



M Ű E G Y E T E M 1 7 8 2

Budapesti Műszaki és Gazdaságtudományi Egyetem
Villamosmérnöki és Informatikai Kar
Híradástechnikai Tanszék

Pásztor András

QoS támogatás vizsgálata LTE alapú femtocellák ütemezési kérdéseiben

TDK DOLGOZAT

KONZULENS

Dr. Imre Sándor

BUDAPEST, 2012

Tartalomjegyzék

1. Bevezetés	5
2. Mobil evolúció.....	7
2.1 Áramkörkapcsolt adat	7
2.2 GPRS	8
2.3 UMTS	10
2.4 HSPA.....	11
2.4.1 HSDPA.....	11
2.4.2 HSUPA.....	12
2.5 Összegzés	12
3. LTE.....	15
3.1 Az LTE architektúra	15
3.2 Közeghozzáférés	17
3.3 Az LTE antennái	17
3.4 Összefoglalás.....	17
4. Az LTE rádiós interfésze	19
4.1 OFDMA	19
4.2 SC-FDMA	20
4.3 Az LTE ütemezés	21
4.3.1 Az LTE keretszerkezete	21
4.3.2 Downlink ütemezés	23
4.3.3 Uplink ütemezés.....	23
5. Cellás mobil rendszerek.....	25
5.1 Makrocellás rendszerek.....	27
5.2 Mikrocellás rendszerek.....	28
5.3 Pikocellás rendszerek	29
5.4 Femtocellás rendszerek	29
6. Quality of Service	31
7. A szimulációs környezet.....	33
7.1 A szimulátor felépítése.....	33
7.2 A szimulátor működése	34
8. A szimuláció kiértékelése.....	37

8.1 Round Robin algoritmus	37
8.2 Best CQI algoritmus	39
8.3 Fair Queueing algoritmus	41
8.4 Weighted Fair Queueing algoritmus	44
8.5 Az algoritmusok összefoglalása.....	46
8.6 A fairness meghatározása	48
9. Összefoglalás.....	51
Rövidítésjegyzék.....	53
Ábrajegyzék.....	55
Irodalomjegyzék.....	57

1. Bevezetés

Az elmúlt évtizedekben a számítástechnika szinte elképzelhetetlen mértékű fejlődésen esett át. Azok a technológiák, szabványok, amelyek a múlt század végén születtek meg, mára váltak igazán kiforrottá. Mind az internet, mind az ahhoz kapcsolódó végkészülékek megkerülhetetlen részévé váltak életünknek. Az interneten elérhető lehetőségek tárháza szinte kimeríthetetlen, így egyre többen használják napi rendszerességgel, egyre gyakoribb, hogy járműveken, vezetékes elérésre alkalmatlan helyeken is rákényszerülünk arra, hogy letöltsük e-mailjeinket, vagy épp online videókat nézzünk. A megnövekedett forgalmat azonban nehéz kezelni, főleg úgy, hogy az elsődleges szempont minden esetben a felhasználói élmény maximalizálása kell, hogy legyen. Ebben a dolgozatban bemutatom az újgenerációs mobil hálózatok felépítését és működését, valamint ütemezési kérdéseken keresztül vizsgálom, hogy hogyan lehet a felhasználókat a lehető legjobban kiszolgálni, hogyan lehet számukra garantált minőségű szolgáltatást nyújtani.

A dolgozat első részében – korábbi rendszerek ismertetésén keresztül – röviden bemutatom azt a folyamatot, amely a jelenlegi modern mobilhálózatokhoz vezetett. Rámutatok azokra a főbb jellemzőkre, amelyek komoly átalakuláson mentek keresztül, amíg elérték mai formájukat. Miután tisztáztam az alapokat, ismertetem a mai modern hálózatok működését. Ezt a jelenleg is aktuális 3GPP (3rd Generation Partnership Project) szabványok figyelembevételével fogom vizsgálni. Ezután áttekintem a garantált minőségű szolgáltatás nyújtásának szabályait, majd egy saját szimulációs környezetben vizsgálom a megvalósítás lehetőségeit. A vizsgálat alapját ütemezési algoritmusok adják. Különböző algoritmusok esetén fogom tanulmányozni az átvitelt és ezek alapján általános megfontolásokat fogok megfogalmazni QoS (Quality of Service) nyújtására vonatkozóan.

