

# Rétegek együttes optimalizálása UMTS-alapú videóátvitelben

FÜLÖP ATTILA, ZSIROS ATTILA, JENEY GÁBOR

Budapesti Műszaki és Gazdaságtudományi Egyetem, Híradástechnikai Tanszék  
{fulopa, zsirosa, jeneyg}@hit.bme.hu

**Kulcsszavak:** együttes forrás és csatornakódolás, multimédia átvitel, IPv6, UMTS rádiós elérési hálózat, erőforrás vezérlés

*Ebben a dokumentumban egy olyan szoftverplatform kerül ismertetésre, amely demonstrációs és teljesítményelemzési céllal lett kifejlesztve. A platform alkalmazási és egyéb (pl. fizikai) rétegek együttes optimalizálásán alapuló vezeték nélküli videó átvitelt szimulál. A fejlesztés az IST-PHOENIX projekt keretein belül valósult meg és a projekt által kifejlesztett rendszer optimalizálási modellen nyugszik. A kifejlesztett platform egy lényeges elemét, a vezeték nélküli hálózati szegmensét azonban egy részletes, szabványos UTRAN hálózati szimulációra cseréltük. A cikk tartalmazza a projekt szimulációs láncának rövid ismertetését, illetve az UTRAN rendszerrészek ismertetését, a két szegmens összekapcsolását. Kitérünk az UTRAN optimalizálásban való szerepére, a konfigurációjára, vezérlésére, majd néhány szimulációs eredmény kerül bemutatásra.*

## 1. Bevezetés

A vezeték nélküli telekommunikációs hálózatok fejlődése egy egyre integráltabb és globálisabb rendszer irányába mutat. A hagyományos felfogás (ISO/OSI) szerint rétegekre bontjuk a kommunikációban résztvevő álmások funkcióit, hogy azokat könnyen megfogható módon, egyénileg alakíthassuk tökéletessé. Manapság azonban a felhasználói és ipari igények kielégítése csak a különféle technológiák konvergenciájával válik lehetővé. Ez egyben a különálló rétegek összemosását, a funkciók átlapolódását, keveredését jelenti. Ebben a cikkben egy olyan projekt és szoftveres demonstrációs platform bemutatására kerül sor, amelynek célja az utóbbi felfogás támogatása. A szoftver lehetővé teszi a rádiós videóátvitel alkalmazási és egyéb rétegek közötti együttes optimalizálásának teljesítményelemzését. A platform az IST PHOENIX projekt keretein belül készült (<http://www.ist-phoenix.org>) és egy, a projekt által fejlesztett modellen alapul.

A rendszer felépítése követi a hagyományos ISO/OSI modellt, ugyanakkor célja egy olyan stratégia megvalósítása, ahol a forráskódolás, csatornakódolás és modulációs paraméterek együttesen kerülnek meghatározásra egy központi vezérlő intelligencia által. Ezt nevezzük közös forrás- és csatornakódolás/dekódolásnak (Joint Source and Channel Coding/Decoding – JSCC/D). A tradicionális esetben a forráskódolás (tömörítés) és a csatornakódolás (hibajavító kódolás) egymástól elválasztva kerül megvalósításra Shannon jól ismert szeparációs tétele alapján [1].

Ez az elv komplex, ám jól teljesítő rendszereket eredményez, de nem különösebben hatékony az olyan mai népszerű alkalmazások esetében, mint például az audió-videó jelfolyam átvitel [2]. A mai modern alkalmazásoknak gyakran olyan, a szeparációs tétel ideális feltevéseit nem kielégítő jellemzőik vannak, amiket réteges megközelítésben nem lehet kielégíteni. Ilyenek többek

közt a valós idejű átviteli követelmény, vagy a változó hibaérzékenységgű forrásadatok változó hibavédelme.

A cikkben közölt architektúra különböző hozzáférési technikákkal alkalmazható, például OFDM vagy WCDMA. Mind a H.264/AVC [3], mind az MPEG-4 videó kódolás támogatott az alkalmazási rétegben. Különböző szállítási rétegbeli protokollok (UDP, UDP-Lite, DCCP) használhatóak az átvitelhez. A projekt által fejlesztett modell és szimulációs eszköz lehetőséget biztosít számunkra, hogy egy valós rendszert, az UMTS hálózatát használva is kipróbáljuk az együttes optimalizálás (JSCC/D) elvét. A cikkben az UTRAN WCDMA rendszerre ültetett rendszermodell bemutatására koncentrálunk, az általunk fejlesztett, részletes, szabványhoz igazodó UMTS szimuláción keresztül [4]. A modell részeinek funkcionális bemutatása után az optimalizálási lehetőségek elemzése következik, részletesebben kitérve az UTRAN által támogatott átviteli konfigurációkra. Különböző futtatási beállítások mellett mutatjuk be az optimalizálás hatását az átvitt videó folyam minőségére.

A 2. fejezetben a szimulációs rendszer felépítését ismerhetjük meg, ezt követően pedig a jelzési mechanizmusokat részletezzük. A protokoll-hierarchiát két részre, alkalmazási és adattovábbítási modulokra bontva a 4. és 5. fejezet tartalmazza az egyes modulok rövid, az optimalizálás szempontjából releváns működésének ismertetését.

