mielőtt alkalmazásra kerül, s a kiválasztott módszertannak nagyon világosan meg kell mutatnia a jelzési hálózat azon képességét, hogy a kért szolgáltatásokat elfogadható késleltetési értékek mellett támogatja. Tekintettel arra, hogy a valós mérések nyújtják a legmegbízhatóbb eredményeket, és ugyanakkor a piacon hozzáférhető ATM kapcsolók közül több

készülék rendelkezik jelzési szoftverrel (kb. 10 ilyen ismert márkájú kapcsoló létezik szerte a világon), úgy döntöttem, hogy *teljesítmény mérésekkel* támasztom alá az 1. tézisben szereplő állításokat. A következő négy típusú ATM kapcsolót vetettük alá vizsgálatainknak: *GDC APEX DV2* (1995), *Fore ASX200BX* (1997), *Newbridge MSX36170* (1999) and *Seabridge XP140* (2001). Mint látható, ez egy jó reprezentatív minta, lefedve különböző generációjú termékeket 1995 és 2001 között. Pont-pont kapcsolatokat vizsgáló mérési eredményeimet az 1.1 tézis mutatja be, és ezek nagyon sok esetben összhangban vannak más kutatói csoportok eredményeivel (lásd. [Kaus97], [Nie97], [Nov99]) amelyeket valójában öt másik kapcsolón végeztek. Az eltérő eredményekre esetenként magyarázatot adtam. További 2, 3 vagy 4 kaszkádolt kapcsolókon elért eredmények (lásd. [Mau01], [Far01]) is alátámasztják az 1.1 tézisben megfogalmazott eredményeimet.

Az 1.2 tézisben a börsztös érkezésű jelzési üzenetek hatását vizsgáltam. Megállapítottam, hogy a hívásfelépítésre, és -elbontásra alkalmazott klasszikus vizsgálati módszerek itt nem bizonyulnak elegendőnek, olykor téves következtetések levonásához vezetnek (lásd. [Pil99], [Mau01]), ezért egy új vizsgálati módszert vezettem be, amelynek segítségével a börszt minden egyes üzenetét ábrázolni lehet egyetlen ábrán, egy ún. "populációs diagram"-ban, új dimenziókat nyitva ezáltal a börsztös események elemzésében. Alapvetően ezen új módszer nem csak börsztös érkezések vizsgálatára alkalmas, hanem bármely típusú teszt sorozatra, főként ahol hosszú tesztelési időkre és rengeteg hívásra kell számítanunk.

A témában publikált eddigi cikkek lényegesen eltérnek az azonos szakkifejezések megnevezését illetően. Emiatt nem volt könnyű feladat a különböző cikkek közötti eredmények összehasonlítása, így első lépésként ezen kifejezéseket „átültettem” az ATM Forum [ATMF00] által meghonosított terminológiákra. Ezenkívül szükséges volt néhány új paraméter bevezetésére, amely lehetővé tette a jelzésforgalom leírását hálózati szinten (lásd. 1.1 tézis és 3. tézis).

Miközben a szakmai világ egyetért az izolált kapcsolón végzett teszt sorozatok jelentőségével, ugyanakkor nyilvánvaló, hogy egy nagyobb LAN illetve WAN hálózati konfigurációban végzett méréseknek szintén kiemelt szerepet kell szánni. A felmerülő probléma viszont ebben az esetben a kellőképpen nagyméretű ATM jelzés-hálózatokhoz való hozzáférés. Pusztán tesztelés szempontjából egy reprezentatív számú (pld. 30 darab) ATM kapcsoló beszerzése túl költséges és nem elfogadható megoldás. Analitikus teljesítményvizsgálat által (pld. folyam-analízis) csak a középértékek becslésére kapunk elfogadhatóan jó eredményeket. A jelzési protokoll emulációja által egy mély betekintést nyerhetünk a jelzés-protokoll viselkedésébe, de teljesítményvizsgálata korlátokba ütközik, különösen akkor, ha egy olyan hálózatot vizsgálunk, ahol több különböző gyártmányú kapcsoló foglal helyet. Ezért én a *szimulációs módszert*¹ választottam. Ez ugyan egy magasabb szintű, absztraktabb megközelítés, de az általunk kifejlesztett eszköz képes bármely gyártmányú ATM-kapcsoló paramétereit modellezni. A szimuláció végén például az alábbi eredményeket láthatjuk: hívás felépítési idő, hívás elbontási idő, a vonalszakaszok sávszélesség foglaltsága, hívás sikerességi ráta, a csomópont processzorának terheltsége a jelzésfolyam által, stb. Vizsgálataim eredményeként egy új jelzési csomópont-modellt állítottam fel, ahol a csomóponton belül a különböző jelzések különböző folyamatokat járnak be, majd vizsgálataimat kiterjesztettem kaszkádolt, illetve tetszőleges topológiájú ATM, illetve UMTS hálózatokra (lásd. 2. és 3. tézis). Továbbá összehasonlítottam végtelen és véges sorbanállási modelleket, kidolgoztam egy optimalizálási algoritmust, valamint prioritásos üzenetkezelést is vizsgáltam (lásd. 4. tézis). A szimulációs eredményeimet valódi mérési eredményekkel vettem össze egy 2-4 csomópontos kaszkád, illetve egy ismert topológiájú 7 csomópontos TEN-155 hálózat esetén [Nov99].

A megalkotott új modell bonyolultsági foka nem teszi lehetővé a könnyed analitikus vizsgálatot, ezért e célból egy egyszerűsített modellt is készítettem, mely természetesen néhány hátrányt is

¹ A szimulációs tanulmány egy ACCEPT nevű, saját szoftveren alapul, készítette a szerző és kollégája, Moldován I., HSN Laboratórium, BME-TTT.

magában rejteget, ezeket a 2.2 tézisben soroltam föl. Ezen modell *analitikus* megoldásához a sorbanálláselmélet ismert eredményeit használtam föl, egészen pontosan a BCMP hálózatok megoldási sémáját [BCMP75], természetesen a szükséges kiegészítésekkel.

Végül az 5. tézisben bevezettem az UNI jelzési protokoll egyszerűsített állapotdiagramját, kiegészítve egy új állapottal, amely lehetővé teszi az elveszett hívások ismételt sorbaállítását az első hálózati kapcsolóban. Végül ezen blokkolt szélessávú hívások sorbanállási analizisét úgy *szimulációs*, mint *analitikus* eszközökkel elvégeztem.

Az alábbiakban néhány fontosabb paraméter *definícióját* ismertetem a tézisek könnyebb érthetősége kedvéért, további definíciók az 1. ábrán és a disszertációm [D-10]-es fejezetében találhatóak.

1. A hívás felépítési idő az az idő, amely alatt egy adott jelzési rendszer (pld. ATM) kapcsolt (virtuális) összeköttetést hoz létre különböző hálózati elemek között.

$$T_C = t_S(Y) - t_S(X), \quad (1)$$

ahol $t_S(.)$ a forrásból kilépő, illetve oda beérkező üzenetek időpontja; $X=SETUP, ERQ$; $Y=CONNECT, ECF$ jelzési üzenetek, lásd. [Q2931], [UNI40], és [AALQ99].

Fenti paraméter valójában a legalapvetőbb hívásszintű teljesítmény mutató.

2. A hívás elbontási idő az az idő, amely alatt egy adott jelzési rendszer elbontja a kapcsolt összeköttetést egy adott hálózati eleme fölött:

$$T_R = t_S(Y) - t_S(X), \quad (2)$$

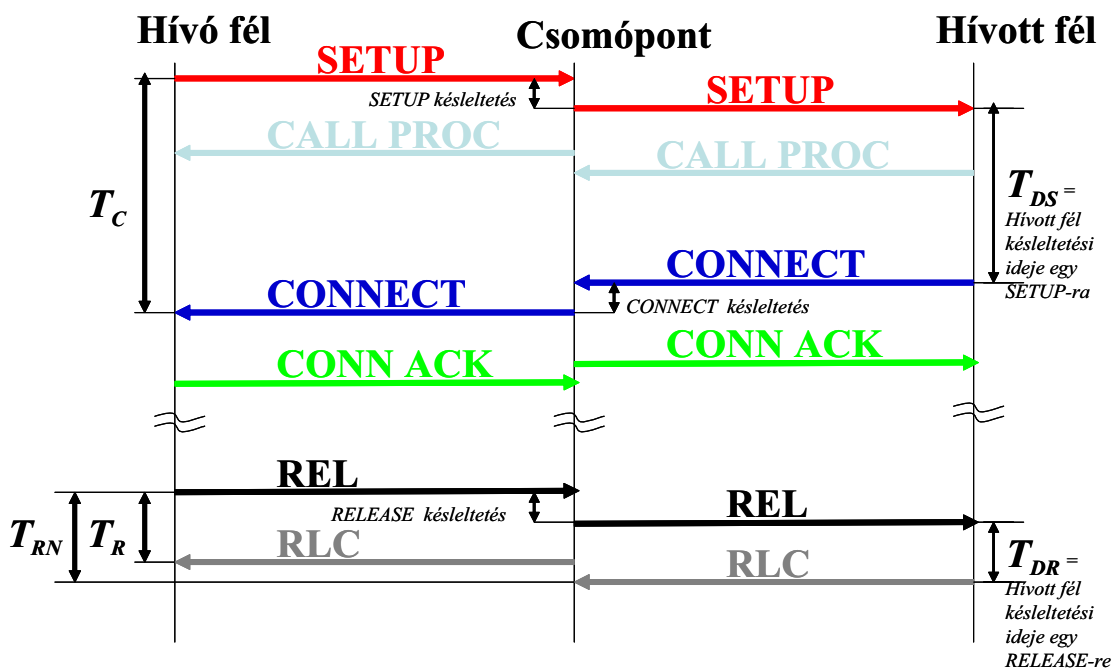
ahol $X=REL, Y=RLC$ jelzési üzenetek, lásd. [Q2931], [UNI40] és [AALQ99].

3. A hívás felépítési késleltetés a hívás felépítési idő és a hívott félnek a *SETUP* üzenetre adott késleltetési ideje közötti különbség:

$$T_{CN} = T_C - T_{DS}, \quad (3)$$

ahol T_{DS} a hívott félnek a *SETUP* üzenetre adott késleltetési ideje.

4. Egy adott kapcsoló (illetve hálózat) hívás sikerességi rátáját (γ_R) a kapcsolón (hálózaton) áthaladó sikeres, illetve összes hívások aránya adja.



1. ábra Hívás felépítési és -elbontási idők, illetve késleltetések definíciója

4. Új tudományos eredmények

Téziseimet öt különböző csoportba soroltam az adott tématerületek szerint. Minden egyes csoport két tézist tartalmaz a 3. tézis kivételével, amely mindössze egy téziscsoportból áll. Minden tézis végén hivatkozások találhatóak a disszertáció egyes fejezeteire a további részletek illetve bizonyítások céljából, ezeket [D-*n*]-nel jelöltem, ahol $n = 1, \dots, 10$ a fejezetek sorszáma.

Az ATM pont-pont összeköttetések jelzési folyamatának sajátos tulajdonságai

1. tézis *Megvizsgálva az ATM kapcsolók belső architektúrájától független jelzési tulajdonságait, megállapítottam és kielemeztem a pont-pont kapcsolatokra irányuló jelzésfolyamok sajátos tulajdonságait. Ezenkívül bevezettem egy új elemzési módszert (ú.n. populációs diagramot) borsztös hívások sűrűsödési helyeinek elemzésére.*

Ebben a tézisben egy kerek egységbe tömörítettem egy egész készletre való jelzési tulajdonságot, amelyek *kapcsoló-függetlenek*, és ezáltal *általánosíthatóak* további ATM kapcsolókra is. Eredményeim összhangban vannak további publikált mérési eredményekkel, lásd. [Nie97], [Nov99], [Far01] és [Mau01]. Több különböző, teljesítményelemzésre vonatkozó méréssorozatot futtattunk le úgy *egyedi-*, mint *párhuzamos pont-pont* kapcsolatokra. Végeztünk méréseket *pont-többpont* kapcsolatokra is, de ezen eredményekből még nem sikerült kellőképpen általánosítható eredményeket leszűrni (lásd. [C-8]), ezért ezek az eredmények nem kerültek be a disszertációmba. A hangsúlyt az OSI 3.-ik rétegbeli (jelzésszintű) elemzésre helyeztük. A leggyakrabban vizsgált konfiguráció egy izolált kapcsolóhoz csatolt generátorból és vevőből állt. Méréseket végeztünk a jelzésfolyamatok állandósult- illetve tranzienst állapotában, ezen mérésekből származó eredményeket két különböző tézisben foglaltam össze:

1.1 tézis *ISDN-től eltérő, különösen az ATM hálózati pont-pont kapcsolatok jelzésforgalmára jellemző sajátos tulajdonságokat mutattam ki az alábbi 7 altézisbe tömörítve, állandósult állapotban végzett mérések esetén. [C-5], [C-9], [J-4], [H-3], [D-4]*

1. táblázat Az 1.1 tézis áttekintése

1.1.1.	A 3.-ik szintű jelzésüzenetek feldolgozási idejének dominanciája + új mértékek definíciója
1.1.2.	Lényeges különbségek a jelzési üzenetek feldolgozási idejében
1.1.3.	A T_C függősége a forgalomirányító táblától, sávszélesség kiosztástól és élő kapcsolatok számától
1.1.4.	Üzenetek késleltetési idejének függősége a hívás-profiltól
1.1.5.	A hívás elbontási folyamat közvetett hatása a T_C időre
1.1.6.	Jelzés túlterhelés hatása a T_C és a T_{RN} időkre
1.1.7.	A T_{CN} idő becslése kaszkádolt kapcsolók esetén

Megmutattam, hogy a korábban használt, szabványszervezetek által definiált teljesítmény mértékek (pld. hívás felépítési idő, hívás elbontási idő) nem elegendőek a szélessávú jelzés-hálózatok kielégítő szintű jellemzéséhez. Kimutattam, hogy az alábbi paramétereket sem szabad figyelmen kívül hagyni: a hívott fél válaszadási ideje, az elbontási késleltetés, a kapcsolók hívás-terheltsége, a forgalomirányító tábla mérete, valamint a hívások profiljának komplexitása (ami a *SETUP* üzenetek információs mezőjének tartalmára van hatással). Megállapítottam, hogy a T_R hívás elbontási idő jelen formájában nem nyújt valódi képet az adott hívás elbontásáról egy adott útvonal mentén. Ezért **bevezettem két új teljesítmény mértéket**, amelyek hiteles mutatói az ATM hálózati szakasz lappangási idejének:

1.1 definíció *A hívás elbontási késleltetés az az idő, amely alatt a RELEASE üzenet eljut a hálózaton át végtől-végig, s amelyet egy RELEASE COMPLETE üzenet követ a célállomásnál:*

$$T_{RN} = t_D (RLC) - t_S (REL), \quad (4)$$

ahol $t_D(.)$ a célállomáshoz bérkező vagy onnan kiinduló üzenetek időpontja.

1.2 definíció Az *összesített hívás kezelési idő* a hívás felépítési idő és a hívás elbontási késleltetés összege:

$$T_H = T_C + T_{RN}. \quad (5)$$

Az alábbiakban bemutatott méréseink a hívások állandósult állapotában készültek. Az eredményeket 7 egymástól független altézisbe soroltam (1.1.1.-1.1.7.), ezen állítások rövid áttekintése az 1. táblázatban található.

1.1.1. Kiindulva a mért eredményekből és az elvégzett analitikus vizsgálatokból, **megállapítottam, hogy a T_C hívás felépítési idő legalább egy nagyságrenddel nagyobb késleltetést szenved a 3. szintű OSI rétegben, mint az alacsonyabb rétegekben.**

A 3. szintű üzenet-feldolgozás (az UNI4.0 rétegben) szoftver alapú, míg a 2. OSI rétegben hardver-szoftver kombináción alapszik, végül az 1. szinten csak hardver alapú. Mérési és számítási eredmények, valamint egy összehasonlító táblázat is található a disszertáció [D-4] fejezetében. Fenti altézis következményeként, a továbbiakban csak a 3. szintű üzenet-feldolgozási időket figyeltem.

1.1.2. **Megállapítottam, hogy az alábbi összefüggés érvényes az összes (tesztelt) ATM kapcsolóra, függetlenül a kapcsoló típusától és teljesítményétől: a *CONNECT* és *RELEASE* üzenetek késleltetésének minimuma mindössze (25...35)%-a a *SETUP*² üzenet késleltetési minimumának.**

1.1.3. **Kimutattam, hogy a T_C hívás felépítési idő *lineárisan* függ a forgalomirányító tábla hosszától, ellenben nem függ a híváshoz hozzárendelt sáv szélességtől.** Továbbá kimutattam, hogy a kapcsolón keresztül létesített élő hívások száma nem befolyásolja az új hívás felépítési idejét.

Saját eredményeink kiegészítéseként e témakörben további mérési eredmények találhatóak a [Nie97] és [Far01] cikkekben, ahol többek között a PNNI hierarchia szintektől való függőséget is megpróbálták behatárolni. A sáv szélességtől való függőség mérésekor figyeltünk arra, hogy ne okozzunk sikertelen hívást azért, hogy a sáv szélesség nem elegendő valamely linken.