2. Mobil evolúció

A mobil telekommunikáció, azon belül is a mobil adattovábbítás a múlt évszázad végén indult komoly fejlődésnek. Az internet egyre szélesebb körben való elterjedése azt hozta magával, hogy a felhasználók elvárása megnövekedett. Míg eleinte a kisméretű adatok továbbítása (e-mail, böngészés) volt a meghatározó, később – a századforduló környékén – megjelentek például a fájlcsere-lők, amelyek sokkal nagyobb sávszélességet igényeltek. A nagyobb sávszélesség-igényű feladatok, valamint az internetfelhasználók számának drasztikus növekedése hozta magával a különböző technológiák fejlődését is. Ebben a fejezetben áttekintem a különböző mobilinternet szabványok főbb jellemzőit, különös hangsúlyt fektetve arra, hogy bemutassam azt a folyamatot, amelynek eredményeképpen kialakult az eddigi legmodernebb mobilinternetes szabvány, az LTE (Long Term Evolution).

2.1 Áramkörkapcsolt adat

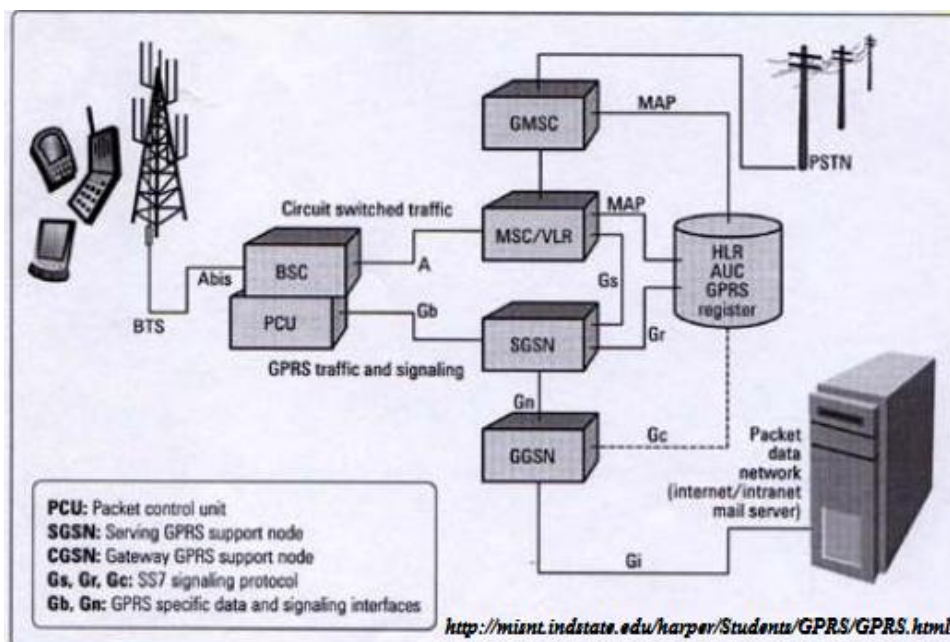
A mobil hálózaton történő adattovábbítás első lépése a CSD (Circuit Switched Data) volt. Ezt a magyar nyelvű szakirodalomban áramkörkapcsolt vagy vonalkapcsolt adatnak nevezik. A technológia elve igen egyszerű: a hangátvitelre szánt időrések egyikében hang helyett adatot továbbít a rendszer. Ez alapján elmondható, hogy áramkörkapcsolt adatot csak TDMA-t (Time Division Multiple Access) alkalmazó hálózatokban lehetett használni. Ezzel a megoldással a maximális adatátviteli sebesség 9.6 kbps volt [1]. Ez ma már mosolyt csal az arcunkra, a '90 években viszont önmagában már a mobil adatkommunikáció nagyon komoly fejlődésnek számított.