## 2. A szimulációs rendszer architektúrája

Az 1. ábrán látható a rendszermodell UTRAN WCDMA hálózatra ültetve. Az ábra egy videófolyam átvitelét mutatja be. Az egymáshoz kapcsolódó elemek vezetékes (IPv6), és vezeték nélküli (jelen esetben UMTS) hálózatokat reprezentálnak, a vezetékes közeg IP hálózat felhőben szerepel, míg a vezeték nélküli közeg a csatornablokk. A rendszer adattovábbítást végző blokkjai

ISO/OSI rétegmodell szerint rendeződnek. A rendszer JSCC/D vezérlőinformációkat használ, amik például az UEP (Unequal Error Protection) védelmet, vagy a JSCC adaptált csatornakódolást vezérlik. Külön vezérlőréteg van definiálva az alkalmazási és a fizikai rétegekhez, ez utóbbi az UTRAN-ban használt rádiós erőforrásvezérlési réteg (RRC). Az alkalmazási rétegbeli visszacsatolt vezérlő (JSCC/D) a forráskódolóhoz, az alkalmazási műveletekhez (titkosítás, UEP) valamint a streaming, transzport és IPv6 csomagképző protollokhoz kapcsolódnak. Ezen utolsó három elem biztosítja egy valós idejű multimédia folyamtnál megkövetelhető QoS-t, sorrendhelyes továbbítást stb. Ezen elemek funkcióját a 4/C szakasz ismerteti bővebben. Az egyes elemek információt nyújtanak a felhasználói igényekről, és a hálózat, illetve a csatorna állapotáról. Az információ a vezérlőelemekhez továbbítódik, amely ezek alapján döntéseket tud hozni. Ilyen módon képes az adaptációt vezérelni, megadja például a videó kódoló tömörítési rátáját, vagy a csatornakódolás szükséges védelmi szintjét.

Az ábrán a folytonos vonal jelöli a hasznos adat, azaz a payload (videó) és a hozzá kapcsolódó információk áramlását. A videóadat a forráskódolás és egyéb alkalmazási műveletek (pl. UEP) után streaming és transzport protollok alkalmazásával egy IPv6 hálózaton halad keresztül. A hasznos adathoz, az optimalizáláshoz elengedhetetlen információk (SSI, SRI) és protokoll felek kapcsolódnak. A videóadathoz nem szinkronizált információk külön jelfolyamban továbbítottak (lásd szaggatott vonal). A visszacsatolt vezérlő megfelelő műkö-

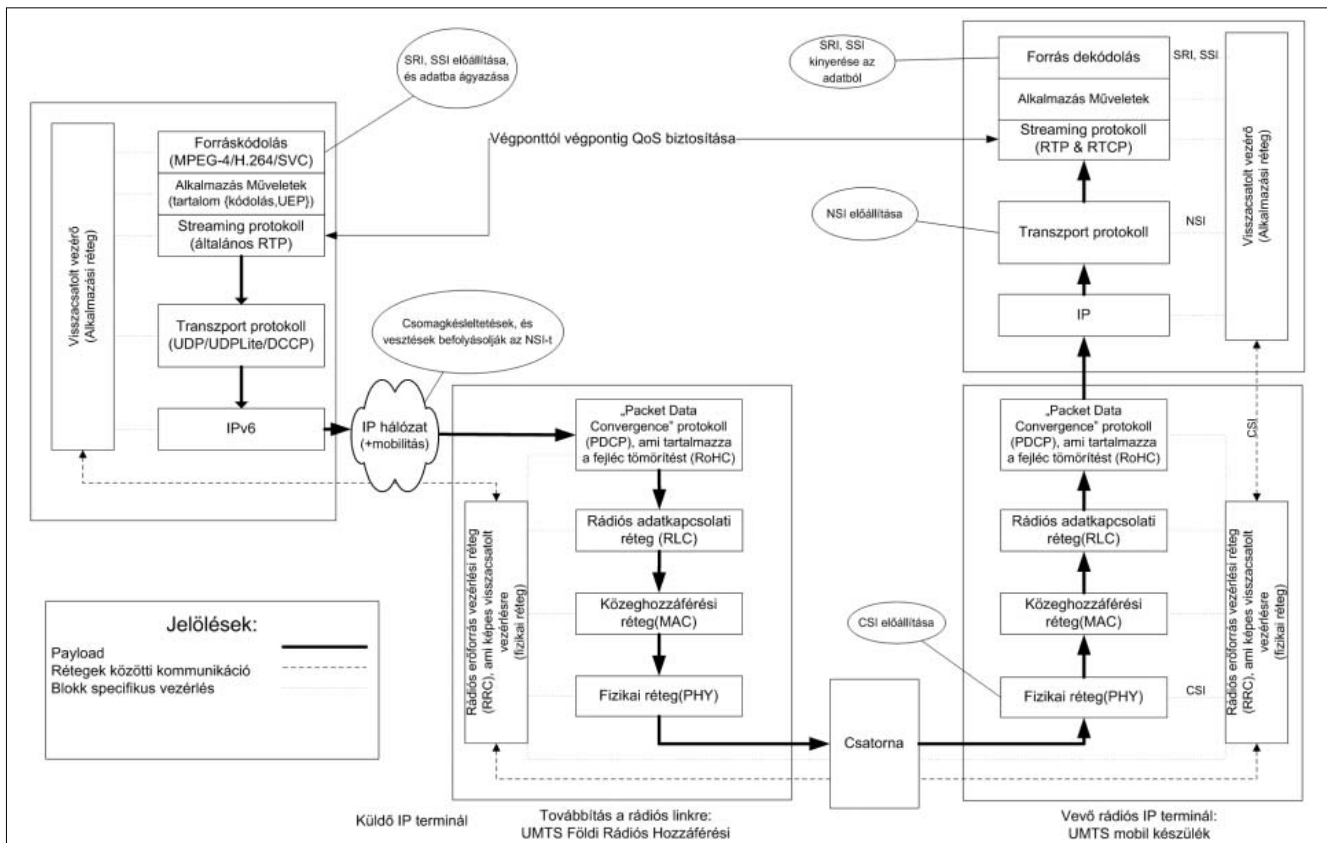
déséhez szükséges volt kialakítani az egyes blokkokkal való interfészeket, melyeket az ábra pontozott vonallal jelöl. Ugyancsak így jelöltük az általunk használt UTRAN modulban a rádiós erőforrás vezérlési réteg és az egyes jelfeldolgozásban résztvevő rétegek (PDCP, RLC, MAC, PHY) közötti interfészeket. Ezek a szolgáltatás hozzáférési pontok (Service Access Points) a szabványokban [8] definiáltak, azonban bővítenünk kellett funkcionalitásukat, hogy lehetővé tegyük az együttes optimalizálást.