1.1.4. **Megadtam egy kvantitatív leírást arra vonatkozóan, hogy hogyan függ a különböző hívás-profiloktól az átlagos *SETUP*-, *CONNECT*- és *RELEASE* késleltetés (és ezáltal a T_C hívás felépítési idő és T_{RN} hívás elbontási késleltetés) egy adott ATM csomópontban:**

- Az átlagos *SETUP* késleltetés akár megkétszereződése a különböző típusú hívás profilok bonyolultságának köszönhető, hiszen a profilok igencsak változatos jelleget mutatnak, ha összehasonlítunk egy egyszerű beszédhívást egy komplex multimédia alapú hívással. Erre néhány példát a [D-4] szolgáltat. Általában, a hívás felépítési idő változása a *SETUP* üzenet bonyolultságának függvényében az alábbi egyenlettel írható le:

$$T_C^{CCP} = (1 + s) \cdot T_C^{default}, \text{ ahol } 0 < s < 1. \quad (6)$$

- Ellentétben az átlagos *SETUP* késleltetéssel, úgy találtam, hogy az átlagos *CONNECT*-, illetve *RELEASE* késleltetés nem növekszik, ha további IE-vel bővítjük a *SETUP* üzenetet. Továbbá megállapítottam, hogy az egyszerű és a komplex hívások elbontási késleltetése azonos:

$$T_{RN}^{CCP} = T_{RN}^{default} \quad \forall s, \quad 0 < s < 1. \quad (7)$$

² Itt csak az előírt Információs Elemeket (IE-eket) tartalmazó *SETUP* üzenet értendő, melynek hosszúsága 3 ATM cella.

- Továbbá kimutattam, hogy az átlagos hívás felépítési idő nem függ a hívás típusától, azaz:

$$\bar{T}_C|_{VBR} \approx \bar{T}_C|_{CBR} \approx \bar{T}_C|_{ABR} \approx \bar{T}_C|_{UBR}, \quad (8)$$

ahol \bar{T} az átlagos időt jelöli. Korábbi cikkek a szakirodalomban azt feltételezték, hogy a VBR típusú hívások felépítési ideje nagyobb lesz, mint a CBR típusú hívásoké (lásd. [Ge197], [Wu97]), de saját méréseink mindennek ellentmondanak. A [Far01] cikk további támpontot nyújt a (8)-as egyenletünkhöz azáltal, hogy kimutatta: az UBR típusú hívások felépítési ideje elhanyagolható mértékben csökken a CBR típusú hívásokéhoz képest egy 4 csomópontos PNNI hierarchiában.

1.1.5. Megvizsgáltam az egyszerű hívások (melyeket alapbeállítású SETUP üzenettel generálunk) felépítési késleltetését (T_{CN}) két különböző esetben: amikor a hívás felépítést röviddel utána egy elbontás követi, illetve ha az elbontás nem következik be, majd az alábbi megállapításokra jutottam:

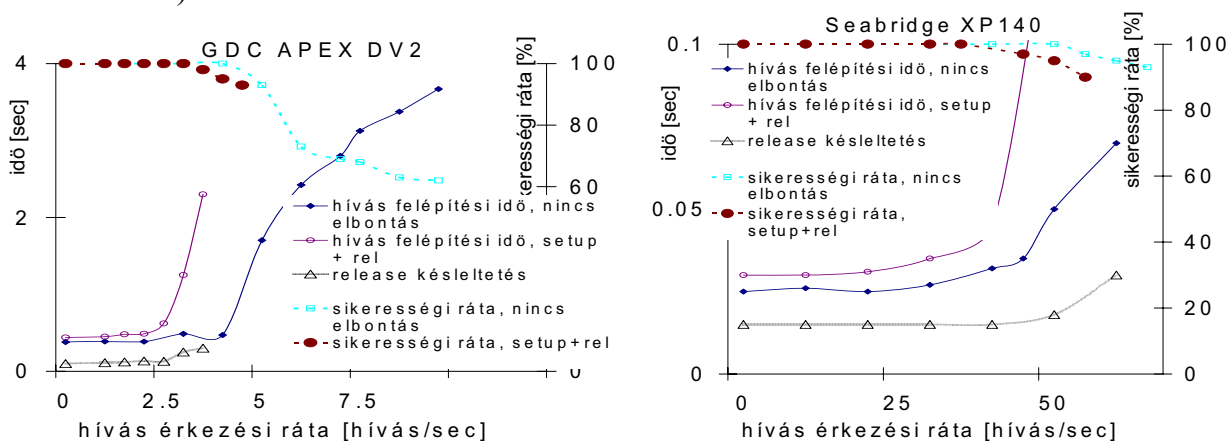
- Ha a hívásokat egy bizonyos idő után elbontjuk, az így mért hívás felépítési késleltetések (T_{CN1}) átlaga (15-20)%-kal nagyobb, mint azon hívásoké amelyeket nem bontunk el (T_{CN2}). Ez a tulajdonság független a hívások érkezési intenzitásától (lásd. 2. ábra).

$$E(T_{CN1}) \approx 1.175E(T_{CN2})$$

- Amennyiben a hívásokat egy bizonyos idő után elbontjuk, az így nyert küszöbérték (λ_{01}), azaz azon hívás-intenzitás mellett a hívás felépítési késleltetések ugrásszerűen megnövekednek, mindössze (65-70)%-a azon küszöbértéknek (λ_{02}), ahol a hívásokat nem bontjuk el. Ezen küszöbértékek fölött megjelennek a blokkolt hívások (lásd. 2. ábra).

$$\lambda_{01} \approx 0.67\lambda_{02}$$

- Kimutattam, hogy a hívások tartási ideje nem befolyásolja a hívás felépítési időt, egy kivétellel, amikor a hívások időtartama végtelen, azaz az elbontás nem következik be (lásd. 2. ábra).



2. ábra Hívás felépítési idő, hívás elbontási késleltetés és sikerességi ráta a hívás intenzitás függvényében
a) GDC APEX DV2; b) Seabridge XP140

Megjegyzés: Eredményeink itt ellentmondanak a [Kaus97]-féle cikkel, amely megállapítja, hogy a nulla (illetve véges) időtartamú hívások felépítési ideje azonos a végtelen időtartamú hívásokéval.

1.1.6. Ha egy ATM kapcsolót enyhén túlterhelünk beérkező hívásokkal (azaz a hívásoknak kevesebb, mint 10%-a blokkol), akkor a hívás felépítési idő ugrásszerűen megnövekszik, ellenben a hívás elbontási késleltetés görbéjének meredeksége ebben a tartományban még nem változik (lásd. 2. ábra). Ha tovább növeljük a hívás intenzitást, akkor az elbontási késleltetés is hirtelen megnövekszik.

Fenti eset akkor következhet be, ha prioritásos üzenetsorokat alkalmazunk, vagy ha a *SETUP* és *RELEASE* üzenetek legalább részben különböző folyamatokat járnak be a processzor belső architektúrájában. Az általunk vizsgált 4 ATM kapcsoló egyikében sem alkalmaztak prioritásos sorokat, hanem osztott belső architektúrát. A blokkolt hívások az üzenetsorok pufférének túlsordulása miatt következtek be.

1.1.7. Feltételezve, hogy egyazon profilú hívások ‘r’ darab azonos típusú kaszkádolt kapcsolón haladnak át (egy PNNI hierarchia csoporton belül), megmutattam, hogy a hívások felépítési késleltetése kielégíti az alábbi egyenlőtlenséget:

$$r \cdot \bar{T}_{CN} \Big|_{1switch} \geq \bar{T}_{CN} \Big|_{r'switches}, \quad r = 1, 2, \dots, k. \quad (9)$$

Fenti összefüggést általánosíthatjuk a heterogén típusú kaszkádra is (egyazon PNNI hierarchia csoporton belül maradva):

$$\sum_{i=1}^P n_i \cdot \bar{T}_{CN}^{type i} \geq \bar{T}_{CN} \Big|_{r'switches}, \quad r = \sum_{i=1}^P n_i, \quad (10)$$

ahol P a különböző típusú ATM kapcsolók száma, és n_i pedig az azonos típusú kapcsolók száma. Fenti tulajdonság annak tudható be, hogy az üzenetek többé-kevésbé egyenletesen oszlanak el a kaszkádolt sorban. A [Mau01]-féle cikk mérési eredményei alátámasztják a (10)-es egyenlőtlenséget. A későbbiekben (2.2 tételben) még egyszer visszatérek erre az egyenlőtlenségre, és ugyanakkor felállítok rá egy megközelítő egyenletet.

1.2 tétel *Egy új elemzési módszer bevezetésével sikerült kimutatnom a kétpontos ATM kapcsolatok börsztös érkezésű hívásainak sajátos tulajdonságait, tranziens állapotban végzett mérésekkel (lásd. 2. táblázat). [C-5], [C-9], [T-2], [D-4]*

2. táblázat Az 1.2 tétel rövid áttekintése

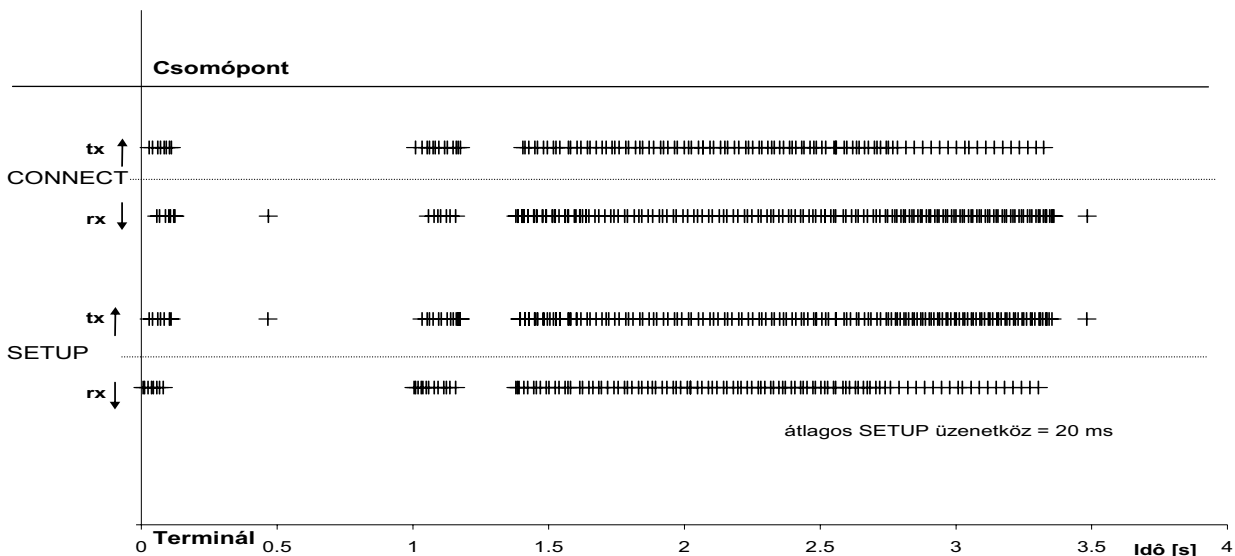
1.2.1.	Egy új elemzési módszer a börsztös hívások vizsgálatára: “populációs diagram”
1.2.2.	Börsztös hívások sajátos tulajdonságai
	1.2.2.1. Az első hívás felépítése hosszabb ideig tart mint az azt közvetlenül követő hívásé
	1.2.2.2. A <i>CONNECT</i> késleltetés görbéjének változása egy adott börszt alatt
	1.2.2.3. Megközelítő képlet felállítása a $T_C(b)$ -re
	1.2.2.4. Megközelítő képlet felállítása a $T_{RN}(b)$ -re

Ha csak az állandó intenzitással érkező hívások teljesítmény analízisét végezzük el (lásd. 1.1 tétel), még nem kapunk egy teljes képet az ATM kapcsolók viselkedéséről jelzésfolyamok esetén. Pont-többpont típusú kapcsolatok felépítése esetén különböző viselkedésre számítottam a kapcsolók részéről, továbbá abban az esetben ha egy ATM hálózati kapcsoló kiesését automatikusan követi az addig ott áthaladó kapcsolatok pillanatszerű áterelése a szomszédos kapcsolókra, amely egyértelműen börsztös jelzésfolyamokhoz vezet. Sajnos az eddigi szakirodalomban a börsztös jelzésfolyamok mérését illetően nagyon ellentmondó eredményekre bukkantam (lásd. például [Mau01], [Pil99]). Ezért elhatároztam, hogy ezeket az eredményeket újraelemzem, kiegészítve saját méréseinkkel. Hamarosan be kellett látnom, hogy a klasszikus módszerekkel végzett elemzés (jelzés-folyamára, hívás felépítés statisztikai analízise) nem hozott áttörést, így még mindig több kérdésem volt a lehetséges, illetve ellentmondásos folyamatokra, mint válaszom. Annyi mindenesetre kiderült, hogy bizonyos “rejtett” rendszer-hiba miatt az eredmények nem konvergálnak. Ennek megfejtésére viszont új módszert kellett kidolgoznom. Eredményeim az 1.2 tételben foglaltam össze, amely valójában két független altételre oszlik:

1.2.1. Bevezettem egy új elemzési módszert, az ún. “populációs diagram”-ot, amely a jelzési üzenetek sűrű folyamát oly módon helyezi egy struktúrált diagramba, hogy a borsztös folyamatok részletei “láthatóvá” váljanak.

A módszer alapvető lépéseit az alábbiakban sorakoztattam föl (egy példát is bemutatok a 3. ábrán):

- helyezzük el az összes (illetve csak az érdemi információt szolgáltató) 3. szintű jelzés-üzenetet, amely a megfigyelt borsztös hívás során keletkezett egy, az alábbiak szerint struktúrált diagramba: a vízszintes síkban több párhuzamos időtengelyt találunk, a függőlegesen pedig ezúttal a “link”-eken áthaladó üzeneteket ábrázoljuk;
- ábrázoljunk egy adott jelzési üzenetet, amely áthalad két csomópont közötti interfészen (linken) egy “+” jellel (vagy ponttal), s ne a szokásos két csomópontot összekötő nyíllal;
- hozzunk létre “m” darab időtengelyt adott interfészenként (ahol “m” az ábrázolásra kerülő üzenettípusok száma), majd helyezzük el a “+” jeleket az adott üzenettípusnak megfelelő tengelyen, az áthaladási időpontok sorrendjében;
- sokszorozzuk meg a függőleges mentén a diagramot, azaz interfészek számát (n+1)-re, ha a vizsgált üzenetek “n” csomóponton haladnak át;
- ha a vizsgált hívások irányítottságára is kíváncsiak vagyunk, osszuk fel mind az “m” tengelyt interfészenként “2xm” altengelyre, a felsőre helyezzük az egyik irányba haladó üzeneteket, míg az alsóra az ellentétes irányítottságúakat;
- helyezzük el (hasonló logika szerint) a 2. szintű jelzési üzeneteket is (pld. *POLL*, *STAT*, *USTAT*), ha az elemzés során erre szükség van.



3. ábra Példa egy “populációs diagram”-ra ($m=2, n=0$), amely 300 (kétirányú) hívás borsztös érkezését mutatja be

A szemléltetés kedvéért a 3. ábra a végfelhasználó és az ATM kapcsoló közötti interfészen áthaladó jelzési üzenetek egy kiválasztott részhalmozát (a *SETUP* és *CONNECT* üzeneteket) ábrázolja, mint említettem érkezésük sorrendjében. Nagyon fontos, hogy az üzenettípusokat egy meghatározott sorrendben helyezzük el egymás fölött, lehetőleg előfordulásuk sorrendjében (lásd. [Q2931], például, *SETUP*→*CALL_PROC*→*CONNECT*→*CONN_ACK*→*STATUS_ENQ*→*STATUS*→*REL*→*REL_COMP*).

Megjegyzés: A fenti populációs diagram nem csak borsztös érkezések vizsgálatára alkalmas, előnye akkor is megmutatkozik, ha hosszú órákig tartó mérési folyamatokat kell kielemeznünk. Erre rengeteg példát találunk a disszertációm [D-4]-es fejezetében és a [T-2]-es szabadalomban. Ezen diagram egy lényegesen jobb “rálátást” biztosít a teljes borsztös folyamatra, mint a klasszikus “nyíl” folyamatábra. Amíg az 1. ábrán mindössze egy hívás felépítését ábrázoltuk, addig kb. ugyanakkora felületen a 3. ábrán 300 hívás borsztös érkezési folyamatát láthatjuk. És ennél jóval nagyobb “sűrítést” is elérhetünk a populációs diagrammal.

Egy új meghatározást adtam a jelzési üzenetek börsztös érkezésének, amely az érkezési csomópontban vizsgálja a jelenséget, nem úgy mint egy ismert korábbi definíció, amely a generátor kiszolgálási folyamatát jellemzi és időklasztereket vizsgál. Alábbi definícióm egy állapotváltozó megfigyelésén alapszik:

1.3 definíció *Jelzési üzenetek börsztös érkezéséről beszélünk, ha egy adott csomópontban egy CONNECT üzenet érkezési pillanatában több, mint 2 darab SETUP üzenet található a csomópont pufférében, azaz:*

$$n[SETUP_{in}(i)] - n[SETUP_{out}(i)]_{\Delta t} > 2 \quad (11)$$

ahol $i = 1, \dots, r$ a vizsgált csomópont, $n[X(i)]$ az X típusú üzenetek száma az i csomópontban, Δt pedig szabadon választható időintervallum.

A továbbiakban, a “jelzések börsztös érkezése” terminológiát az 1.3 definíció szerint értjük. Miután újraelemeztem a börsztös mérési eredményeinket az új definíció figyelembe vételével, az alábbi következtetésekre jutottam, melyeket az 1.2.2 tézisben foglaltam össze és a *börsztös jelzési struktúrák tulajdonságait* emeli ki:

1.2.2. Jelzési üzenetek börsztös érkezése során az alábbi tulajdonságokat figyelhetjük meg (a hívások felépítési, illetve –elbontási fázisában):

1.2.2.1 Az első hívás felépítési ideje mindig 80-100%-kal hosszabb ideig tart, mint a börsztben közvetlenül utána következő hívásé.

Ez a kivételes jelenség azért következik be, mert az első hívás előtt egy IP-ről ATM-re történő címfeloldási konverzióra van szükség, esetleg még a jelzés csatornát is ki kell építeni végponttól végpontig, majd ezen információk tárolódnak egy ideiglenes tárolóban, s így a következő hívásnál már erre a lépésre nincs szükség. Természetesen, egy nagyobb börsztnél, a várakozási idők miatt, egy újabb graduális növekedés figyelhető meg a hívás felépítési időkben, a 3. hívástól kezdődően.

1.2.2.2 Ellentétben az egyensúlyi állapottal (1.1 tézis), itt a börszt első felében a SETUP késleltetés alacsonyabb mint a CONNECT késleltetés, de a két görbe gradiense ($tg \alpha$) azonos (és növekvő) mindaddig amíg a (11)-es egyenlőtlenség fennáll, majd a CONNECT késleltetés görbéje irányt vált és csökkenni kezd a $tg(-\alpha)$ gradienssel, ahol $0 < \alpha < \pi/2$.

Mivel minden üzenet azonos prioritással rendelkezik, ezért mire az első CONNECT üzenet visszatér a célállomástól, a hálózati csomópont puffere már jócskán feltelik a börsztben elküldött SETUP üzenetekkel, ezáltal érthető, hogy a CONNECT késleltetés hosszabb lesz, mint a SETUP-é. Később, amikor a SETUP üzenetek kiürülnek a kapcsolóból, a CONNECT késleltetés graduálisan lecsökken.

1.2.2.3 Börsztös hívások felépítési ideje (T_C) mindig hosszabb, mint a determinisztikus hívásoké. Továbbá, a T_C az alábbiak szerint függ a börszt méretétől:

Ha a börszt mérete (b) \leq a jelzési processzor max puffere mérete (BS), akkor

$$\begin{aligned} T_{Ci}(b) &= 2 \cdot T_{C2} && \text{ha } i=1, \\ T_{Ci}(b) &= T_{C2} + (i-2) \cdot tg \alpha(b), && \text{ha } 1 < i \leq b \leq BS, \end{aligned} \quad (12)$$

különben $T_{Ci}(b) = 0$, ha a hívás blokkolt,

vagy $T_{Ci}(b) = T_{C2} + T_{303} + (j-1) \cdot tg \alpha(b)$, ha a hívás megismétlődött,

ahol:

i = az üzenet pozíciója a börsztben, $i = 1, 2, \dots, b$ (ha $i=1$, lásd. 1.2.2.1 altézist);

$tg \alpha(b)$ = a T_C görbe érintője, amely függ a börszt ‘ b ’ méretétől;

T_{303} = a SETUP üzenet megismétlési időzítője, standard értéke 4 sec, lásd. [Q2931];

T_{C2} = a börszt 2. hívásának felépítési ideje (ez a *minimális* érték);
 j = az üzenet pozíciója az úraküldött börsztben, $j = 1, 2, \dots, b$.

1.2.2.4 Hasonlóképpen, a börsztös hívások *elbontási késleltetése* (T_{RN}) így írható le:

Ha a börszt mérete (b) \leq a jelzési processzor *max puffer mérete* (BS), **akkor**

$$T_{RNi}(b) = T_{RI} + (i-1) \cdot tg \beta(b), \quad (13)$$

különben $T_{RNi}(b) = 0$,

ahol:

i = az üzenet pozíciója a börsztben, $i = 1, 2, \dots, b$;

$tg \beta(b)$ = a T_{RN} görbe érintője, amely függ a börszt 'b' méretétől;

A T_C és T_{RN} görbék érintői ($tg \alpha$, $tg \beta$) az alábbi paraméterektől függenek: börszt mérete, üzenet típusa, hívás profilja, processzor befogadóképessége. Mindezen függőségeket a disszertációban táblázatos formában megtaláljuk (lásd. [D-4]).

Az 1. tézis validációja

Az 1. tézisben bemutatott sajátos tulajdonságok egy nagyon "népszerűtlen" kutatási munka gyümölcsei, hiszen a mérési munkálatok elhúzódtak több mint négy éven keresztül. A kutatói csoport tagjai is kicserélődtek, miközben 4 különböző generációs kapcsolót sikerült letesztelni, ráadásul két különböző tesztelő berendezést is használtunk (*Agilent BSTS 75000* és *Navtel iWatch 95000*), amelyek eredményeit össze kellett hangolni, de ugyanakkor adott egy plusz biztonságot az eredmények valóságát illetően. Továbbá, minden egyes mérést 10-szer ismételtünk meg, egy adott mérés 60-100 hívást tartalmazott. A minimum-, maximum- és átlagértékeket, valamint a szórásnégyzetet és szórást ezen mérések egyesített halmazán számítottuk ki. Végül, de nem utolsósorban, tőlünk függetlenül 4 másik kutatócsoport is végzett méréseket, melyek eredményeit a [Nie97], [Nov99], [Far01] és [Mau01] ismertetik. Ezen kutatócsoportok többnyire más ATM kapcsolókat teszteltek, de eredményeik hasonlóak, megerősítve állításaink általános érvényét.

Ha egy adott mérés jelzési üzeneteit elhelyezzük a "populációs-diagram"-ban (lásd. 1.2.1 tézis), egy azonnali "vizuális" betekintést nyerünk a lejátszódó folyamatok részleteibe, ami nem más, mint egy elsőfokú validációja eredményeinknek. Továbbá olyan belső összefüggéseket is feltár ez az ábrázolásmód, amely nem derülne ki ha csak a hívás felépítési és –elbontási időket mérjük (pld. megismételt üzenetek, egy adott állapotban nem várt üzenetek, adatkapcsolati szintű folyamszabályozás hatása a mért eredményekre, stb.).

Eredményeink jelentősége, hogy részletes elemzést nyújt a hívás felépítési és –elbontási idők komponenseiről, és kiemeli azon összetevőket, amelyek a processzor architektúrájától és feldolgozási sebességétől függetlenek, azaz az ATM jelzés-forgalom sajátosságai.

Egy új hívásfeldolgozó modell készítése ATM hálózatokra

Egy új ATM hívás-modell kidolgozását (UNI és PNNI interfészekre) az tette szükségessé, hogy a távközlési szolgáltatók többnyire ATM technológián alapuló xDSL hálózataikban egy új szolgáltatás, az IP telefónia (VoIP) bevezetését tervezik, viszont az ATM hálózataikban meglévő sáv szélességek nem teszik lehetővé, hogy minden hangcsatornának előre lefoglalt sáv szélességet biztosítsanak. Egyik lehetséges alternatíva a dinamikus sáv szélesség-foglalás bevezetése, amely ATM jelzésrendszert feltételez.

2. tézis Kidolgoztam két különböző hívás-modellt az ATM jelzésüzenetek feldolgozásának leírására, egyet szimulációs vizsgálatokra (2.1 tézis) és egyet analitikus elemzések eszközeül (2.2 tézis). Ez utóbbi nyilvánvalóan egy leegyszerűsített modellje az előzőnek.

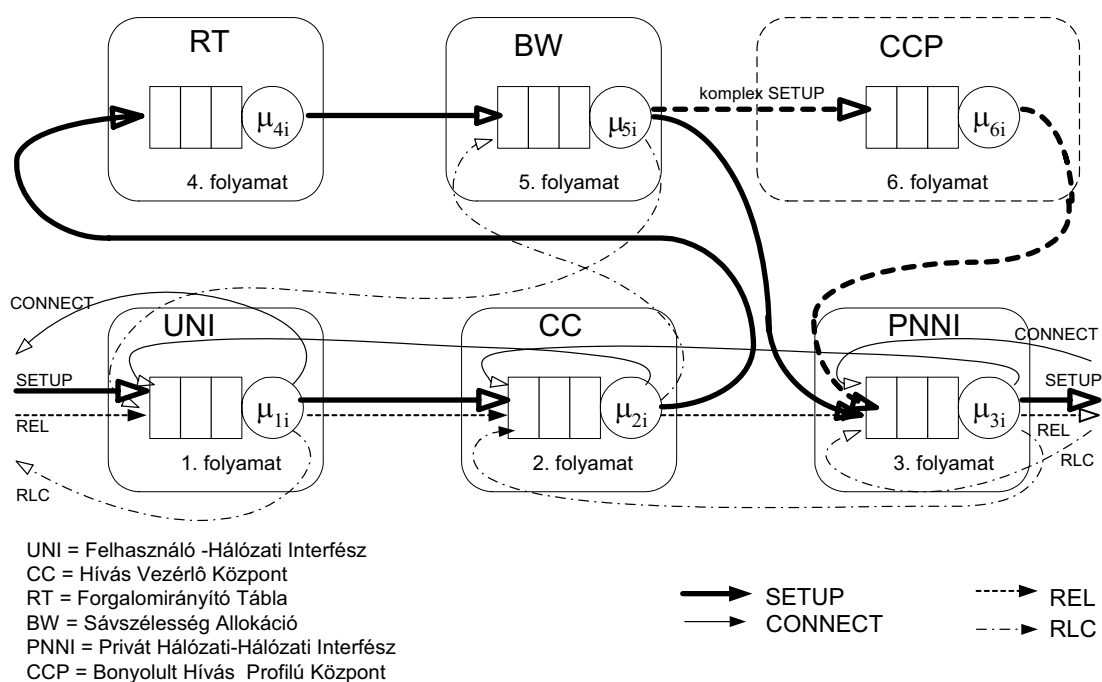
Napjainkban szinte lehetetlen hozzáférni UNI vagy PNNI jelzésprotokolt használó kiterjedt ATM hálózatokhoz. Néhány kezdetleges, jelzésekre irányuló mérési eredmény a *TEN-155* Pán-Európai ATM hálózaton született [Nov99], amelyet természetesen felhasználtunk szimulációs eredményeink igazolásához. Ez a hálózat hét ATM csomópontot tartalmaz, a leghosszabb útvonal (a hálózat átmérője) négy kapcsolón halad át. Így nem maradt más hátra, mint szimulációval modellezni ennél jóval nagyobb méretű hálózatokat.

2.1 tézis Kidolgoztam egy új ATM hívás-modellt (lásd. 4. ábra), továbbá egy algoritmust ezen modell paramétereinek beállítására. [J-4], [C-9], [C-10], [D-5]

A modell architektúrája

2.1.1. Kiindulva a [Q2931], [UNI40], [PNNI] szabványokból és az 1. tézisben bemutatott mérési eredményekből, egy új hívásfeldolgozó modellt dolgoztam ki, amely ráilleszthető bármely létező ATM kapcsoló jelzési aktivitására.

A [Q2931], [UNI40] és [PNNI] szabványok nem határoznak meg semmiféle hívás modellt, csak leírják az ide vonatkozó jelzési üzenetek formátumát, tartalmát és a kommunikációs protokoll lehetséges eseteit. Az 1.1 tézis viszont elénk tár néhány részletet. Így az 1.1.2, 1.1.4 és 1.1.6 pontok alapján kiderül, hogy az átlagos hívás elbontási késleltetés mindig alacsonyabb, mint az átlagos hívás felépítési késleltetés, továbbá fenti két paraméter aránya változik az érkezési intenzitás függvényében. Emiatt két különböző mechanizmust is megvizsgáltam az általam javasolt modellben: *FIFO*, illetve prioritásos érkezési sorokat. Ezen túlmenően, egy adott csomópontban több *processzor fázist* különböztettem meg az elvégzendő feladatok bonyolultságának függvényében (lásd. 4. ábra), például egy *alapbeállítás* *SETUP* üzenet öt különböző fázist jár be, míg egy *komplex* *SETUP* üzenet mind a 6 folyamatot végigjárja, ellenben a *CONNECT* és *RELEASE* üzenetek mindössze 3-at "látogatnak meg".



4. ábra Egy ATM jelzési csomópont új hívás feldolgozási modellje

Ezen elkülönített processzor fázisok a következők: 1) az *UNI* folyamat, a bejövő üzenetek dekódolására, 2) a *CC* folyamat, az egy adott híváshoz tartozó objektumok létrehozására és frissítésére, 3) a *PNNI* folyamat, a kimenő üzenetek kódolására, 4) a forgalomirányító *RT* folyamat, 5) a kimenő link sávszélesség allokációját, illetve felszabadítását végző *BW* folyamat, valamint 6) a bonyolult hívásprofilokat, puffer allokációt és QoS mechanizmusokat feldolgozó *CCP* folyamat. A modell ATM-es sajátosságait (korábbi modellektől való eltérését) a következők adják meg: különböző sávszélesség rendelhető egy adott híváshoz, különböző üzenetek más és más folyamatokat járnak be a processzorban, valamint különböző kiszolgálási sebességek rendelhetőek egy adott üzenethez a típusa, útvonala, PNNI hierarchiája és QoS követelményei alapján.

Két hasonló hívás-modell került kidolgozásra a [Wu97] és [Gel97] cikkekben. A [Wu97]-es modell még bonyolultabb, mint a mienk, és ráadásul a CBR, VBR és UBR forgalmi osztályokhoz különböző kiszolgálási sebességeket rendel (ami az 1.1.4 altézis alapján szükségtelen), továbbá egyik modell sem veszi figyelembe a hívás elbontást, holott ezen esemény 15-20%-kal növeli az átlagos hívás felépítési időt (lásd. 1.1.5 altézis). Ráadásul, a [Gel97] csomópont modell egy egyszerű *FIFO* sorból áll, amely nem tud különbséget tenni a *SETUP* és *RELEASE* késleltetések között.

A modell paramétereinek beállítása

A feladatot a következőképpen fogalmazhatjuk meg:

- Az ismert változókat az egyes üzenetek mért késleltetései, a kapcsoló hívás-sikerességi rátája, a hívás felépítési idők és –elbontási késleltetések (lásd. 2. ábra);
- Ismeretlenek az egyes folyamatok kiszolgálási sebességei (μ_i) és a pufferek méretei (BS_i), valamint az elveszett üzenetek ismétlési késleltetése ($RDLM$)³.

Megmutattam, hogy a folyamatok *kiszolgálási ideje* a (14)-es egyenletrendszerből kiszámítható. A folyamatok *puffer méretét* (BS_i) és az *RDLM* paramétert szimuláció útján határozzuk meg.

2.1.2. Kidolgoztam az alábbi négy lépésből álló algoritmust a μ_i , $BS = \sum_{i=1}^6 BS_i$ és *RDLM* változók kiszámítására:

I. Határozzuk meg a kiszolgálási sebességeket a (14)-es egyenlet-rendszer alapján:

$$\begin{aligned} \sum_{i=1}^5 \frac{1}{\mu_i} &= \min \text{ SETUP késleltetés} & \sum_{i=1}^6 \frac{1}{\mu_i} &= \min \text{ 'komplex' SETUP késleltetés} \\ \sum_{i=1}^3 \frac{1}{\mu_i} &= \min \text{ CONNECT késleltetés}^4 & \sum_{i=1,2,3,5} \frac{1}{\mu_i} &= \frac{1}{2} \cdot \min(\text{RELEASE idő} + \text{RELEASE késleltetés}) \quad (14) \\ 1/\mu_1 &= a/\mu_2 = 1/\mu_3, \end{aligned}$$

ahol $a \in (1,2]$ a korrekciós tényező. Módosítsuk $a \in (1,2]$ -t addig amíg: $|T_{CN}^{meas} - T_{CN}^{sim}| < 0.05 \cdot T_{CN}^{meas}$ és $|T_{RN}^{meas} - T_{RN}^{sim}| < 0.05 \cdot T_{RN}^{meas}$, miközben $\gamma_R = 1$.

II. Állítsuk be azon (szűk keresztmetszetű folyamat) puffer hosszát (BS_{j^*})-t, ahol $j^* = \min\{j \in \{1, \dots, 6\} \mid \forall k \in \{1, \dots, 6\} : q_j \geq q_k\}$ úgy, hogy az érkezési intenzitás küszöbértékeire

³ *RDLM* az a késleltetés, amely alatt egy elveszett üzenet (egyéb mint *SETUP*) újragenerálódik a kapcsolatban.

⁴ vagy a \min *RELEASE* késleltetés

fennálljon az $|\lambda_{Th}^{sim} - \lambda_{Th}^{meas}| < 0.05 \cdot \lambda_{Th}^{meas}$ összefüggés, ahol $\lambda_{Th} = \min_{i \in I} \{\lambda_i \mid \frac{T_{CN,i+1} - T_{CN,i}}{\lambda_{i+1} - \lambda_i} \geq 1\}$, az

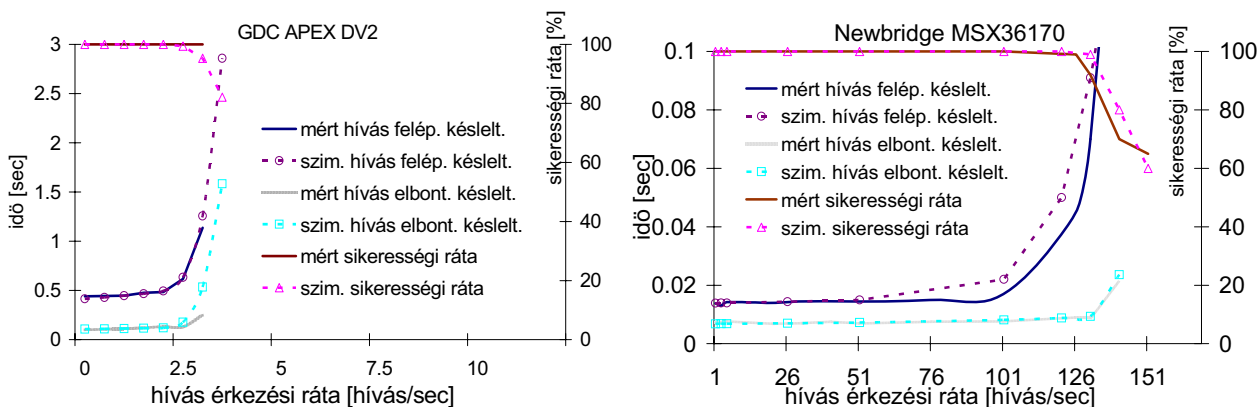
“ I ” a mért vagy szimulált adatok véges halmaza, és ‘ q_j ’ a ‘ j ’ sorban észlelt igények száma.