### 3. Együttes optimalizálás vezérlése

Vezetéknélküli videó átviteli rendszerekben alkalmazott rétegek közötti optimalizálásra több megoldás is létezik. Ezek közül egy a közeghozzáférési rétegbeli (MAC) automatikus ismétléskérés (ARQ), és fizikai rétegbeli előremutató hibajavító kódolás (FEC) adaptív kiválasztására alapuló módszer [5]. Ezzel szemben a Phoenix projektben követett megoldás (JSCC/D) a teljes protokollvermen keresztül alakít ki kommunikációt, mind az alkalmazási, mind a fizikai és közeghozzáférési rétegek viselkedését próbálja összehangolni a tényleges átviteli körülményektől függően.

Az alábbi releváns információk cseréjére kerül sor [6] az optimalizálás során: SSI (source significant information), azaz a forrás csatornahibákra mutatott érzékenysége; SRI (source a-priori information), azaz a forrás statisztikai információja az egyes adatelemek gyakorisá-

1. ábra Teljes rendszermodell és szimulációs összeállítás



gáról; valamint a DRI (decision reliability information), a csatornadekódoló megbízhatósági információja (ha szoft értékeket tartalmazó kimenete van); a SAI (source a-posteriori information), a forráskódolás statisztikai kimenete és végül a visszacsatolt információk, a CSI (channel state information), például jel-zaj arány; NSI (network state information), ami késleltetési, jitter és csomagvesztési értékeket tartalmaz többek között és a videó minőség.

Az egyes vezérlő információk jelentésbeli és funkcionális jellegük miatt eltérnek frissítési gyakoriságukban, generálási módjukban, adatmennyiségükben és bitsebességükben, így különböző módon kell azokat kezelni adattovábbítás szempontjából. Az SSI-t a felhasználói adattal kell szinkronizálni, mivel az adatforrás csatornahibákra mutatott érzékenységét írja le, amit az UEP modulnak kell kezelni. Ezzel szemben például az NSI információ a hálózat állapotát írja le és visszacsatolás útján jut el az együttes, rétegek közötti vezérlőbe (JSCC/D), azaz nem kell szinkronizálni az adatcsomagokkal.

A szinkronizálásra megjelölt információkat a felhasználói adatfolyamban továbbítjuk, melyre megoldás lehet egy extra RTP fejléc (2. ábra) vagy az úgynevezett IPv6 kiterjesztett fejléc (extension header). A nem szinkronizált visszacsatolt adatok továbbítására pl. ICMPv6 (Internet Control Message Protocol v.6) használható.

Az ábrán a hasznos videó-adat (Payload) és az egyes protokoll fejlécek (header) mellett az extra információ (extra information) kap helyet, mint specifikus RTP fejléc, tartalmazva a titkosítási (CIPHER), SRI, SSI, a ritkítás paraméterét UEP modulban (PUNC), illetve a szoft információhoz köthető adatokat. Ez utóbbi feltételezi, hogy a csatornadekódoló kimenetén 4 biten kvantált szoft információt is rendelkezésre áll. A szoft információt a rendszer a rádiós csatorna kimenetétől a video dekódolóig továbbítja. A szoft információ átvitelre az UTRAN szimuláció nem lett felkészítve, így ez a lehetőség jelenleg nem áll rendelkezésre.

## 4. Az alkalmazási rész

Ebben a fejezetben kerülnek részletezésre az együttes rétegek közötti vezérlő (JSCC/D) által közvetlenül vezérelt egységek, melyek klasszikus értelemben a szállítási réteg és az alkalmazási réteg közötti funkciók szerepét töltik be. A JSCC/D vezérlő az IP hálózat után a vezeték nélküli hozzáférési hálózatot a fizikai rétegbeli alegység-

vel tudja befolyásolni (lásd 1. ábra). Ezekre a modulokra koncentrálunk az ezt követő fejezetben.

### A. Forráskódoló

Az 1. ábrán látható forráskódoló és dekódoló modulok legfelül helyezkednek el. A forráskódolásnak gyakorlatilag a használt videókódek felel meg, feladata a nyers videofolyam hatékony tömörítése. A vevőoldalon a videodekódolónak a kódolt, ám hibás adatfolyamot a lehető legkisebb észlelhető torzítással kell visszaadnia. A szimuláció fejlesztése során 3 kódoló került vizsgálatra: az MPEG-4, H.264/AVC és H.264/SVC kodekek.