III. Állítsuk be a fennmaradó pufferek hosszát, miközben a hívás sikerességi rátára fennáll az $|\gamma_R^{meas} - \gamma_R^{sim}| < 0.05 \cdot \gamma_R^{meas}$, $\forall \lambda > \lambda_{Th}$ összefüggés.

IV. Határozzuk meg az RDLM paraméter értékét úgy, hogy minimálisra csökkenjen a mért és szimulált eredmények közötti eltérés a túlsordulás tartományban ($\lambda > \lambda_{Th}$).

A 2.1 tézis validációja

Elsősorban bebizonyítottam, hogy mind a négy általunk vizsgált ATM kapcsoló esetén a szimulációs eredmények ráilleszthetőek a mért eredményekre mindhárom paraméter (T_{CN} , T_{RN} , γ_R) együttes figyelembevételével (lásd. 5. ábra). A BS puffer méretet és az RDLM paramétert úgy határoztuk meg, hogy a szimulált és mért eredmények közötti eltérés kisebb legyen mint 5%.



5. ábra Az új modell validációja alapbeállítású SETUP üzenet esetén:
a) GDC APEX DV2 kapcsolóra; b) Newbridge MSX 36170 kapcsolóra

Ahhoz, hogy a hívott fél késleltetési idejének káros hatását kiküszöböljük, az (1 forrás; 1 kapcsoló; 1 nyelő) konfiguráció helyett bevezettük a (10 forrás; 1 kapcsoló; 10 nyelő) konfigurációt. Szimulációs vizsgálatainkban úgy determinisztikus, mint Poisson érkezést használtunk. Az átlagértékeket mindannyiszor egy 10000-es mintából származtattuk. Az így hitelesített modellünket egy konkrét valós környezet ellenében is leteszteltük, ehhez a TEN-155 európai kutatói hálózaton mért eredmények szolgáltattak alapot [Nov99]. Az ott bemutatott eredmények egy 5-6 alkalommal megismételt „ping” sorozatból származtatott átlagok. Ezzel szemben mi 3000 hívásból számoltunk átlagot, 1 hívás/sec intenzitás mellett. Az összehasonlított eredményeket a disszertációm [D-5]-ös fejezetében mutattam be. Az esetek több mint 80%-ban a szimulált eredmények egybeesnek a mért eredményekkel, a fennálló eltérések a pontatlannak számító “ping-típusú” mérésekből származnak.

2.2 tézis Készítettem egy egyszerűsített jelzési folyam-modellt analitikus vizsgálatok céljából, továbbá egy heurisztikus egyenletet is felállítottam a T_C és T_{RN} értékek meghatározására. [C-6], [C-9], [D-5]

A 2.2.1 altézisben megmutattam, hogy a 2.1 tézisben bemutatott folyam-modell leegyszerűsíthető oly módon, hogy ezáltal analitikus módszerekkel megoldhatóvá válik. Kiindulva abból a tényből, hogy a kaszkádolt sorban a T_C idő mindig hosszabb, mint bármely más hálózati struktúrában (lásd. 3.1 tézist a későbbiekben), ezért itt csak a kaszkádolt sorra adtam analitikus megoldást. Az általam

választott analitikus megoldási séma egy sajátos esete a már ismert BCMP modellnek (lásd. [BCMP75]), ahol a csomópont modell egy egyszerű várakozási sorból és egy ekvivalens kiszolgáló egységből áll. A jelzési üzenetek itt megfelelnek a BCMP osztályoknak, és a kötelező osztályváltásra is sor kerül a végpontokban: *SETUP* (1. osztály) átváltozik *CONNECT*-té (2. osztály), *CONNECT*-ből lesz *RELEASE* (3. osztály), *RELEASE*-ből *RELEASE COMPLETE* (4. osztály). Az egyetlen külső forrás, amely 1. osztálybeli hívásokat (azaz *SETUP* üzeneteket) generál az első csomópontban feltételezésünk szerint λ_N paraméterű Poisson érkezési folyamat.

2.2.1. Megmutattam, hogy egy 'r' csomópontos kaszkádolt hálózatban a hívás felépítési idő analitikusan közelíthető, ha a csomópontokat egy $M/M/1/K$ modellel írjuk le és a hívástartási időt egy $M/M/\infty$ sorbanállási rendszerrel modellezzük.

Ebben a sorbanállási hálózatban a terhelés eloszlás elemezhető, ha az alábbi forgalmi egyenletrendszer megoldjuk:

$$e_{js} = (p_{0,js} - q_{0,js}) + \sum_{i,u} e_{ir} \cdot (p_{iu,js} - q_{iu,js}), \quad i, j = 1, \dots, r+2; \quad u, s = 1, 2, 3, 4 \quad (15)$$

ahol: $p_{0,11} \neq 0$, $p_{0,js} = 0 \quad \forall j, s > 1$. A forgalomirányítási valószínűség minden csomópontban eleget tesz az alábbi feltételnek: $\sum_{j,s} (p_{iq,js} + q_{iq,js}) = 1$. Ezen egyenletrendszer mátrix alakja egy $[s_{\max} \cdot (r+2) \times s_{\max} \cdot (r+2)]$ méretű forgalomirányítási mátrix.

Feltételezzük, hogy: $p_{iu,js} > 10 \cdot q_{iu,js}$, $p_{iu,js} + q_{iu,js} = 1$. Akkor a (15)-ös egyenletrendszerben minden tag, amely legalább kettes 'q' szorzatot tartalmaz (pld. $q_{iu,js} \cdot q_{kv,lw}$), elhanyagolható. Továbbá, bevezetjük az alábbi jelölést: $q_{\max} = \max_{i,j,u,s} \{q_{0,11}, q_{iu,js}\}$.

Akkor a legbonyolultabb tag (e_{14}) az összes e_{js} közül az alábbi formában közelíthető:

$$e_{14} \geq \prod_{\substack{i=0 \\ s=1,3}}^r p_{is,(i+1)s} \cdot \prod_{\substack{i=0 \\ s=2,4}}^r p_{(i+1)s,is} - q_{\max} \cdot \left(\sum_{\substack{i=0 \\ s=1,3}}^r p_{is,(i+1)s} + \sum_{\substack{i=0 \\ s=2,4}}^r p_{(i+1)s,is} \right)$$

Fenti kifejezésben találunk kell egy feltételt a q_{\max} -ra úgy, hogy a negatív előjelű tag elhanyagolhatóvá váljék:

$$q_{\max} \cdot \left(\sum_{\substack{i=0 \\ s=1,3}}^r p_{is,(i+1)s} + \sum_{\substack{i=0 \\ s=2,4}}^r p_{(i+1)s,is} \right) \leq 10^{-2} \cdot \prod_{\substack{i=0 \\ s=1,3}}^r p_{is,(i+1)s} \cdot \prod_{\substack{i=0 \\ s=2,4}}^r p_{(i+1)s,is}$$

Ebből kifejtve a q_{\max} -t, majd helyettesítve a nevezőt annak felső korlátjával, illetve a számlálót annak alsó korlátjával, egy még szigorúbb feltételhez jutunk:

$$q_{\max} \leq \frac{10^{-2} \cdot (p_{\min})^{4(r+1)}}{4 \cdot (r+1)}$$

ahol, $p_{\min} = \min \{p_{iu,js} > 0, \forall i, j = \overline{0, r}; \forall u, s = \overline{1, 4}\}$. Példa: $p_{\min} = 0.99$, $r = 8$, $q_{\max} \leq \frac{10^{-2} \cdot (0.99)^{36}}{36} = 2 \cdot 10^{-4}$.

Fenti példa egy valós esetet tükröz, enyhe túlterhelést mutat ($p_{\min} = 0.99$). Ha elfogadjuk ezen feltételt, akkor a q_{\max} -t tartalmazó kifejezést elhanyagolhatjuk és következésképpen ugyanahhoz a (16)-os egyenlethez jutunk, mint az $M/M/1$ -es sorbanállási modellt használva $M/M/1/K$ helyett.

Néhány további lépés után eljutunk az alábbi egyenlethez, ahol 'r+1' csomópontban geometriai eloszlást találunk, míg a hívás-tartást modellező csomópont poisson eloszlású lesz.

$$\rho_i = \begin{cases} \frac{\sum_{s=1}^4 \lambda_N \cdot e_{is}}{\mu_{is}}, & i = 1, \dots, r+1 \\ \frac{\lambda_i}{\mu_i}, & i = r+2 \end{cases}; \quad (16)$$

A (17)-es egyenlet viszont az alábbi eltérést mutatja az M/M/1-es esethez képest:

$$p_i(k) = P(n_i = k) = \begin{cases} \frac{(1 - \rho_i) \cdot \rho_i^k}{1 - \rho_i^{K+1}}, & i = 1, \dots, r+1; k \leq K \\ 0, & i = 1, \dots, r+1; k > K \\ \frac{e^{-\rho_i}}{k!} \cdot \rho_i^k, & i = r+2 \end{cases} \quad (17)$$

ahol n_i az i pontban található összes üzenet. A stabilitási feltétel: $\rho_i < 1$. Mint azt az 1.1.5 altézisben láthattuk, a véges időtartamú hívások nem befolyásolják a T_C időt, ezért az ($i=r+2$)-es csomópont hatását egyszerűen elhanyagolhatjuk. A megoldásunk viszont csak akkor egyszerűsödik le szorzat-formára, ha $\mu_{is} = \mu_i \forall s=1,2,3,4$ (lásd. a (16)-os egyenletrendszer), ez a feltétel azonban ebben a formában nem teljesül.

Ezért használjuk fel az 1.1.2 altézis eredményét és vezessük be az alábbi empirikus kapcsolatot a

kiszolgálási idők között: $\frac{1}{\mu_{i1}} = 3 \cdot \frac{1}{\mu_{i2}} = 3 \cdot \frac{1}{\mu_{i3}} = 12 \cdot \frac{1}{\mu_{i4}}$, $i = 1, \dots, r$, majd jelöljük: $\frac{1}{\mu_{i1}} = \alpha_i$, amely nem más mint a minimum *SETUP késleltetés* (lásd. 2.1.2 altézist). Továbbá, mivel $\frac{1}{\mu_i^C} = \frac{1}{\mu_{i1}} + \frac{1}{\mu_{i2}} = \frac{4}{3} \cdot \alpha_i$, ezért kifejtethetjük ρ_i -t a μ_i^C függvényében. Végül az i sorban található

összes igény számának várható értéke kifejtethető az ismert Little-formula segítségével:

$$E(T_i) = \frac{1}{\lambda_i} \cdot \sum_{k=0}^K k \cdot p_i(k) = \frac{1}{\lambda_i} \cdot \left[\frac{\rho_i}{1 - \rho_i} - \frac{(K+1) \cdot \rho_i^{K+1}}{1 - \rho_i^{K+1}} \right], \quad i = 1, \dots, r+1 \quad (18)$$

A stabilitási feltétel továbbra is $\rho_i < 1$. Ebből a képletből kifejtethetjük az 'r' csomópontos kaszkádolt sor hívás felépítési idejét és –elbontási késleltetését (tegyük fel az *ideális esetet*, azaz hogy minden hívás sikeres, és feltételezzük, hogy az összes csomópont azonos típusú).

$$\begin{aligned} E(T_C) \Big|_{r, \text{switches}} &= \sum_{s=1,2}^{r+1} \sum_{i=1} E(T_i) = 2 \cdot r \cdot E(T_r) + E(T_{r+1}), \\ E(T_{RN}) \Big|_{r, \text{switches}} &= \frac{1}{2} \cdot \sum_{s=3,4}^{r+1} \sum_{i=1} E(T_i) = r \cdot E(T_r) + \frac{1}{2} \cdot E(T_{r+1}), \end{aligned} \quad (19)$$

ahol T_r az az idő amit egy üzenet egy csomópontban eltölt, $E(T_i) = E(T_j) = E(T_r) \forall i, j = 1, \dots, r$, mialatt a T_{r+1} a végpontokban eltöltött időt jelöli ($i=r+1$). A 'K' maximális puffer méretet úgy kell meghatározni, hogy a kapott eredmény lehető legjobban megközelítse a mért eredményeket. Az M/M/1 végtelen pufferű eset megoldását a [C-6]-ban írtam le, amely valójában egy sajátos esete a (18)-as egyenletnek, azaz ha $K \rightarrow \infty$, akkor a második tag tart a zéróhoz. A T_C -re így elért megközelítés pedig elfogadható mindaddig amíg a jelzésforgalmi terhelés $\lambda_N / \mu^C < 0.65$.

Ha viszont a hálózatunk nem homogén, azaz több különböző gyártó terméke van jelen, vagy különböző komplexitású hívásokra kerül sor, és ráadásul nem csak egy PNNI hierarchia szintű a hálózat, akkor a (19)-es egyenlet sajnos csak nagyon körülményesen alkalmazható. Ezért erre az esetre egy külön megközelítő becslést dolgoztam ki (amelyet pld. egy hálózat dimenzionálásánál használhatunk fel, ha a T_C és T_{RN} értékek előre ismertek):

2.2.2. Kimutattam, hogy a (10)-es egyenlőtlenség, amely ‘r’ kaszkádolt kapcsolón áthaladó hívás felépítési késleltetésre vonatkozik, közelíthető az alábbi heurisztikus képlettel:

$$E(T_{CN})|_{r'} \approx \begin{cases} (1+s) \cdot \frac{m+1}{2} \cdot [1-0.1 \cdot \log(r)] \cdot \exp\left(\frac{\lambda_N}{r \cdot \mu_0^C}\right) \cdot \sum_{i=1}^P (n_i \cdot \bar{T}_{CN}^{\text{type}i}), & r = \sum_{i=1}^P n_i, \mu_0^C = \min_{i=1..P}(\mu_i^C), \lambda_N < 0.9 \cdot \mu_0^C \\ (1+s) \cdot \frac{m+1}{2} \cdot [1-0.1 \cdot \log(r)] \cdot \frac{C \cdot \left(\frac{\lambda_N}{\mu_0^C}\right)^2}{\sqrt{\exp\left(\frac{\lambda_N}{\mu_0^C}\right)}} \cdot \sum_{i=1}^P (n_i \cdot \bar{T}_{CN}^{\text{type}i}), & \lambda_N \geq 0.9 \cdot \mu_0^C \end{cases} \quad (20)$$

Hasonlóképpen, a T_{RN} becslése az alábbi módon történhet:

$$E(T_{RN})|_{r'} \approx \begin{cases} [1-0.1 \cdot \log(r)] \cdot \exp\left(\frac{\lambda_N}{r \cdot \mu_0^R}\right) \cdot \sum_{i=1}^P (n_i \cdot \bar{T}_{RN}^{\text{type}i}), & r = \sum_{i=1}^P n_i, \mu_0^R = \min_{i=1..P}(\mu_i^R), \lambda_N < 0.9 \cdot \mu_0^R \\ [1-0.1 \cdot \log(r)] \cdot \frac{C \cdot \left(\frac{\lambda_N}{\mu_0^R}\right)^2}{\sqrt{\exp\left(\frac{\lambda_N}{\mu_0^R}\right)}} \cdot \sum_{i=1}^P (n_i \cdot \bar{T}_{RN}^{\text{type}i}), & \lambda_N \geq 0.9 \cdot \mu_0^R \end{cases} \quad (21)$$

A $\log(r)$ -t tartalmazó kifejezés az üzenetek átfedéséből adódó hatást tükrözi, az m a PNNI hierarchia szintet jelöli, $s=0$ a kötelező IE -re (alapbeállítású $SETUP$ üzenet), $0 < s < 1$, P a különböző típusú ATM kapcsolók száma, és n_i az egy adott típusú kapcsolók száma. A (μ_i^C, μ_i^R) páros a kapcsoló ekvivalens kiszolgálási intenzitása hívás felépítés és –elbontás esetén, ahol $\mu_i^C < \mu_i^R$, míg λ_N a hívás intenzitás. Végül,

$$C = \exp\left(\frac{0.9}{r}\right) \cdot \frac{\sqrt{\exp(0.9)}}{(0.9)^2}$$

az ú.n. normalizáló faktor. Megfigyelhető, hogy a T_{RN} független a hívásprofil bonyolultságától és a PNNI hierarchiától egyaránt.