### B. UEP, tikosítás

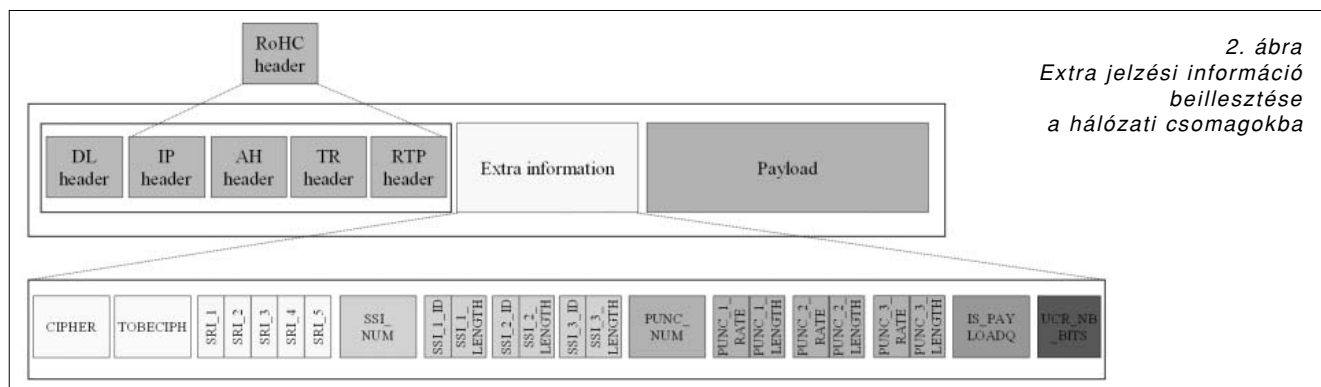
Egy tömörített videofolyam elemei különböző érzékenységet mutatnak a hibák (bithibák és csomaghibák) iránt. Ennek megfelelően különböző védelmet kell biztosítani a különböző érzékenységi osztályoknak. A feladatot az eltérő hibavédelem (Unequal Error Protection) blokk oldja meg. Az adatok védelme az UEP modulban ritkított (punctured) konvolúciós kódolóval van megvalósítva.

### C. Streaming modul

A streaming modul feladata adóoldalon az adatfolyam IP-s csomagokba ültetése, a vevőoldalon pedig a fogadott IP-s csomagok alapján az adatfolyam visszaállítása a felsőbb rétegek részére. Az IP-s csomag azonban nem csak az IP protokollt rejti: a szállítási rétegbeli funkciók kívül ide tartoznak még az RTP és az RTCP is.

A PHOENIX szimulációs láncban mind az RTP, mind az RTCP protokollok implementálva vannak. Ezek a protokollok végponttól végpontig terjedő QoS-t biztosítanak a valós idejű multimédia folyamának időbéllyeggel, és sorszámmal címkézik fel a multimédia csomagokat.

A szimulációs láncban UDP, UDP-Lite és DCCP transzport protokollok vannak implementálva. Az UDP protokoll összeköttetés-mentes, best-effort szolgáltatást nyújt, amelynél nincsen sorszámozás, és duplikátumok is előfordulhatnak. Az UDP-Lite protokoll az UDP protokoll továbbfejlesztése, amely annyiban változott az eredeti UDP-hez képest, hogy egy részleges ellenőrző összeget (CRC) vezettek be. A részleges ellenőrző összeg használatával a teljes csomag helyett csak a fejrész és a payload egy részére képződik ellenőrző összeg. Amennyiben a csomagnak abban a részében ke-



lethezik hiba, amelyet az ellenőrző összeg lefed, a csomag eldobásra kerül.

Ha az UDP ellenőrző összege az egész csomagot lefedi, az UDP-Lite működése megegyezik a hagyományos UDP működésével. Ha viszont csak a fejrészt védjük ellenőrző összeggel, akkor jelentősen javul a hatások. Ezzel az eljárással ugyanis körülbelül 40%-kal csökken az eldobások száma. Egy másik alkalmazott szállítási protokoll a DCCP, ami egy nem megbízható, ütközésvezérelt adatfolyam-szolgáltatást nyújt visszajelzéssel a helyesen megérkezett adatokról. A megvalósított DCCP protokoll nem tartalmaz újraküldési lehetőségeket a datagrammokra és az UDP-Lite-hoz hasonlóan tartalmaz egy részleges ellenőrző összeget. A DCCP ellenőrző összeg minden esetben lefedi az egész fejléct, és a payload  $n \cdot 4$  byte-os részét. Megjegyzendő, hogy a IPv4/IPv6 csomaggenerálás megköveteli, hogy a szállítási rétegbeli protokollok ellenőrzőösszeget alkalmazzanak, habár ez lehet részleges, mint ahogy az UDP-Lite és DCCP esetében említettük.

A szállítási protokoll fejrészt tartalmazó csomagok végül IP-s csomagba lesznek ültetve. A szimulációs lánc kizárólag az IPv6, azaz 6-os verziójú Internet Protokollt támogatja, mivel ez a jövő Internetének protokollja. Az Európai Bizottság minden általa támogatott információtechnológiai projektben az IP 6-os verzióját követeli meg.

A mai hatékony audió/video dekóderek sérült, hibás csomagok feldolgozására is képesek, növelve a képminőséget. Ennek következtében kihasználható a fentebb említett protokollok által nyújtott részleges hibaellenőrző összeg használata, aminek következtében a csomagvesztés csökken és a videó dekódoló által feldolgozott csomagok száma nő.

A RTP/RTCP protokollok jól használhatóak az UDP/UDP-Lite protokollokkal, viszont DCCP használata esetén nélkülözhetők, mivel a DCCP ellátja azokat a szük-

séges funkciókat, amik az RTP/RTCP-t szükségessé tennék. A szimulációs lánc jelenlegi verziója azonban gyakorlati okokból mindenképpen alkalmazza az RTP/RTCP protokollokat.

## 5. Az adattovábbítási rész

### A. IPv6 hálózat

A szimulációban modellezett IP hálózat egy IP routerek ismeretlen halmazának tekinthető. A virtuális hálózat átviteli kapacitása, és buffer mérete szoftveresen tetszőleges értékre beállítható. Minden router okozhat késleltetést, illetve csomagvesztést. A végponttól végpontig terjedő késleltetés Gamma-eloszlású, míg a csomagvesztés egyenletes eloszlású minden csomópontban. A routerek száma, és a hozzájuk tartozó paraméterek szintén beállíthatóak. Ilyen módon minden hálózat modellezhetővé válik. A vezetékes közeg megbízhatósága magas. Emiatt az általunk szimulált 20 másodperces átvitel ideje alatt a csomagvesztés valószínűsége elhanyagolhatóan kicsi. Az IPv6 hálózat jellemzően a csomagok késleltetését változtatja meg, aminek következtében azok sorrendje változhat, illetve borsztössé válhat. Szimulációink során az UTRAN vezetéknélküli hálózati szegmensre koncentráltunk, így az IP hálózat torlódásból adódó csomagvesztését minimálisra (0-ra) állítottuk. Ezáltal a 20 másodperces videó szekvenciák ideje alatt nem volt az IPv6 hálózatban csomagvesztés.

### B. UTRAN

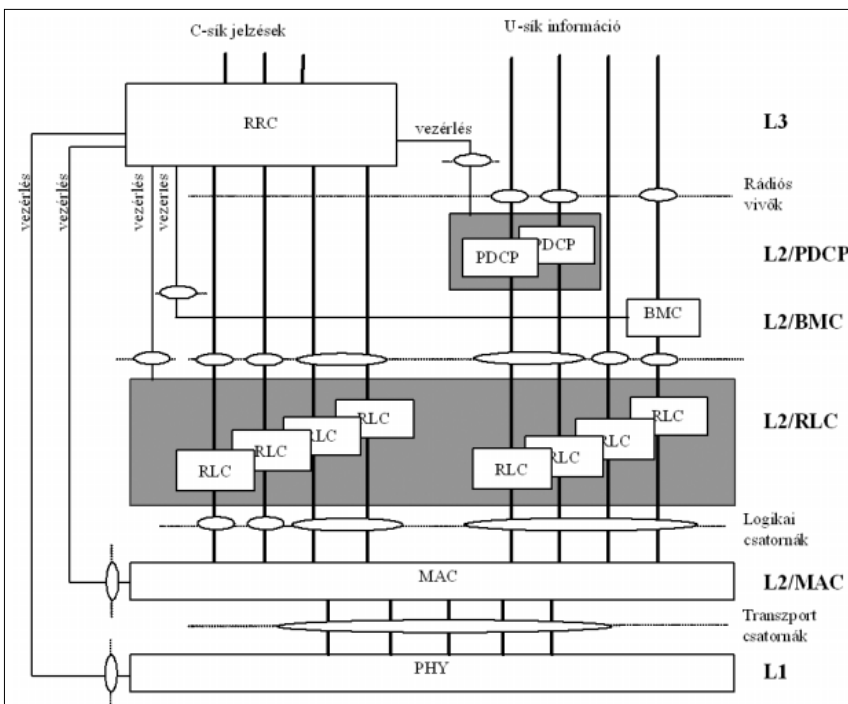
#### 1) Felépítés és funkcionalitás

Az általunk implementált UTRAN modulok tervezési szempontjai a szabványosság, a teljesség és részletesség, illetve a hatékonyság voltak. A szabványosságot a [7-10] ETSI technikai specifikációk követése biztosította, míg a funkcionalitás teljességét egy korlátozott működési mód teljes körű implementálásával valósítottuk meg. Az UMTS-ből csak a dedikált adatátvitelt, illetve a hozzá kapcsolódó járulékos kontroll információk átvitelét implementáltuk. Célunk szempontjából a járulékos funkciók, mint a hálózati hívás felépítés nem fontosak. Vizsgálataink során feltételeztük a kapcsolat kiépített voltát, és csak az adat továbbítására koncentráltunk.

A 3. ábrán látható a UTRAN protokoll architektúrája, melyben az RRC réteg került módosításra a szabványhoz képest, hogy képes legyen fogadni a JSCC/D vezérlő utasításait és a PDCP, RLC, MAC, illetve fizikai rétegeket annak megfelelően konfigurálni.

A 3. ábrán látható a UTRAN protokoll architektúrája, melyben az RRC réteg került módosításra a szabványhoz képest, hogy képes legyen fogadni a JSCC/D vezérlő utasításait és a PDCP, RLC, MAC, illetve fizikai rétegeket annak megfelelően konfigurálni.

3. ábra  
UTRAN protokoll-architektúra



## 2) Adatfolyam továbbítás

Az UMTS vízszintes rétegei függőleges irányban két részre oszthatóak, C-síkra, és U-síkra. A C-sík felel meg a vezérlésnek, az U-sík pedig az adatfolyam továbbításáért felelős. A legfelső szinten külön rétegek felelősek a vezérlésért, és adattovábbításért, az alsóbb szinteken pedig ugyanaz a réteg valósítja meg mindkét funkciót.

A harmadik (L3) rétegbeli rádiós erőforrást vezérlő réteg (RRC) irányítja az UMTS többi rétegét. Vezérli az RLC, MAC és fizikai rétegeket és ez a réteg valósítja meg a felsőbb rétegek felé a kapcsolatok jelzését, ezzel biztosítva az adatfolyam megfelelő továbbítását az UMTS rendszeren keresztül. Az adatcsomag konvergencia protokoll réteg (PDCP) az RRC réteg mellett helyezkedik el. Ez a réteg kapja felsőbb rétegektől az IP adatcsomagokat, amit fejléctömörítés után SDU csomagokban továbbít az RLC rétegnek.

A rádiós link vezérlő réteg (RLC) szabványai három támogatott módot írnak le, AM (Acknowledged Mode) nyugtázott mód, UM (Unacknowledged Mode) nyugtázatlan mód és TM (Transparent Mode) transzparens mód. A nyugtázott mód használata ellentmondana az együttes optimalizálás elvének, mivel az JSCC/D kontroller által elért adaptáció nem adhatna jobb eredményt a hagyományos átvitelnél, ha mindig újraküldjük a meghibásodott adatcsomagokat (legalábbis csak nagyon szélsőséges esetben, a rádiós csatorna tartósan rossz minősége esetén, amikor az újraküldés sem vezet jobb átvitelre). Az UM és TM módok használata célravezető számunkra. Az RLC réteg szegmentálja a felülről kapott adatcsomagokat (eltérően UM és TM módok esetén), sorszámozza azokat (csak UM esetén), majd PDU egységekben továbbítja azokat a MAC rétegnek. UM, illetve TM mód esetén hibás csomagoknál nincsen újraküldés. A közeghozzáférési réteg (MAC) a kapott PDU-kat transzportcsatornákra képezi le esetleges paddinggel kiegészítve, majd a kiválasztott transzportformátummal együtt továbbítja azokat a fizikai réteg felé.