A 2.2 tézis validációja

A 2.2 tézis mindkét formuláját igazoltuk mért, illetve szimulációs eredmények ellenében $r \leq 4$ kaszkádolt kapcsoló esetén. A 2.2.1 altézisben bemutatott modellt hálózattervezésnél használhatjuk, és az első megközelítő értékek becslésére alkalmas. Hátránya a módszernek, hogy csak az átlagértékekre ad becslést, és az ATM jelzésekre jellemző sajátosságokat elveszti azáltal, hogy egyszerű $FIFO$ sort használ a csomópont modellezésnél. Azonkívül a modell prioritásos sor kezelésére sem alkalmas, hiszen ekkor a megoldás elveszíti a szorzat-formát. A forgalomirányító mátrix két végberendezés és két kaszkádolt kapcsoló esetén is már $[16 \times 16]$ -os méretű, minden további csomópont esetén a mátrix további 4 sorral és 4 oszloppal bővül, mialatt egy forrás 8 újabb sort, illetve 8 oszlopot eredményez. Az $M/M/1/K$ véges pufferű modell felső becslést ad a hívás felépítési időre kaszkádolt sort esetén, ha $\lambda_N / \mu^C > 0.5$, továbbá ez a modell akkor is pontatlan, ha a hívás elbontási késleltetést becsljük vele ($T_C \approx 2T_{RN}$ -t eredményez). Ezzel szemben, a 2.2.2 altézis (20)-as és (21)-es egyenletei a gyakorlatban kielégítő becslést nyújtanak. Ezen becsléseket több különböző típusú kaszkádolt kapcsoló esetén validáltuk a mért, illetve szimulált eredményekkel.

Kiterjedt ATM hálózatok jelzésforgalmának teljesítmény-analízise

3. tézis Elvégeztem valós méretű ATM hálózatok jelzésforgalmának teljesítmény-analízisét. [C-4], [C-10], [J-4], [D-6]

A 2.1 tézisben bemutatott modell eléggé bonyolult, analitikus vizsgálatokra nem alkalmas igazán, különösen ha prioritásos sorokra kellene alkalmazzuk vagy a hálózatunk nagy kiterjedésű. Ezért ezen modellt a továbbiakban *szimulációs vizsgálatokban* alkalmaztam a hívás felépítési idő, az –elbontási késleltetés és a csomópontonkénti jelzés-terhelés becslésére valós méretű hálózatokban. Amint azt az 1.1.7.-es altézisben láthattuk, az ATM jelzési csomópontok által okozott üzenet-késleltetés additív jellegű. Következésképpen, egy hívás késleltetése annál nagyobb minél több csomóponton haladt át az adott hívás (üzenet). Tehát, egy kisszámú kapcsolón mért késleltetés felhasználható hasonló típusú kapcsolókból álló nagykiterjedésű hálózat teljesítményének *becslésére*. Ezáltal például meghatározható az ITU-T ajánlásokban [ITU97] szereplő maximális késleltetésnek megfelelő felső korlát (hálózati átmérő, Φ).

3.1 definíció Definiáltam a *hálózati átmérőt*, úgy mint a legrövidebb utat a hálózat két legtávolabbi csomópontja között:

$$\Phi = \max \{L(i, j) \mid L(i, j) < L'(i, j), \forall i, j \in \{1, \dots, N\}\}, \quad (22)$$

ahol $L(i, j)$ az i és j csomópontok közötti útvonal hossza.

3.2 definíció Definiáltam az N -csomópontos hálózat *átlagos felhasználói sűrűségét* (\bar{D}_N):

$$\bar{D}_N = \left(1 - \frac{N_{tr}}{N}\right) \cdot \frac{\bar{L}}{\Phi}. \quad (23)$$

ahol N_{tr} a tranzit-pontok száma a hálózatban (azaz azon pontok száma, amelyekhez nem csatlakoznak közvetlenül végfelhasználók). A felhasználói sűrűség egy fontos hálózati szintű paraméter, hiszen megmutatja a végfelhasználók eloszlását a csomópontok között. Általában $0 < \bar{D}_N \leq 1$, minél nagyobb a \bar{D}_N , annál "sűrűbb" a hálózat. Példáinkban ($N = 4, 30, 35$) ezen értékek $\bar{D}_4 = 0.93$, $\bar{D}_{30} = 0.59$, $\bar{D}_{35} = 0.45$, míg egy izolált kapcsoló esetén $D_1 = 1$. A (23)-as képlet alkalmazhatósága attól is függ, hogy mennyire könnyedén sikerül az átlagos útvonalhosszat meghatározni (\bar{L}). A mi esetünkben az \bar{L} -t szimuláció útján határoztuk meg.

3.3 definíció Definiáltam egy hálózati kapcsoló *jelzés terhelésének relatív szórását* az alábbiak szerint, ahol λ_i az adott kapcsoló jelzés-terhelése:

$$\theta_i = \frac{\lambda_i - \frac{1}{N} \cdot \sum_{i=1}^N \lambda_i}{\max_{1 \leq i \leq N} (\lambda_i)}; \quad -1 \leq \theta_i \leq 1; \quad 1 \leq i \leq N \quad (24)$$

Szimulációval megvizsgáltam $N = \{2, 3, \dots, 10\}$ kaszkádolt ATM kapcsoló hívás felépítési idejét és –elbontási késleltetését a jelzés-terhelés függvényében. Ebből $N = \{2, 3, 4\}$ esetén mérésekkel validáltam az eredményeket. Továbbá szimulációval becslést adtam az alábbi hálózati topológiákra: $N=4$ -es *teljes háló*, $N=7$ csomópontos 4-es *gyűrűvel* rendelkező struktúra, majd egy $N = \{30, 35\}$ csomópontos ATM hálózatot is megvizsgáltam, amely egy *tipikus xDSL topológiát* tartalmaz, kb. 25 DSLAM-mel (lásd. 6. ábra). Ez utóbbiak magukba foglalják az összes ismert alapvető topológiát (multiplexerek a hozzáférési hálózatban, redundáns linkek, gerinchálózati gyűrűs topológia, stb.). Továbbá ezekben a hálózatokban megtalálunk minden útvonalhosszat 2-től 8 kapcsolóig, így össze is tudtam hasonlítani az itt becsült T_{CN}^L hívás felépítési- és T_{RN}^L hívás elbontási késleltetést a korábban a kaszkádolt kapcsolókra elért eredményekkel (ahol $L(i, j) = r = \{1, \dots, 8\}$).

3.1 tézis Kimutattam, hogy egy ‘ r ’ csomópontos kaszkádolt hálózat alkalmas egy N -csomópontos ($N > r$, $\Phi = r$) általános topológiájú hálózat alábbi hívás-teljesítmény paramétereinek becslésére: T_{CN} , T_{RN} , γ_R .

Először egy *homogén hálózatot* vizsgáltam meg, ahol minden csomópont azonos típusú. Kimutattam, hogy a T_{CN}^L és T_{RN}^L paraméterek minimum értékei bármely L útvonalhosszra azonosak úgy a kaszkádolt, mint az általános topológia esetén. Az átlagértékek, illetve maximum értékek esetében viszont a kaszkádolt topológiára enyhén magasabb értékeket érünk el, azaz 1-5%-kal növekednek az átlagértékek, illetve 15-20%-kal a maximális értékek, amennyiben a hálózati terhelés $0 < \lambda_N < \lambda_N^{\max}$, ahol λ_N^{\max} a maximális hálózati szintű hívás érkezési intenzitása úgy, hogy nincs blokkolt hívás a hálózatban ($\gamma_R=1$), N a csomópontok száma. Szinte egyértelmű, hogy a λ_N^{\max} az alábbiak szerint függ a hálózati topológiától:

$$\lambda_1^{\max} \approx \lambda_r^{\max} \leq \lambda_N^{\max} \quad \text{vagyis} \quad \max\{r \mid r \geq 1, \text{kaszkád}, \gamma_R = 1\} \leq \max\{\lambda \mid \Phi(N) = r > 1, \gamma_R = 1\}.$$

A λ_1^{\max} a maximális érkezési ráta egy izolált kapcsoló esetében úgy, hogy nem blokkolnak a hívások a kapcsolóban ($\gamma_R=1$). Az átlagos útvonalhosszat az alábbiak szerint becsülhetjük:

$$\bar{L} = \sum_{r=1}^{\Phi} p_r \cdot r, \quad \text{ahol } p_r \text{ annak a valószínűsége, hogy a hívás ‘}r\text{’ kapcsolón halad át.}$$

3.2 tézis Megmutattam, hogy a λ_N^{\max} az alábbi analitikus formulával közelíthető:

$$\lambda_N^{\max} \approx \left[1 - \frac{\theta_N^{\max} \cdot (1 + N_r)}{N} + \left| \frac{N_{ISP} - 1}{N_{ISP} + 1} \right| \cdot \frac{\Phi}{(1 - \frac{N_r}{N}) \cdot \bar{L}} \right] \cdot \frac{\lambda_1^{\max}}{1 + \frac{\bar{s}}{2 - p_s}}, \quad (25)$$

ahol, N_{ISP} az internet szolgáltatási szerverek száma (ISP = Internet Service Provider),

$$\bar{s} = \frac{1}{m} \cdot \sum_{i=1}^m s_i \quad \text{az átlagos jelzés-információs kitöltési mutató (bonyolult hívás-profilra } s = 1),$$

m a különböző típusú bonyolult hívás-profilok száma,

p_s annak a valószínűsége, hogy egy adott hívás bonyolult profilú.

Másodszor, egy *inhomogén hálózatot* vizsgáltam meg (ahol nem minden csomópont azonos típusú). A gyakorlatban elég sokszor találkozunk két vagy három különböző típusú ATM kapcsolóval egy szolgáltató hálózatában. Ebben az esetben a (25)-ös egyenlet az alábbi formában általánosítható:

$$\lambda_N^{\max} \approx \left[\max[\min_{i=1..P}(\lambda_1^{\max})^{\text{type } i}, \min_{j=1..R}(\lambda_1^{\max})^{\text{type } j}] \cdot \left(1 - \frac{\theta_N^{\max}}{N/(1 + N_r)} \right) + \left| \frac{N_{ISP} - 1}{N_{ISP} + 1} \right| \cdot \frac{\min_{i=1..P}(\lambda_1^{\max})^{\text{type } i}}{\bar{D}_N} \right] \cdot \frac{1}{1 + \frac{\bar{s}}{2 - p_s}} \quad (26)$$

ahol $(\lambda_1^{\max})^{\text{type } j}$ az ISP szerverre csatlakozó kapcsoló maximális hívás érkezési rátája ($j=\{1, \dots, R\}$).

A továbbiakban megvizsgáltam a T_C és T_{RN} paraméterek változását egy N csomópontos hálózatban, amennyiben az alapbeállítású *SETUP* üzenet helyett (amely csak a kötelező információs elemeket tartalmazza) különböző bonyolultságú profillal rendelkező hívások érkeznek a hálózatba. A komplex hívásprofil (*CCP*) az ‘ s ’ paraméterrel jellemezhető, ahol $0 < s < 1$ (lásd. 1.1.4. altézis).

3.3 tézis Kimutattam, hogy ha a $T_C^{CCP} = (1 + s) \cdot T_C^{default}$ összefüggés fennáll egy kapcsolóra (lásd. 1.1.4 altézis), akkor az átlagos hívás felépítési idő a hálózatban így növekszik:

$$\bar{T}_C^{CCP} = \left(1 + \frac{\bar{s}}{2 - p_s}\right) \cdot \bar{T}_C^{default}, \quad (27)$$

miközben az átlagos hívás elbontási késleltetés (\bar{T}_{RN}) nem változik:

$$\bar{T}_{RN}^{CCP} = \bar{T}_{RN}^{default}. \quad (28)$$

A p_s annak a valószínűsége, hogy a hálózatba érkező hívások bonyolult profillal rendelkeznek. Ha az összes hívás a hálózatban bonyolult profilú, akkor $p_s = 1$, ha viszont az összes hívás egyszerű profilú, akkor $p_s = s = 0$. Ezért a 3.3 tézis különös jelentőséggel bír a hálózat dimenzionálásakor, hiszen megóv a kapcsolók jelzési túlterhelésétől, s aláhúzza a T_C és T_{RN} idők eltérő alakulását.

Jelöljük az ekvivalens kiszolgálási rátát egy csomópontban az alábbiak szerint: $\mu^C = \mu_{eq}^{setup}$ a hívás felépítésre, illetve $\mu^R = \mu_{eq}^{release}$ a hívás elbontásra, a kapcsolót pedig jelölje: $(\mu^C; \mu^R)$.

3.4 tézis Kimutattam, hogy ha egy hálózat valamely kapcsolóját $((\mu_i^C; \mu_i^R), i=1, \dots, N)$ helyettesítjük egy új kapcsolóval, melynek kiszolgálási rátájára igaz az

$$\left(\mu_0^C \leq \frac{1}{\Phi + 1} \cdot \min_{i=1..N} \mu_i^C; \mu_0^R \leq \frac{1}{\Phi + 1} \cdot \min_{i=1..N} \mu_i^R\right)$$

összefüggés, akkor meghatározhatóvá válik ezen csomópont jelzés-terhelése. Rendre helyettesítve a hálózat összes kapcsolóját, meghatározhatóvá válik a hálózat legjobban leterhelt csomópontja (ú.n. “szűk keresztmetszete”).

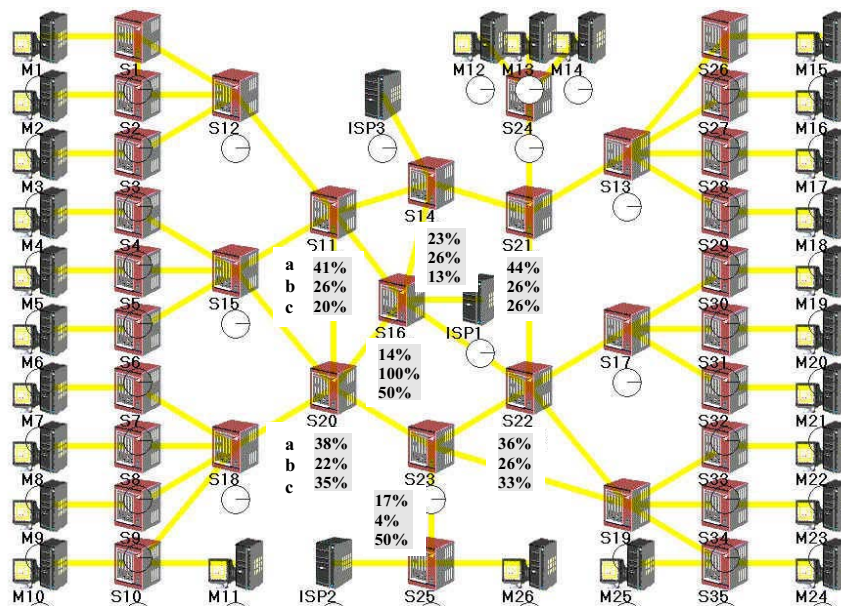
Amennyiben nem lehetséges a gyakorlatban egy hálózat összes pontját egyenként monitorizálni és elemezni, akkor a fent leírt módszer segít a jelzésterhelés meghatározásában. Mindössze a leírt módon helyettesíteni kell a csomópontokat, majd a szimulációt újra meg újra lefuttatni alacsony érkezési intenzitás mellett. Az eredmények értékelésénél látni fogjuk, hogy azon hívások, melyek hívás felépítési ideje $T_C > \frac{1}{\mu_0^C}$, mind-mind áthaladnak a vizsgált ponton. Ezen hívások számát

elosztva az összhívások számával megkapjuk az adott csomópont hívás terhelési valószínűségét. Majd innen kifejezhetjük a csomópont hívás terhelését: $\lambda_i = \lambda_N \cdot p_i$, $i=1, \dots, N$, ahol p_i az a valószínűség, hogy egy adott hívás az i -ik csomóponton halad át. A hálózat “szűk keresztmetszeté”-nek kiválasztásában az alábbi összefüggés segít:

$$\lambda_i^{\max} = \max\{\lambda_i \mid \lambda_i \geq \lambda_j, \forall i, j = 1, \dots, N\}.$$

A 3. tézis validációja

Vizsgálataink során a hívás érkezési intenzitást Poisson típusúnak választottuk, míg a forrás-cél párost egyenletes eloszlásúnak tekintettük LAN Emulációs kísérleteinkben, illetve központilag elhelyezett szervereket feltételeztünk (ISP szolgáltatók) az internet hozzáférése tesztjeinkben (lásd. 6. ábra). Szimulációt futtattunk le az alábbi hálózati szintű hívás intenzitásokra: 1, 5, 33, 100, 200, 400, 500 hívás/sec. Minden esetben 10000 hívást dolgoztunk fel. Eredményeinket $N=\{2, 3, 4\}$ kaszkádolt kapcsoló esetén mérési eredményekkel validáltuk az alábbi típusok esetén: *FORE ASX200BX* és *SEABRIDGE XP140*. Inhomogén hálózatra végzett szimulációnkat a [Mau01] és a [Nov99] cikkekben bemutatott mérési eredményekkel hasonlítottuk össze. Végül eredményeinket összehasonlítottuk a [Gel97] cikk szimulációs eredményeivel is, és megmutattam, hogy az általunk készített modellel pontosabb eredmények érhetőek el.