A fizikai réteg sok más funkció mellett CRC-t fűz az adatfolyamokhoz, csatornakódolás után pedig leképezi fizikai csatornákra azokat. A fizikai csatornákon levő adat QPSK moduláció után kerül továbbításra a rádiós interfészen keresztül. A komplex alapsávi jelként kiadott adat egy többutas fadinges rádiós átviteli közeget szimuláló modulba kerül, mely AWGN zajt ad a jelhez, mielőtt az a vevőhöz jutna. A vevő egy koherens gereblye-vevő (rake receiver), mely optimálisan becsli a csatorna jelútjainak (6 fő-terjedési, 20 alút) komplex csillapításait. A vevő után a vételi oldalon az egyes rétegek (fizikai, MAC, RLC, PDCP) inverz jelfeldolgozási folyamatai mennek végbe.

## 3) A modul vezérlése

Az UMTS rétegek vezérlését a rádiós erőforrás vezérlési réteg (RRC) valósítja meg. A Phoenix projekt szimulációs lánc együttes optimalizálást vezérlő paraméterei ezen a rádiós erőforrás vezérlési rétegen keresztül hatnak az UMTS rétegeire. Az UTRAN-ban az adat átvitelének módját az aktuálisan érvényes transzportformátum határozza meg, mely összefogja az RLC, a

MAC és fizikai rétegek beállításait. Az aktuálisan érvényes transzportformátumot a MAC réteg választja az RRC által beállított transzportformátum-halmazból. A kiválasztás algoritmusá figyelembe veszi az egyes adatok prioritását és az RLC bufferében lévő adat mennyiségét [4]. Ez az elsődlegesnek is nevezhető konfigurálás, melyet a MAC réteg végez viszonylag gyorsan nevezhető, mivel 10-80 ms időközökkel történik. Egy transzportformátum tartalmazza

- a csatornakódolás módját (turbó, konvolúciós, nincs kódolás);
- a csatornakódolás kódsebességét;
- az alapvető időtartomány alatt átvitt adatmennyiség (transzportblokk-méret, transzportblokkok száma);
- az alapvető időtartomány nagyságát (10, 20, 40 vagy 80 ms);
- az alkalmazott CRC méretét (0, 8, 12, 16, 24 bit);
- sebességillesztési paramétert (rate matching, puncturing) stb.

A RRC réteg azonban változtathatja a transzportformátumok azon halmazát, melyből a MAC választ. Ezt nevezhetjük másodlagos konfigurációnak, ami az elsődlegeshez képest lassan mehet csak végbe, mivel az adónak és a vevő egységnek szinkronizálnia kell ezt a halmazt egymással. A szimulációs láncunkban az JSCC/D által vezérelt adaptáció egy másodperces időközönként megy végbe, ami időben összemérhető az RRC által meghatározott formátumhalmaz kialakítással.

Az RRC réteg a JSCC/D-től kapott vezérlési információk alapján [7] választ egyet az előre elkészített konfigurációs beállítások közül. A beállítás-halmaz a [11] alapján lettek meghatározva, többek között tartalmazza a transzportformátumok halmazát, az RLC réteg által használt módot, a csatorna típusát, a payload és fejlec méreteket bitben, valamint a maximális átviteli sebességet stb. Egy szimulációs futtatás során a transzportformátumok halmazán kívül más nem változik. Ez természetesen mind feltöltési (uplink) és mind letöltési (downlink) irányokban szabályozható.

A szimulációban használt konfigurációs beállítások a szabványban leírt értékeknek felelnek meg [11], letöltési irányban 8-2048 kbps, feltöltési irányban pedig 8-384 kbps átviteli sebességet biztosítva a rendszernek.

A JSCC/D kontroller a teljes torzítás (DS+C) figyelembevételével választja ki az átvitt adat mennyiségének, a sebességillesztési paraméternek illetve csatornakódolási rátának a tartományát. Ezen adatok alapján az RRC választ az előre elkészített konfigurációs beállításokból, azon szempont szerint, hogy melyik beállítás felel meg legjobban a paramétertartományoknak.

A CRC használata is hathat a rendszer teljesítményére. Mint már említettük, a fizikai rétegben történik a CRC megfelelő adategységekhez (transzport blokkokhoz) való csatolása az adóoldalon, majd a vételi oldalon a hibás (nem megfelelő CRC ellenőrzőösszegű) csomagokat az RLC nem továbbítja a felsőbb PDCP rétegnek. Azonban, ha nem használunk CRC-t, akkor több hibás adat jut el a video dekódolóig, mivel kevesebb kerül eldobásra az UMTS-ben. A csomagok persze el-