6. ábra Példa jelzés terhelési valószínűség (p) meghatározására egy 35 csomópontos hálózatban
a) LAN Emuláció esetén; b) Internet hozzáférés 1 ISP-vel; c) Internet hozzáférés 2 ISP-n keresztül

A 3.1 tézis jelentőségét az adja meg, hogy a hálózati topológiától függetlenül a hívás felépítési és – elbontási idők mindig kisebbek, mint az azonos úthosszú kaszkád esetében. A 3.2 tézisben bevezetett (25)-ös és (26)-os egyenletek alsó becslést eredményeznek, igazolásuk azáltal történt, hogy nagyon sok és egymástól eltérő hálózati topológiára, illetve alkalmazásra lefuttattuk a már jól bevált szimulációt. Fenti egyenleteket nemcsak ATM, hanem IP hálózatokban is használhatjuk, például a PPP kapcsolatok felépítése és elbontása által okozott terhelés vizsgálatára. A 3.3 tézisben újra kimutattam a T_C és T_{RN} idők közötti különbséget, ezúttal komplex hívások esetén egy N csomópontos hálózatban. Végül a disszertációm [D-6]-os fejezete egy esettanulmányt is bemutat, ahol a fenti 6. ábrán bemutatott topológiában megvizsgáltuk a szomszédos csomópontok terhelésének változását, amennyiben egy link, illetve egy kapcsoló hirtelen meghibásodott, és emiatt a forgalmat át kellett terelni jelzési üzenetek segítségével a szomszédos útvonalakra.

Egy AAL2-es kapcsoló hívás feldolgozó processzor-architektúrájának optimalizálása UMTS hálózatokban

4. tézis Készítettem egy új modellt AAL2-es jelzési csomópontokra és ezen model teljesítmény-analízisét is elvégeztem FIFO, illetve prioritásos sorok esetén.

Az [AAL197]-ben definiált 2-es típusú ATM adaptációs réteget (AAL2-t) választották átviteli technológiájaként a szabványosítási intézetek az UMTS hálózatok rádiós hozzáférési részének földi összeköttetésére [Ene99]. Ahhoz, hogy az AAL2-t egy olyan hálózatra lehessen illeszteni, ahol a bázisállomások közötti hívásátadási funkció (soft-handoff) egy nagyon lényeges elem, ahhoz egy újabb kapcsolási réteget kellett az ATM réteg fölé beépíteni. Az AAL2-es jelzésrendszer egy új protokoll, amely dinamikusan kezelendő AAL2-es kapcsolatok kiépítésére és elbontására szolgál [AALQ99]. Lehetővé teszi rövid adatsomagok továbbítását az ATM infrastruktúra fölött minimális késleltetés és gazdaságos sávszélesség kihasználtság mellett [Bal97], [Won97]. Ebben a tézisben az AAL2-es jelzés pontok bizonyos tervezési elemeivel foglalkoztam, és új technikákat javasoltam az AAL2-es kapcsolatfelépítési idők minimalizálására. Itt kell megemlítenem, hogy az eredményeim megírása és publikálása időpontjában az AAL2-es kapcsolók még nem voltak a gyakorlatban elterjedve (lásd. [C-7]).

4.1 tézis Kidolgoztam egy új modellt és egy új algoritmust, amellyel egy AAL2-es jelzési csomópont architektúráját optimalizálni lehet. [C-7], [D-7]

A modell architektúrája

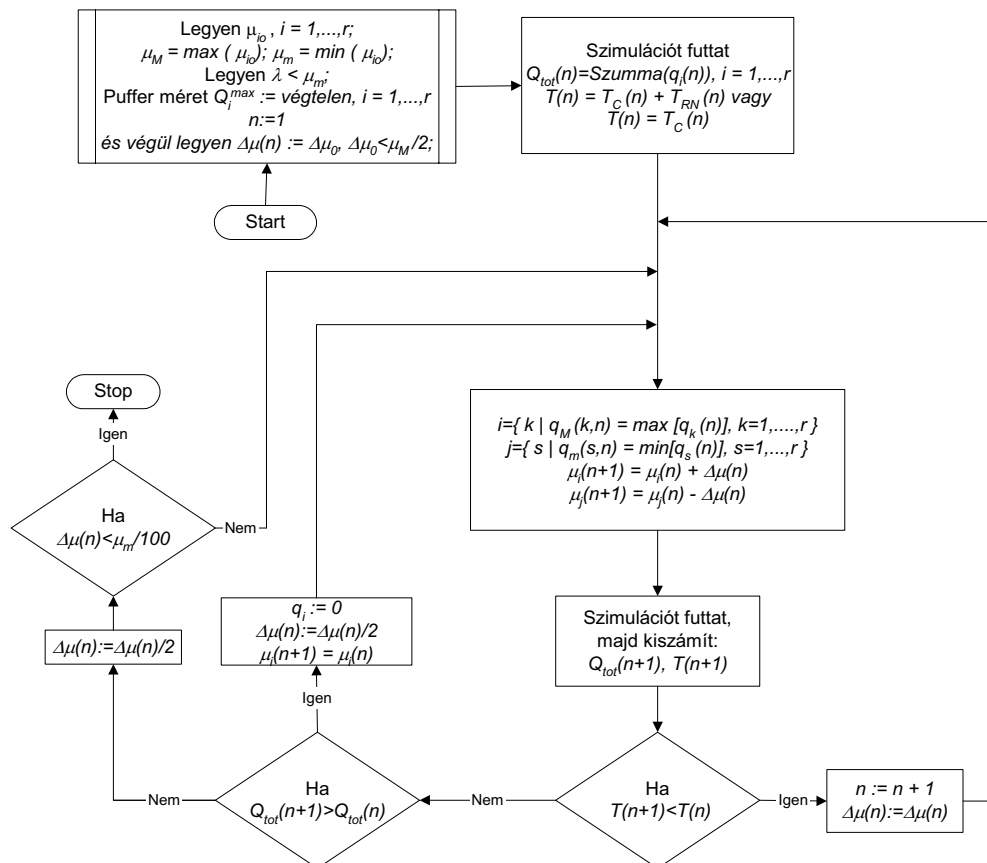
4.1.1. Megmutattam, hogy a 2.1 tézisben leírt folyam-modell bizonyos átalakításokkal alkalmazható egy AAL2-es jelzés pont modellezésére, figyelembe véve az [AALQ99] ajánlást.

A modell átalakítása alatt értendő, hogy elmarad a 3. ábra 6. folyamata (CCP), az üzenetek formátuma rövidebb lesz, valamint az üzenetek megnevezése is különbözik az eddigiektől. További tulajdonságok viszont hasonlóak a 2.1 tézisben leírtakkal.

A modell paramétereinek beállítása

A feladat a következőképpen fogalmazható meg: tekintetbe véve, hogy a jelzési processzor kapacitása véges és állandó, osszuk el a processzor folyamatai között az erőforrásokat úgy, hogy vagy az AAL2-es kapcsolatfelépítési idő (T_C), vagy az 1.1.1 altézisben definiált összesített hívás kezelési idő (T_H) minimális legyen. Az optimalizálási folyamatban egy darab AAL2-es kapcsolót használtunk, amelyre két berendezés kapcsolódott: egy SHO berendezés, amely kezdeményezte a hívásátadáshoz az új kapcsolatokat, illetve egy BS bázisállomás, amely csak végződtette a hívásokat. Az érkezési folyamat Poisson típusú, a kapcsolat tartási idő pedig exponenciális.

4.1.2. Kidolgoztam egy szimuláción alapuló *QUE* nevű iteratív algoritmust, amely a processzor erőforrásait úgy osztja el, hogy ezáltal minimalizálja a hívás felépítési időt és –elbontási késleltetést az UMTS bázisállomások közötti hívásátadási folyamatban.



7. ábra A *QUE* optimalizációs algoritmus blokk-diagramja (FIFO sorok)

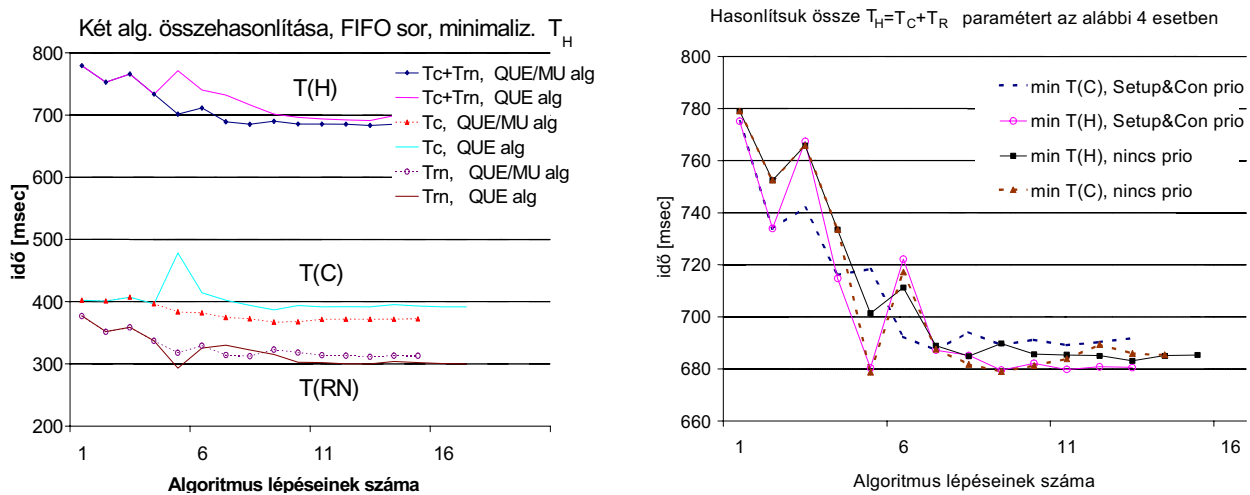
A 7. ábrán látható algoritmus kezdetén beállítjuk a processzor folyamatainak kezdeti kiszolgálási intenzitásait, gondosan megválasztjuk a hívásintenzitást (λ) és a lépésközt ($\Delta\mu$). A puffer méreteket végtelenre állítjuk, elkerülve a puffer-túlsordulás miatt bekövetkező blokkolt hívásokat. A következő fázisban lefuttatjuk a szimulációt, majd kiszámítjuk a $Q_{tot}(n)$ és $T(n)$ értékeket, ahol n az algoritmus sikeres lépéseinek száma. $Q_{tot}(n)$ a folyamatok átlagos sorhosszának összege az n -ik lépésben. A $T(n)$ egyenlő a kapcsolat felépítési idővel vagy az összesített hívás kezelési idővel az n -ik lépésben, attól függően, hogy melyik optimalizációs kritériumot alkalmazzuk. A továbbiakban a processzor erőforrásait osztjuk újra úgy, hogy a leghosszabb sorral rendelkező folyamat kiszolgálási rátáját növeljük $\Delta\mu$ -vel, illetve a legrövidebb sorral rendelkező folyamat kiszolgálási rátáját csökkentjük $\Delta\mu$ -vel. Ezáltal a processzor összkapacitása nem változik. Ezután újra lefuttatjuk a szimulációt, majd kiszámítjuk a $Q_{tot}(n+1)$ és $T(n+1)$ értékeket. Ha ezúttal jobb eredményt értünk el, mint az előző lépésben, azaz $T(n+1) < T(n)$, akkor ezen lépést sikeresnek könyveljük el, n -t megnöveljük eggyel, és az erőforrásokat újraosztjuk a folyamatok között a fent leírt módon. Ha $T(n+1) > T(n)$, akkor a $\Delta\mu$ lépésközt felezzük folyamatosan, mindaddig, amíg az újra meg újra elosztott erőforrások eredményeként $T(n+1) < T(n)$ vagy pedig $\Delta\mu$ lecsökken egy előre meghatározott minimális értékre, ekkor az algoritmust megállítjuk. A $Q_{tot}(n+1)$ és a $Q_{tot}(n)$ összehasonlításából eredő hurokra azért volt szükség, hogy az algoritmust kimozdítsuk az esetleges lokális minimum pontokból, garantálva ezáltal a globális minimum megtalálását.

Ahhoz, hogy tovább csökkenthessük a hívás felépítési időket, bevezettem a prioritásos sorkezelést az AAL2-es jelzési üzenetek egy bizonyos csoportjára. Konkrétan, a hívásfelépítési szakaszban résztvevő üzenetek kapnak magasabb prioritást. A QUE optimalizáló algoritmus viszont ebben az esetben nem mindig találja meg a globális minimumot, ezért egy új kritériumot kellett bevezetnem.

4.1.3. Kimutattam, hogy ha a $Q_{tot}(n) = \sum_{i=1}^r q_i(n)$ összeget helyettesítjük a $Q_{tot}(n) = \sum_{i=1}^r \frac{q_i(n)}{\mu_i(n)}$

összefüggéssel a QUE algoritmusban, akkor az újonnan nyert QUE/MU algoritmussal jobb eredményt érünk el, és ugyanakkor korlátlanul használhatjuk úgy prioritásos, mint FIFO sorok esetén.

Miközben a QUE algoritmus a processzor folyamatainak összesített sorhosszát minimalizálja, addig a QUE/MU algoritmus azt az időt minimalizálja, amelyet egy átlagos üzenet a processzorban eltölt. Ezért FIFO sorokra mindkét algoritmus szinte azonos eredményt szolgáltat, függetlenül attól, hogy valamely köztes lépésben az eredmények eltérőek lehetnek (lásd. 8a. ábra).



8. ábra a) Az algoritmusok lépésenkénti hatása a T_C , T_H és T_{RN} értékekre FIFO sorok esetén;
b) Esettanulmány a T_H minimalizálására (a 3. táblázatban felsorolt 4 esetben)

A globális minimum pont kijelölése attól függ, hogy a hívás felepítési időt vagy az összesített hívás kezelési időt kívánjuk minimalizálni. Elvárásainknak megfelelően a hívás felépítési időre jobb eredményt érünk el a *QUE/MU* algoritmus esetén, ugyanakkor a hívás elbontási időre mindez fordítva igaz, így végül az összesített T_H időre mindkét esetben szoros eredményt érünk el. Kiinduló pontnak mindkét algoritmus esetében az *egyensúlyi helyzetet* választottam, azt a pontot ahol a processzor minden folyamata azonos erőforrással rendelkezik. Legyen ez a továbbiakban a referencia pontunk.

4.2 tézis *Elvégeztem a QUE/MU optimalizáló algoritmus és a prioritásos sorkezelés hatásának teljesítmény analízisét.* [C-7], [D-7]

A *QUE/MU* optimalizáló algoritmus segítségével elvégeztem egy jelzés-processzor erőforrásainak újraelosztását az alábbi négy sajátos esetben (lásd. 3. táblázat). A logikailag lehetséges további két esetre nem végeztem vizsgálatokat, mivel nem találtam arra gyakorlati példát, hogy a hívás elbontási fázisban résztvevő üzenetek előnyben részesülnének a hívás felépítő üzenetekkel szemben.

3. táblázat *Négy logikai eset definíciója (n.v. = nem vizsgáltuk)*

Prioritásos kezelés	Minimalizáljuk az AAL2 kapcsolat	
	T_C idejét	T_H idejét
nincs prioritás (FIFO)	1. eset	2. eset
prioritás a felépítési fázisban	3. eset	4. eset
prioritás az elbontási fázisban	n.v.	n.v.

A 2. tézisben láthattuk, hogy a hálózati késleltetések arányosak egy izolált kapcsolón fellépő késleltetésekkel, ezért a továbbiakban csak egy kapcsoló esetében mutatjuk be az új eredményeinket. A választott kiinduló és egyben referencia pontunk nem a vélhetően legrosszabb eset, hanem az az *egyensúlyi állapot*, ahol a processzor erőforrásai egyenlően oszlanak el a folyamatok között ($i = \{1, \dots, 5\}$).

4.2.1. **Kimutattam, hogy az *ERQ* (\equiv *SETUP*) és *ECF* (\equiv *CONNECT*) üzenetek prioritásos kezelése által a T_C hívás felépítési idő 8-12%-kal csökken (bármely állapotban). Ugyanakkor, a T_{RN} hívás elbontási késleltetés 9-11%-kal növekszik.**

A vizsgálataim során megállapítottam, hogy a prioritásos üzenetkezelés bevezetése által a T_C időt hatékonyabban lehet csökkenteni, mint magával az algoritmussal. A 4.2.1 altézisből viszont az is következik, hogy a T_H idő nem változik a prioritás bevezetésével ($\Delta T_H \approx 1-3\%$, $\forall n=1, \dots, 16$, lásd. 8b. ábra). Figyelembe véve ezen tényt, megállapítottam, hogy *a kapcsolóból kilépő linken a sávszélesség-kihasználtság nem növelhető azáltal, hogy a hívás felépítési üzenetekre prioritásos sorkezelést alkalmazunk, hiszen a T_H összesített kezelési idő továbbra is konstans marad.*

Ha figyelmesen megnézzük a 4 felvázolt esetet, észre vesszük, hogy a *QUE/MU* algoritmus alkalmazása az első két esetben (amikor nincs prioritás) a T_C idő 10%-os csökkenéséhez vezet, ám a 3. esetben a T_C mindössze 6%-kal csökken, míg a 4. esetben még mérsékeltebben, azaz 3%-nyit. Továbbá megállapíthatjuk, hogy az prioritásos kezelés és az algoritmus együttes hatásaként a 3. esetben sikerült a leghatékonyabban csökkenteni a T_C -t, ellenben a legkevésbé a T_H -t.

4.2.2. **Megmutattam, hogy a négy vizsgált esetben a *QUE/MU* algoritmus különböző végpontba vezet, kiindulva ugyanabból az egyensúlyi pontból, de ezen optimum-pontokban a T_H idő legalább 15%-kal csökken az *egyensúlyi állapothoz* képest.**

Az analízis során arra is figyelmesek lehetünk, hogy a T_{RN} hívás elbontási késleltetés legalább 25%-kal csökken bármely esetben (a referencia ponthoz képest), de azt is láthatjuk, hogy míg ez a csökkenés a 3. esetben a legkevesebb (25%), addig a 4. esetben eléri a 32%-ot. Ugyanakkor azt is

észrevehetjük, hogy a T_{RN} idő akkor is csökken, hogy ha az algoritmus kitűzött célja nem a T_{RN} , hanem a T_C idő csökkentése volt. Ez annak köszönhető, hogy a processzor első 3 folyamatán úgy a hívásfelépítő, mint az –elbontó üzenetek áthaladnak.

A 4. tézis validációja

Mindkét algoritmust rendre leteszteltük, kiindulva különböző eredeti állapotból, többek között a szélsőséges értékekből is, és bebizonyosodott, hogy minden esetben mindhárom vizsgált paraméter (T_C , T_{RN} és T_H) konvergált az optimum pontba, mindössze az algoritmus lépés-számban voltak különbségek (10 és 20 között). Az algoritmus átlagosan 13-15 lépésben konvergált, legrosszabb esetben ($EXTREME_x$ szélsőértékekből kiindulva) érte el a 20 lépést. Kipróbáltuk több különböző C processzor-kapacitás értékre is az algoritmust, de az optimum pontban az arányok nem változtak. Megállapítottuk, hogy a QUE algoritmust alapvetően nem használjuk prioritásos sorok esetén (csak $FIFO$ -ra), mert különben nem garantált a globális minimum pont megtalálása. A QUE/MU algoritmus viszont minden esetben a globális minimum pontba konvergál. Sajnálatos módon a szakirodalomban nem találtam hasonló algoritmust, hogy eredményeimet össze tudjam hasonlítani másokéval. A [Wu97]-es cikk bevezetett ugyan egy modellt, amely mindössze a hívás felépítést modellezi, s ezt a folyamatot optimalizálta is a Lagrange-módszer segítségével, de sajnos numerikus eredményeket nem mutatott be.

A 4.2.1 altézis jelentősége abban rejlik, hogy megmutatja a prioritásos üzenetkezelés pozitív hatását a hívás felépítési időre, ami által alkalmassá válik a hívásátadási folyamatok kezelésére, ugyanakkor azt is megmutatja, hogy prioritásos kezeléssel nem lehet sáv szélességet spórolni a hálózati linkeken. Ugyanakkor a 4.2.2 altézis megmutatja, hogy a QUE/MU algoritmus bevezetésével a processzor állapotai optimalizálhatóak oly mértékben, hogy az összesített kezelési idő jelentősen lecsökken, ami által viszont a sáv szélesség-igény csökkenthető a linkeken. Erre több példát is bemutattunk a [C-7]-es cikkben, a 3. és 4. esetet tanulmányozva UTRAN hálózati szintű szimulációval. Következtetésként ajánlani tudom úgy a prioritásos üzenetkezelés, mint a QUE/MU algoritmus használatát UTRAN hálózati tervezésnél.

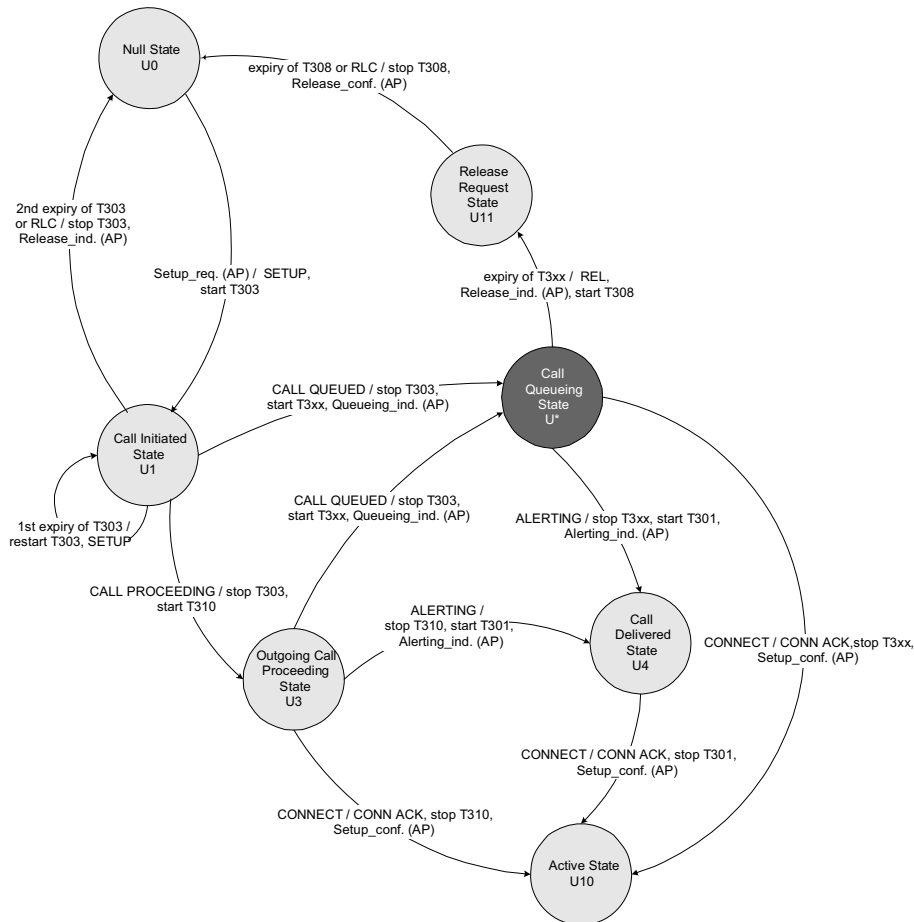
Egy új mechanizmus alkalmazása ATM hálózatokban. Blokkolt szélessávú hívások várakoztatása: BCQ.

5. tézis *Bevezettem egy új mechanizmust (BCQ) az ATM hozzáférési hálózatban és elvégeztem ezen sorbanállási mechanizmus teljesítmény analízisét.*

A BCQ mechanizmus lényege, hogy a hálózatban blokkolt hívásokat nem dobja el, hanem tárolja a híváshoz tartozó jelzési információkat a hozzáférési hálózat első csomópontjában, elkülönítve az élő, illetve az újabb hívásoktól, majd bizonyos idő múlva ezen várakozó hívások automatikusan újra próbálkoznak, ha a hálózatban a szükséges erőforrások felszabadultak. Azt már a [Ber96]-os cikk is kimutatta, hogy a szélessávú hívásoknak nagyobb a blokkolási valószínűsége, mint a keskenysávú hívásoké, ha viszont a blokkolt szélessávú hívásokat elutasítás helyett várakoztatjuk, akkor lecsökkenthető ezen típusú hívások blokkolási valószínűsége, cserébe megnő valamelyest a keskenysávú hívások felépítési ideje és blokkolási valószínűsége [Bla96]. Ezen információk ismeretében kidolgoztam a BCQ mechanizmust és megvizsgáltam annak lehetőségét, hogyan lehet a már szabványosított jelzésrendszerbe ezen új mechanizmust beilleszteni. Továbbá megvizsgáltam a BCQ által okozott többlet jelzésforgalmat és szolgáltatási minőséget. Az elveszett szélessávú hívások egy bizonyos hányada érdekes itt, amelyek a hálózatban nem pedig a végeken blokkolnak. Javaslatom alapján, ebben az esetben nincs szükség a hívó fél közvetlen beavatkozására.

5.1 tézis Megmutattam, hogy a BCQ mechanizmus alkalmazható az érvényben lévő [Q2931], [UNI40] jelzési protokollok megfelelő kiegészítése révén. Ehhez bevezettem egy új állapotot (U^*), egy új üzenetet (CALL QUEUED) és egy új időzítőt (T_{3xx}). [C-2], [C-4], [D-8]

Az említett protokollok kiegészítésének részleteit a [C-2]-ben és a [D-8]-ban ismertettem. A szélessávú BCQ mechanizmus implementálása nem okoz további bonyodalmakat, ugyanis a [Q2931], [UNI40] folyammodellek kiegészítése során hasonló komplexitású elemeket használtam, mint a meglévők. Az állapotdiagram öt eredeti állapotát kellett kiegészítenem a BCQ bevezetése során. Mindezen változtatások és az új állapot a 9.ábrán láthatóak. Ennek megfelelően a szimulációs modellünket is kiegészítettem a BCQ mechanizmussal, ezáltal a hozzáférési pontokban a szélessávú hívásokat opcionálisan várakoztatni lehet. Amíg az első négy téziscsoportban azt feltételeztük, hogy az összes hívás sikeres, addig az 5. tézisben kifejezetten a szélessávú blokkolt hívásokra koncentráltam, és ezek sáv szélesség-igényét elemeztem.



9. ábra A BCQ mechanizmus állapot diagramja az UNI interfészen, a forrás oldalán (egyszerűsített gráf)

5.2 tézis Elvégeztem a blokkolt hívás várakoztatási BCQ mechanizmus teljesítmény analízisét. [C-2], [C-3], [C-4], [D-8]

Ebben a tézisben analitikus megoldást kerestem a blokkolt, várakoztatott, majd megismételt szélessávú hívások felépítési idejére. A hívás elbontási idő, illetve –késleltetés vizsgálata itt nem célom, hiszen ezen paraméterek megegyeznek a nem blokkolt hívások eredményeivel, amelyeket az előző tézisekben már bemutattam.

5.2.1. Kidolgoztam egy analitikus megoldást a blokkolt, majd megismételt hívások felépítési ideje (T_C^{BCQ}) alsó és felső korlátjainak kiszámítására:

$$ET_C^0(i, j) \leq ET_C^{BCQ}(i, j) \leq ET_C^0(i, j) + T_{3xx} \quad (29)$$

ahol: $ET_C^0(i, j)$ egy becslés a keskenysávú hívások felépítési idejére a főútvonal mentén i és j végpontok között. A blokkolt szélessávú hívások új időzítője (T_{3xx}) rövidebb, mint egy előre definiált paraméter, $T_{3xx} < T_{GoS}$, de elég hosszú ahhoz, hogy több próbálkozást is lehetővé tegyen.

Tegyük fel, hogy $E(N)$ a hívásismétlések számának várható értéke, és d_n annak az időnek a várható értéke, amelyet egy megismételt hívás tölt az elérési csomópontban az n -ik próbálkozás után.

$$N = \max \{n : t_1 + t_2 + \dots + t_n < T_{3xx}\},$$

ahol: $t_1 = d_1 + T_C^1(i, j); t_2 = d_2 + T_C^2(i, j); \dots; t_n = d_n + T_C^n(i, j)$, illetve $T_C^k(i, j)$ az az átlagos hívás felépítési idő, amelyet egy keskenysávú hívás tölt az i és j végpontok között a $\pi_k(i, j)$ alternatív útvonalon, $k=1, \dots, n$. Lorden tétele [Lord70] alapján behatároljuk az $E(N)$ alsó és felső korlátait:

$$\frac{T_{3xx}}{E[T_C^0(i, j) + d_0]} - 1 \leq E(N) \leq \frac{T_{3xx}}{E[T_C^0(i, j) + d_0]} + \frac{E[T_C^0(i, j) + d_0]^2}{[E[T_C^0(i, j) + d_0]]^2}$$

ahol $T_C^0(i, j)$ az az átlagos hívás felépítési idő, amelyet egy keskenysávú hívás tölt az i és j végpontok között a $\pi_0(i, j)$ főútvonal mentén, illetve d_0 annak az időnek a várható értéke, amelyet egy megismételt hívás átlagosan várakozik az elérési csomópontban a BCQ folyamat során.

5.2.2. Továbbá feltéve, hogy $N=n$, az alábbi megközelítő eljárást dolgoztam ki a blokkolt, majd megismételt szélessávú hívásfelépítési idő várható értékének meghatározására:

$$ET_C^{BCQ}(i, j) \approx (1 - p_0^{n+1}) \cdot T_C^0(i, j) + \frac{p_0}{1 - p_0} \cdot (1 - p_0^n) \cdot d_0, \quad (30)$$

ahol p_0 a szélessávú hívások blokkolási valószínűsége a $\pi_0(i, j)$ főútvonal mentén i és j végpontok között, $T_C^0(i, j)$ és d_0 ugyanaz, mint az előző összefüggésben. Ezen paramétereket az alábbiak szerint határoztuk meg:

$$T_C^0(i, j) = 2 \cdot \sum_{k \in \pi_0(i, j)} E_{kz}(m) \cdot E_k(T) = \frac{2}{\mu^C} \cdot \sum_{k \in \pi_0(i, j)} E_{kz}(m) \quad (31)$$

Feltételezve, hogy az elveszett hívás egy w_t idő után áll be újra a sor végére az elérési csomópontban, ahol $w_t = E(MHT_t)$ a 't' típusú hívás átlagos tartási ideje, a d_0 így számítható:

$$d_0 = w_t + \frac{1}{\mu^C} \cdot \rho_A \cdot (1 - p_t) \quad (32)$$

ahol ρ_A az elérési csomópont kihasználtsága, μ^C a homogén hálózat egy csomópontjának ekvivalens kiszolgálási intenzitása, és p_t a 't' típusú hívás átlagos blokkolási valószínűsége egy adott linken. Jelölje p_{kz} a Z csomópontból eredő k link hívás blokkolási valószínűségét, ekkor:

$$p_0 = 1 - \prod_{k \in \pi_0(i, j)} (1 - p_{kz}) = 1 - (1 - p_t)^{\bar{L}(i, j)}.$$

A p_{kz} valószínűség meghatározása az összes létező linkre nem egy triviális feladat. Itt azonban feltételezzük, hogy az elérési- és a gerinchálózatban ugyanaz a link-szintű 'p_t' blokkolási valószínűség a 't' típusú hívásokra.

Néhány további lépés után elérkezünk a végső összefüggéshez:

$$ET_C^{BCQ}(i, j) \approx (1 - p_0^{n+1}) \cdot \frac{2}{\mu^C} \cdot (1 - p_t) \cdot \sum_{k \in \pi_0(i, j)} \rho_{kz} + \frac{p_0}{1 - p_0} \cdot (1 - p_0^n) \cdot \left[w_t + \frac{1}{\mu^C} \cdot \rho_A \cdot (1 - p_t) \right]. \quad (33)$$

A disszertációm [D-8]-as fejezetében egy sor numerikus eredményt ismertettem a p_t, w_t, n paraméterek különböző értékeire.

Végül szimulációval megvizsgáltam és meghatároztam azt a kihasználtsági intervallumot, ahol a BCQ mechanizmus előnyösen alkalmazható, majd arra a következtetésre jutottam, hogy:

5.2.3. A BCQ mechanizmus alkalmazása akkor a előnyös ha $0.5 \leq \rho_{kz} \leq 0.67$, miközben a hálózatban a szélessávú hívások aránya meghaladja a 10%-ot. Ha $\rho_{kz} < 0.5$, a mechanizmusra nincs szükség, míg ha $\rho_{kz} > 0.67$, a BCQ nem hatékony ($T_C^{WB} \gg, p_0 \gg$).

Az 5.2 tézis validációja

Analitikus eredményeim igazolásához szimulációs vizsgálatokat végeztem a hívás felépítési idők, p_0 blokkolási valószínűségek és a sorhosszak meghatározására úgy keskenysávú, mint szélessávú hívások esetén. Több különböző hálózati topológiát és hívás-keverési arányt is megvizsgáltam, melynek részleteit a [C-4]-ben és a [D-8]-ban találjuk. Kimutattam, hogy a BCQ használata esetén egy nagyterjedésű hálózatban a 2-10 Mbps-os szélessávú hívások p_0^{WB} blokkolási valószínűsége akár 50%-kal is csökkenhet, természetesen a 0.1 Mbps-os keskenysávú hívások rovására, amelyek blokkolási valószínűsége viszont csak mérsékelten növekszik meg (max. 20%). Ezáltal növelhető a hálózati kihasználtság, amely úgy a végfelhasználók, mint a hálózati operátorok szempontjából előnyös. A [C-4]-ben további eredményeket is bemutatunk, mint például az elérési pontok kihasználtságától, a hívás eloszlástól és az átlagos hívás tartási időtől való függőséget. Továbbá kimutattam, hogy a T_C^{WB} átlagos hívás felépítési idő alig növekszik (<10%) a BCQ bevezetésével, miközben $p_0 \leq 0.05$, függetlenül a w_i hosszától.

5. Az új eredmények alkalmazása

Az 1. téziscsoportban bemutatott mérési eredmények széles körben alkalmazhatóak az ATM hálózati operátorok által a jelzési hálózatuk dimenzionálása során. Az xDSL alkalmazások széleskörű elterjedésével napjainkban (pld. Európában, az USA-ban és Dél-Koreában) nagyon sok szolgáltató esetében aktuálissá vált az ATM jelzésrendszer bevezetése a már meglévő elérési hálózatukba, ahol az eddigi összeköttetések PVC alapúak. Amennyiben a szolgáltató a minőségi garancia biztosítását szem előtt tartja, a leggazdaságosabb megoldás az ATM jelzésrendszer bevezetése, hiszen ez nem jár hardver kiegészítéssel.

Az 1.2.1 altézisben ismertetett “populációs-diagram” elnevezésű módszert a szerző és jelenlegi munkahelyi kollégái (Siemens AG, Kommunikációs hálózatok rendszer-tesztelési részlege) intenzíven használják a mindennapi munkájukban, a VoDSL és az MGCP, H.323, illetve SIP alapú VoIP rendszerek tesztelése során. A “populációs-diagram” használatát Németországban és az USA-ban saját szerzői szabadalom védi (lásd. [T-2]), és elsőként egy kanadai székhelyű neves tesztberendezés gyártó cég (Navtel Communications, a volt GN Nettest) már implementálta saját készülékében 2005-ben, abban az analizátorban (*interWatch 95000*), amellyel az 1. tézis méréseit végeztük. További gyártókkal tárgyalások folynak.