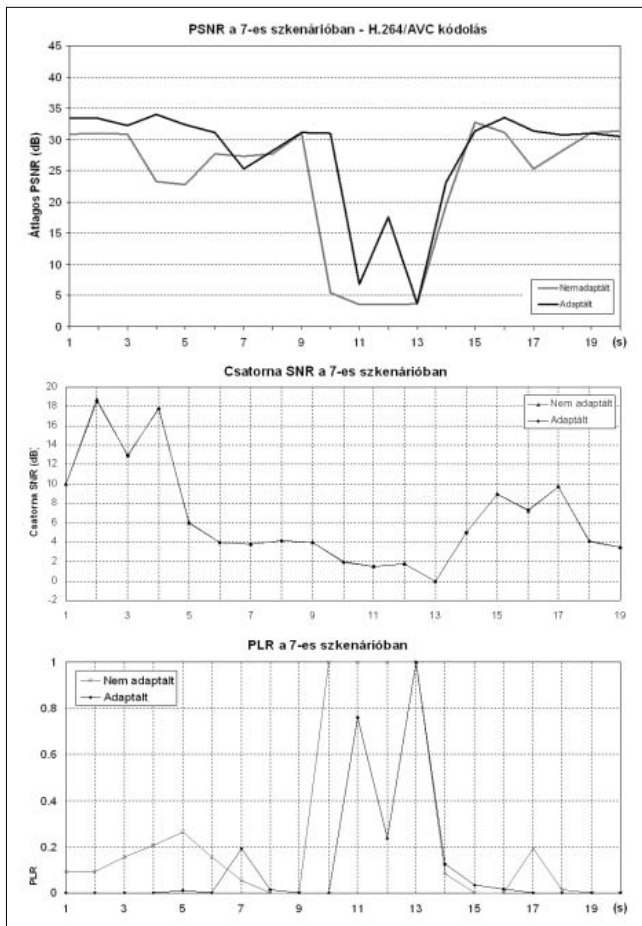
veszhetnek akkor is, ha nem használunk CRC ellenőrző összeget, mivel értelmezhetetlen vezérlő információk (szekvenciaszám, adathossz információk) esetén az RLC rétegben és a fejléc kitömörítéskor a PDCP rétegben is eldobódhatnak. A szimulációban az optimalizált eset teljesítményét javítottuk a CRC mellőzésével. A csomagvesztés ilyen módon való csökkentése természetesen teljesítménynövekedést okoz, azonban nem kizárólag ez felelős az adaptációval elérhető javulásért. A CRC használat hatásaira a [12] tartalmazza az általunk generált részletesebb szimulációs eredményeket.

## 6. Szimulációs eredmények

### A. Szimulációs futtatások, tesztbeállítások

Ebben a fejezetben egy futtatási példát emeltünk ki, bemutatva a szimulációs lánc teljesítményét. Az egyes modulok beállításai a projekt által leírt 7-es scenárió alapján történtek, a részletes beállítások megtalálhatók a [13] dokumentumban. Ez a scenárió az úgynevezett sürgős videóátvitelt (pushed video information) próbálja utánozni, mint például egy élő hírműsor, melynek jellemzője az alacsony késleltetés, többes adás és streaming mód, illetve mobil felhasználók jelenléte. Természetesen szimulációnk egy mobil állomás, mint vevő és a „híreket” adó másik végpont között szimulál átvitelt.

4. ábra Szimulációs eredmények a 7-es scenárióban mérve 20 sec videószekvencia esetén: átlagos PSNR, rádiós csatorna SNR és csomagvesztés (PLR)



A nyers videószekvencia a „foreman” és az „akiyo” QCIF formátumú YUV fájlok voltak, melyeket H.264/AVC forráskódolóval tömörítünk a szimulációs lánc alkalmazási rétegében. A szimulációs futtatás RTP, UDP-Lite szállítási protokollokkal történt, mely után IPv6 csomaggenerálás következett. Szimulált IPv6 hálózaton vezetjük keresztül a médiafolyamot, amelyet az UTRAN hálózati szegmensében lévő hálózati terminál vesz. Az RLC réteg által használt módot viszont nyugtázatlan UM módra állítottuk, amely a csomagok sorszámozása mellett nem tartalmaz újraküldési lehetőséget a hibás csomagok számára. Mint már említettük, ez az adaptáció miatt szükséges, mivel az adathibákat nem újraküldéssel, hanem együttes optimalizálással, az egész rendszer újrakonfigurálásával akarjuk csökkenteni, újraküldéssel az RLC réteg viszont elrejtene a hibás csomagokat a felsőbb rétegek előtt, ahol a döntés születik az együttes optimalizálásról. Rádiós csatorna szintjén a [14] alapján implementált mikrocellás belvárosi környezetet szimuláltunk. A vezeték nélküli UTRAN szakasz letöltési (downlink) irányban vitte át a videószekvenciát az 1. ábrának megfelelően.

Az együttes rétegek közötti vezérlővel (JSCC/D) optimalizált futtatást jelöltük adaptált esetnek. A nyers videóadat forráskódolása másodpercenként történik, amivel szinkronban, azaz másodpercenként történik az egyes modulok átkonfigurálása adaptált esetben. A nem adaptált esetben a JSCC/D kontroller nem működött és 384 kbit/s-os átvitelnek megfelelő transzport formátumot használt a MAC réteg [11]. Nem adaptált esetben 8 bites CRC-t használtunk, illetve adaptációnál kerültük a CRC használatát az UTRAN rendszerbeli átvitel során.

Adaptív és nem adaptív átvitelünk csatornakódolás szempontjából ugyanazon szabványos turbó csatornakódolást használnak. Jelentős különbség adódna a két eset között, ha nem azonos erősségű hibajavítást használnánk, azaz például nem adaptív esetben konvolúciós kódolást. Az adaptivitást így a kódoló típusára nem terjesztettük ki, mivel a viszonylag rossz minőségű rádiós környezet (fading hatás, alacsony jel-zaj viszony) megköveteli az erős csatornakódolás alkalmazását.

Az átvitt adat mennyisége azonban változik az engedélyezett transzport formátumok alapján, itt a sebességillesztési paraméter által meghatározott bitkiszűrés (puncturing), illetve az átviteli időintervallum (TTI) alatt átvihető bitek száma stb. A forráskódoló szintjén természetesen ugyancsak jelentősen változik az adaptáció hatására, azaz a tömörítés mértékére az adott másodpercben átvitt videó adat mennyisége.