A 2. tézisben ismertetett modellt módosítottam a 4. tézisben úgy, hogy alkalmas legyen a 3G mobil hálózatok UMTS rádiós elérési hálózatának dimenzionálására. Ezen vizsgálataim során kimutattam, hogy az AAL2-es kapcsoló általam optimalizált processzor-architektúrájával a lehető leggyorsabb hívás-felépítési idők érhetőek el, amely az egyik legfontosabb követelmény a bázisállomások közötti hívásátadások esetében. Ezért fenti modell jó kiindulópontként szolgálhat az AAL2-es kapcsolók jelzésprocesszorának implementációjához.

Végül, az 5. tézis aktualitását a digitális műsorszolgáltatók és TV-stúdiók közötti dinamikus sáv-szélesség-kihasználás szükségessége adja, a blokkolt szélessávú hívások hatékony újraindításával.

6. Irodalmi hivatkozások

- [AALI97] ITU-T Recommendation, “*B-ISDN ATM Adaptation Layer Type 2 Specification*”, I.363.2, Geneva, Switzerland, 1997
- [AALQ99] Draft ITU-T Recommendation, “*AAL type 2 Signalling Protocol (Capability Set 1)*”, Q.2630.1, March 1999
- [ATMF00] ATM Forum Technical Committee, “*UNI Signalling Performance Test Suite*”, (ed. J. Orvis, A Francis), *af-test-0158.000*, October 2000
- [Baf93] M. Bafutto, P.J. Kuhn, G. Willmann, “*Modelling and Performance Analysis of Common Channel Signalling Networks*”, Hirzel-Verlag, AEU, Vol.47, No.5/6, 1993, pp. 411-419
- [Ber96] S.A. Berezner, A.E. Krzesinski, “*Call queueing in circuit switched networks*”, *Telecommunication Systems* No.6, 1996, pp. 147-160
- [Bal97] J. H. Baldwin, B. H. Bharucha, B. T. Doshi, S. Dravida, “*AAL-2 – A New ATM Adaptation Layer for Small Packet Encapsulation and Multiplexing*”, Bell Labs Tech Journal, April 1997, pp. 111-131
- [BCMP75] F. Baskett, K.M. Chandy, R.R. Muntz, F.G. Palacios, “*Open, Closed and Mixed Networks of Queues with Different Classes of Customers*”, *Journal of ACM*, 22 (2), 1975, pp. 248-260
- [Bla96] S. Blaabjerg, G. Fodor, A.T. Andersen, “*Reducing Wide Band Blocking by Allowing Wide Band Calls to Queue*”, *COST 244 Technical Report*, TD(1996/14)
- [Ene99] G. Eneroth, G. Fodor, G. Leijonhufvud, A. Racz, I. Szabo, “*Applying ATM/AAL2 as a Switching Technology in Third-Generation Mobile Access Networks*”, *IEEE Communications Magazine*, June 1999, pp. 112-122
- [Far01] S. Farraposo, E. Monteiro, “*Evaluating PNNI Performance*”, *Proceedings of 4th IEEE International Conference on ATM and High Speed Intelligent Internet Symposium, ICATM*, Seoul, South Korea, April 22-25, 2001, pp. 295-299
- [Gel97] E. Gelenbe, S. Kotia, D. Krauss, “*Call Establishment Overload in Large ATM Networks*”, *Proceedings of the ATM'97 Workshop*, Lisbon, Portugal, May 26-28, 1997, pp. 560-569
- [Kaus97] A. Kaushal, S. Shumate, R. Hill, S. Murthy, D. Niehaus, V. Sirkay, B. Edwards, “*Performance Benchmarking of ATM Signaling Software*”, *Proceeding of OPENSIG Workshop*, Columbia University, USA, October 1997, <http://www.itc.ukans.edu/~niehaus/>
- [Lord70] G. Lorden, “*On excess over the boundary*”, *Annals of Mathematical Statistics*, Vol. 41, 1970, pp. 520-527
- [Mau01] M. Maurogiorgis, N. Papadoukakis, E. Sykas, G. Tselikis, “*ATM Signalling Overview and Performance Measurements in a Local Area ATM Network*”, *IEEE Symposium on Computers and Communications, ISCC'01*, Hammamet, Tunisia, July 3-5, 2001, pp.635-640
- [Mer00] A. Mertz, M. Pollakowski, “*xDSL & Access Networks*”, *Grundlagen, Technik und Einsatzaspekte von HDSL, ADSL und VDSL, Kapitel 6*, Prentice Hall, ISBN 3-8272-9593-9, Germany, 2000
- [Nie97] D. Niehaus, A. Battou, A. McFarland, B. Decina, H. Dardy, V. Sirkay, B. Edwards, “*Performance Benchmarking of Signaling in ATM Networks*”, *IEEE Communications Magazine*, Vol. 35 No. 8, August 1997, pp. 134-143
- [Nov99] J. Novak, A. Pouélé “*Interim Report on the Results of the Quantum Test Programme*”, November 1999, <http://www.dante.net/quantum/qtp/QUA-99-070.pdf>
- [Pil99] R.Pillai, K.Su, J.Biswas, C.Tham “*Call Performance Studies on ATM Forum UNI Signalling Implementations*”, *Computer Communications*, Elsevier Science, Vol. 22, Issue 5, April 1999, pp. 463-469
- [PNNI] ATM Forum Technical Committee, “*PNNI Draft Specification*”, Ver.1.0, 94-0471R11, 1994
- [Q2931] ITU-T Recommendation Q.2931, “*B-ISDN. Dig. Subscriber Sign. Syst. No.2 (DSS2). UNI Layer 3 Specification for Basic Call/Connection Control*”, *COM 11-R 78-E*, October 1994
- [UNI40] ATM Forum Technical Committee, “*ATM User-Network Interface (UNI) Signalling Specification*”, Version 4.0, *ATM Forum/95-1434R8*, April 1996
- [Won97] D. Wong, T. J. Lim, “*Soft Handoffs in CDMA Mobile Systems*”, *IEEE Personal Communications*, December, 1997
- [Wu97] C. S. Wu, J. C. Jiau, K.J. Chen, M. Choy, “*Minimizing Call Setup Delay in ATM Networks via Optimal Processing Capacity Allocation*”, *IEEE Comm. Letters*, Vol.2, No.4, April 1998, pp. 110-113

7. Saját publikációk

[J] *Idegen nyelvű folyóirat cikkek*

- [J-4] S. Székely, G. Szűcs, Cs. Simon, I. Moldován, S. Molnár: On the Impact of Link/Node Failures and Network Applications on the Load and Call Processing Times in ATM Networks, *Periodica Polytechnica Electrical Engineering, Journal of the B.U.T.E.*, Vol.46, No.1-2, 2002, pp.99-122, <LH1> 0.5x4=2p
- [J-3] O. Pop, S. Székely, M. Naornita: Performance Evaluation of Forward Error Correction Schemes in ATM Networks, *Buletin Stiintific of T.U. Timisoara*, 42 (56), 1997, (in Romanian), pp. 89-98, 0.3x3=1p
- [J-2] I. Cselényi, P. Haraszi, S. Székely: The First Hungarian ATM Pilot Network, *Selected Papers from the Hungarian Telecommunication Periodical*, 1997, pp. 6-10, <LH1> 0.3x2=0.6p
- [J-1] M. Ottesteanu, S. Székely: User-Network Interfacing Problems in ATM Networks, *Buletin Stiintific of T.U. Timisoara, Special Edition*, 40 (54), 1995, (in English), pp. 153-160, 0.5x3=1.5p

[C] *Idegen nyelvű konferencia és workshop cikkek*

- [C-11] S. Székely, Sz. M. Kis: Benchmarking Symmetrical DSL Modems with Different Protocol Stacks, 10th IEEE Symposium on Computers and Communications, ISCC'05, Murcia, Spain, 27-30 June 2005, will be avail. online in *IEEE DL*, www.ieee.org, <LRE> 0.5x4=2p
- [C-10] S. Székely: Signalling Performance Evaluation of Large ATM Networks Based on Performance Measurements of Isolated Switches, 7th IEEE Symposium on Computers and Communications, ISCC'02, Taormina, Italy, 27-30 July 2002, pp.670-675, avail. online in *IEEE DL*, www.ieee.org, <LRE> 1x4=4p
- [C-9] S. Székely, G. Szűcs, Cs. Simon: Modelling of Call Processing in ATM Switches Based on Performance Measurements, *Proc. of the IEEE International Conference on Telecommunications, ICT'01*, Bucharest, Romania, 4-7 July 2001, pp. 327-335, <LE> 0.3x4=1.3p
- [C-8] Cs. Simon, S. Székely, K. Németh: Point-to-Multipoint ATM Signalling Performance Measurements, *Proc. of the 7th IFIP Workshop on Performance Modelling and Evaluation of ATM/IP Networks, PMEAN'00* Ilkley, U.K., 16-19 July 2000, pp. W17/1-10, <R> 0.3x3=1p
- [C-7] I. Szabó, S. Székely, I. Moldován: Performance Optimisation of AAL2 Signalling for Supporting Soft Handoffs in UMTS Terrestrial Radio Access Networks, *Proc. of the 5th IEEE Symposium on Computers and Communications, ISCC'00*, Juan-les-Pins, France, 4-6 July 2000, pp.46-52, available online in the *IEEE Digital Library*, www.ieee.org <LREH1> 0.3x4=1.3p
- [C-6] S. Székely, G. Szűcs: Performance Measures of Call Establishment in ATM Networks, *Proc. of the ATMTU'99 Int. Symposium*, Kosice, Slovakia, 17-19 February 1999, pp.116-121, <LE> 0.5x3=1.5p
- [C-5] S. Székely, Cs. Simon, G. Szűcs: Performance Testing on Switched Virtual Connections in ATM Networks, *Proc. of the 6th IFIP Workshop on Performance Modelling and Evaluation of ATM Networks, PMEAN'98* Ilkley, U.K., 20-22 July 1998, pp. 102/1-10, <RH1> 0.3x3=1p
- [C-4] S. Székely, I. Moldován, Cs. Simon: Overload Generated by Signalling Message Flows in ATM Networks, *IFIP TC6 WG6.3 Conference "Performance of Information and Communications Systems"*, PICS'98, Chapman&Hall Publisher (eds. A. Nilsson, U. Körner), Lund, Sweden, 25-28 May 1998, pp. 51-64, available at *DBLP Uni-Trier*, <LRH1> 0.3x4=1.3p
- [C-3] S. Székely: On Bandwidth Allocation Policies in ATM Network using Call Queueing, *Proc. of the 5th IFIP Workshop on Performance Modelling and Evaluation of ATM Networks, PMEAN'97* Ilkley, U.K., 21-23 July 1997, pp. 46/1-10, <R> 1x3=3p
- [C-2] S. Székely, G. Fodor, S. Blaabjerg: Call Queueing: The Design and Performance Analysis of a New ATM Signalling Functionality, *Proc. of the IEEE Workshop, ConTEL B&MW*, Zagreb, Croatia, 11-12 November 1996, pp.99-113, <LEH1> 0.3x3=1p
- [C-1] A. Faragó, S. Blaabjerg, W. Holender, B. Stavenow, T. Henk, L. Ast, S. Székely: Enhancing ATM Network Performance by Optimizing the Virtual Network Configuration, *International Conference on Data Communications and their Performance*, Chapman&Hall Publisher (eds. R. Onvural, H. Perros), Istanbul, Turkey, 23-26 October 1995, pp. 401-414, available at *DBLP Uni-Trier*, <LREH1> 0.14x4=0.5p

[T] *Idegen nyelvű szabadalmak, műszaki jelentések*

- [T-2] S. Székely: Schaltungsanordnung und Verfahren zur Fehleranalyse (*Eine neue Methode für System*

Tester zur Fehlersuche: Repräsentierung von Signalisierungsnachrichten in einem Plot-diagram), Phase 1: Erfindungsmeldung – Siemens AG, 2003-E18241DE; Phase 2: 10.2004.002.452.9, Germany, patent registered, www.dpma.de; Phase 3: 2005-P00559US Nachmeldung, USA, patent pending, 1x2=2p
[T-1] S. Székely: VINCE, Technical Report, T.U. Karlsruhe, Institute of Telematics, Germany, pp.1-37, March, 1996, (<http://www.telematik.informatik.uni-karlsruhe.de/forschung/atm-info/vince/>), <RES1>

[H] *Magyar nyelvű folyóirat cikkek*

[H-4] Székely S., Kis Sz. M.: Adatátviteli teljesítményvizsgálat szimmetrikus DSL berendezéseken, *Híradástechnika*, Vol. LIX, szeptember 2004, 48-57 oldal, 0.5x1=0.5p

[H-3] Szűcs G., Székely S.: Hívásszintű teljesítmény-mérések ATM hálózatokban, *Magyar Távközlés*, 99/5, május 1999, 3-7 oldal, 0.5x1=0.5p

[H-2] Székely S., Arató A.: Videókonferencia alkalmazások ATM hálózaton, *Magyar Távközlés*, 96/6, június 1996, 6-10 oldal, <H1> 0.5x1=0.5p

[H-1] Székely S., Szarkowicz K.: Bepillantás az ATM jelzésekbe, *Magyar Távközlés*, 95/5, május 1995, 27-30 oldal, 0.5x1=0.5p

[R] *Akadémiái munkák, melyek leihivatkozták saját cikkeimet*

[R-6] G. Fehér: Resource Control in IP Networks, Ph.D. dissertation, Budapest University of Technology and Economics, April 2004, *igény szerint elérhető*, hivatkozott a [J-4]-re

[R-5] S. Waller: New AAL2 signalling protocol to support the UMTS Terrestrial Radio Access Network, M.Sc. thesis, University of Cape Town, 2003, <http://crg.ee.uct.ac.za/progress.html>, hivatkozott a [C-7]-re

[R-4] A. Schwarz: Modellierung und Bewertung von Verfahren zur Last- und Leistungsregelung in Steuereinheiten von B-ISDN/ATM Vermittlungssystemen, Ph.D. dissertation, University of Stuttgart, LDB #33612, 2000, *igény szerint elérhető*, hivatkozott a [C-4]-re és [C-5]-re

[R-3] I. Cselényi, R. Szabó, I. Szabó, A.L.-Henner, N. Björkman: Experimental Platform for Telecommunication Resource Management, *Computer Communications*, Vol. 21, pp.1624-1640, 1998, <http://qosip.ttt.bme.hu/papers/local/cselenyi98compcomm.pdf>, hivatkozott a [C-2]-re

[R-2] B. Kaan: A Virtual Path Routing Algorithm for ATM Networks based on the Equivalent Bandwidth Concept, M.Sc. thesis, Bogaziti University, 1998, <http://mercan.cmpe.boun.edu.tr/~kaanbur/thesis.pdf>, hivatkozott a [C-1]-re

[R-1] G. Seres, F. Baumann, G. Gordos, T. Henk: ATM alkalmazása a lokális hálózatban. A BME-TTT tanszék helyi ATM hálózata, *Networkshop'97*, Keszthely, Hungary, May 27-29, 1997, <http://www.iif.hu/rendezvenyek/networkshop/97/tartalom/NWS/1/7/index.htm>, hivatkozott [J-2]-re, [H-2]-re

[D-n] *Disszertáció fejezetei* n = 1-től 10-ig

Min. szükséges pontszám: 12

Össz-pontszám: 27

8. Köszönetnyilvánítás

Kiemelt köszönetet szeretnék az alábbi személyeknek és intézményeknek mondani:

- legelsősorban Henk T.-nak, a HSN Laboratórium vezetőjének, 1995 óta tartó szüntelen biztatásáért;
- konzulenseimnek, Molnár S.-nak és Frajka B.-nak folytonos segítségükért és kritikus megjegyzéseikért;
- H. Perros-nak, A. Nilsson-nak, P.J. Kühn-nak, J. Eberspächer-nak, G. Krüger-nak, G. Carle-nak, Sztrik J.-nak, Maricza I.-nak, Kerekes I.-nak, Harangozó J.-nek és az ismeretlenségbe burkolózó névtelen cikk-bírálóimnak az értékes kommentárokért és az útbaigazító bírálatokért;
- kollégáimnak, Moldován I.-nak az ACCEPT nevű szimulátor elkészítéséért (lásd. 2., 3., 4., 5. tézis);
- kollégáimnak, Simon Cs.-nak, Szűcs G.-nak, Németh K.-nak, Fábíán F.-nek, valamint diákjaiknak a felmérhetetlen és hónapokig tartó segítségükért a tesztelések és a cikkírások során (lásd. 1. tézis);
- kollégáimnak, Antal Cs.-nak, Cselényi I.-nak, Szarkowicz K.-nak, Szabó I.-nak és Fodor G.-nak a gyümölcsöző szakmai beszélgetésekért és közös cikkekért (lásd. 4., 5. tézis);
- publikációim további társszerzőinek;
- a Magyar Oktatási Minisztériumnak, s az Erdélyi Múzeum Egyesületnek a 4 évig folyósított ösztöndíjért;
- végül, de nem utolsósorban Gordos G.-nak, a BME-TTT korábbi vezetőjének, Boda M.-nak, az Ericsson Magyarország K&F csoport korábbi vezetőjének és családomnak folytonos támogatásukért.