### B. Futtatási eredmények

A 4. ábra jobb felső diagramja az átvitel egyes másodperceire vonatkozó átlagos jel-zaj viszonyt ábrázolja szimulált fadinges csatornán. A csomagvesztési arány (PLR – Packet Loss Ratio), mely az ábra jobb alsó részén látható, a két esetben eltér a csatorna ingadozása illetve az eltérő adatmennyiség miatt. A nem adaptált esetben a forráskódoló a rádiós csatornától függetlenül működve, mindig ugyanolyan minőségben kódol.

Adaptált esetben azonban az egyes másodpercekben különbözhet a videójel kódolása, illetve a fizikai rétegbeli konfigurációtól is függ a rádiós csatornára érkező adat mennyisége. Az átvitt videófolyam PSNR görbéje (az ábra bal oldali grafikonja) átlagosan 3-4 dB-lel jobb az adaptált esetben [GJ1].



Az 5. ábrán láthatóak a „foreman” QCIF és „akiyo” QCIF videó szekvenciák átviteléből pillanatképek ugyanazon képkockára vonatkoztatva adaptált és nem adaptált beállításoknál. A vizuális példán is látható, hogy bizonyos másodpercekben mennyivel jobb minőséget produkálhat a JSCC/D-vel vezérelt eset. Más futtatási környezetekre, beállításokra vonatkozó eredményeinket a projekt [13] dokumentuma tartalmazza.

## 7. Összefoglalás

A túlzott részletekre nem kiterjedő, de a szimulációs modell átfogó működését ismertető leírás után körvonalaztuk az UTRAN hálózati szegmens funkcionalitását és beillesztését a modellbe. Az egyes rendszerblokkok között áramló adat struktúráját is felvázoltuk, melyre a végponttól végpontig terjedő optimalizálás alapul. A valósághoz még jobban közelítettük rendszerünket egy általunk implementált, szabványokhoz nagymértékben igazodó, vezeték nélküli UMTS hálózati szegmens használatával. A UTRAN modul konfigurálhatóságát úgy alakítottuk, hogy megfelelően illeszkedjen a JSCC/D kontroller vezérlő interfészéhez.

Kapott eredményeink is mutatják az alkalmazási rétegek szintjén (forráskódoló) és fizikai réteg szintjén (csatornakódolás, moduláció stb.) használt együttes adaptáció nyereségét egy sokkal realisztikusabb környezetben, mint ami eddigi tudományos kutatásokban szerepelt. Remélhetőleg eredményeink a szabványosítók és a gyártók figyelmét is felkeltik majd, és a jövő rendszerei, berendezései alkalmazni fogják az együttes forrás- és csatorna(de)kódolás (JSCC/D) alapelvét.

## Köszönetnyilvánítás

A szerzők köszönetüket fejezik ki a PHOENIX projekt minden résztvevőjének, akik értékes munkájukkal hozzájárultak a projekt szimulációs rendszerének kifejlesztéséhez, lehetővé téve jelen munka megszületését is.

## Irodalom

- [1] C. E. Shannon, „A mathematical theory of communication”, Bell System Technical Journal, July-October 1948. Vol. 27., pp.379–423; 623–656.
- [2] S. B. Zahir Azami, P. Duhamel, O. Rioul, „Joint source-channel coding: panorama of methods”, in Proc. of CNES workshop on data compression, Toulouse, France, November 1996.
- [3] Final Draft Int. Standard of Joint Video Specification (ITU-T Rec. H.264, ISO/IEC 14496-10 AVC), Doc JVT-G050r1, Geneva, Switzerland, May 2003.
- [4] A. Zsiros, A. Fülöp, G. Jeney, „Easily Configurable Environment of UTRAN Physical Layer”, in Proc. 5th EURASIP Conf. Speech and Image Processing, Multimedia Communications and Services, Smolenice, Slovakia, 2005, pp.93–98.
- [5] M. van der Schaar, S. Krishnamachari, S. Choi, X. Xu, „Adaptive Cross-Layer Protection Strategies for Robust Video Transmission Over 802.11 WLANs”, IEEE Journal on Select. Areas Comm. Vol. 21., Dec. 2003, pp.1752–1763.
- [6] C. Lamy-Bergot, J. Huusko, M. G. Martini, P. Amon, C. Bergeron, P. Hammes, G. Jeney, S. X. Ng, G. Panza, J. Peltola, F. Sidoti, „Joint optimisation of multimedia transmission over an IP wired/wireless link”, European Symposium on Mobile Media Delivery, September 2006.
- [7] ETSI Technical Standards, TS 125.212 V6.1.0 (2004-03), UMTS „Multiplexing and Channel Coding (FDD)”
- [8] ETSI Technical Standards, TS 125.301 V6.0.0 (2003-12), UMTS „Radio Interface Protocol Architecture”
- [9] ETSI Technical Standards, TS 125.211 V6.0.0 (2003-12), UMTS „Physical Channels and Mapping of Transport Channels onto Physical Channels (FDD)”
- [10] ETSI Technical Standards, TS 125.213 V6.0.0 (2003-12), UMTS „Spreading and Modulation (FDD)”
- [11] ETSI Technical Standards, TS 134.108 V6.0.0 (2005-10), UMTS „Common test environments for User Equipment (UE) conformance testing”
- [12] M. G. Martini, M. Mazotti, C. Lamy-Bergot, P. Amon, G. Panza, J. Huusko, J. Peltola, G. Jeney, G. Feher, S. X. Ng, „A Demonstration Platform for Network Aware Joint Optimization of Wireless Video Transmission”, IST Mobile Summit, Juny 2006.
- [13] „Deliverable 4.4b: Experiments and test results”, PHOENIX project deliverable, December 2006.
- [14] ETSI Technical Report, TS 125.996 V6.1.0 (2003-09), UMTS „Spatial Channel Model for Multiple Input Multiple Output (MIMO) Simulations”