Hamar megjelent a CSD továbbfejlesztett változata, azaz a HSCSD (High Speed Circuit Switched Data), a nagysebességű áramkörkapcsolt adat. Ez a technológia az elődhez képest annyiban változott, hogy egy helyett négy időrést használt adattovábbításra, ezzel négyszeresére növelve az átviteli sebességet. Az így elérhető sebesség 38.4 kbps volt. Ezt gyorsan sikerült tovább növelni, hiszen kidolgoztak egy csökkentett méretű hibajavító kódot, aminek következtében több hasznos adat fért el egy időrésben. Így az áramkörkapcsolt adatok maximális átviteli sebessége 14.4 kbps időrésenként, vagyis összesen 57.6 kbps [1].

Noha az áramkörkapcsolt adat komoly fordulópontra volt a mobil távközlés történetében, mégsem terjedt el igazán. Erre csak utódja, a GPRS (General Packet Radio System) volt képes.

2.2 GPRS

A GPRS (General Packet Radio Service) szabvány – mint a GSM (Global System for Mobile communications) hálózatok adatátviteli szabványa – nagy ugrás volt a mobil adattovábbítás szempontjából. Ahogyan azt a GPRS „P” betűje is mutatja, az átvitel csomagkapcsolt lett, tehát IP (Internet Protocol) alá került. Emiatt azonban az eddigi hálózatot ki kellett egészíteni egy csomagalapú hálózattal. Ennek értelmében a BSC-hez (Base Station Controller) csatlakoztattak egy szerkezeti egységet, amely a PCU (Packet Control Unit) nevet kapta. Ennek a feladata, hogy a BSC-be beérkező forgalmat kettéválassza [2]. Egy GPRS hálózat felépítése látható a 2.1 ábrán.

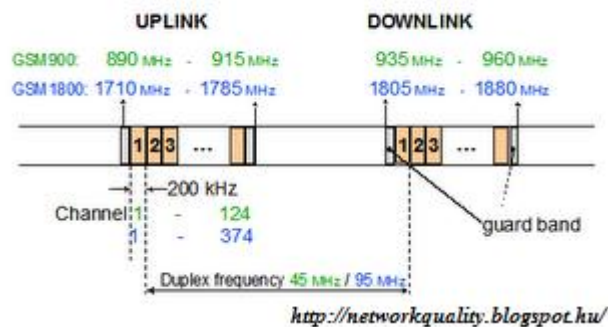


2.1 ábra – Egy GPRS hálózat felépítése

Az áramkörkapcsolt adatot az eddig is meglévő hálózaton keresztül továbbítják, viszont a csomagkapcsolt adatot egy csomagkapcsolt hálózat felé küldik. Ehhez azonban ki kellett építeni egy GPRS maghálózatot. Ezt képviseli az SGSN (Serving

GPRS Support Node) és a GGSN (Gateway GPRS Support Node) [2]. A adat ezen a két egységen áthaladva kerül végül a kívánt csomagkapcsolt hálózatba.

A GPRS hálózatok TDMA elven működnek, azonban szétválasztásra kerül a végkészülék irányába (downlink) és a végkészülék irányából (uplink) érkező adat. Ezeket FDD (Frequency Division Duplex) segítségével osztják meg [2]. Ez azt jelenti, hogy a downlink és az uplink adat különböző vivőkön kerül továbbításra. A két legelterjedtebb, máig üzemelő GSM rendszer a 900 MHz-es és az 1800 MHz-es GSM. 900 MHz-es technológia esetén az uplink – szolgáltatótól függően – 890 és 915 MHz között helyezkedik el, míg a downlink frekvenciák 935 és 960 MHz között vannak. 1800 MHz-es esetben az uplink 1710 és 1795 MHz között, míg a downlink 1805 és 1880 MHz között található. Az operátorok az általuk megvásárolt frekvenciasávokat csatornákra osztják és ebben végzik az átvitelt. Egy GSM csatorna sávszélessége 200 kHz. A vivőket GMSK (Gaussian Minimum Shift Keying) segítségével modulálják [2]. A GSM sávszélesség menedzsmentjét a 2.2 ábra mutatja.



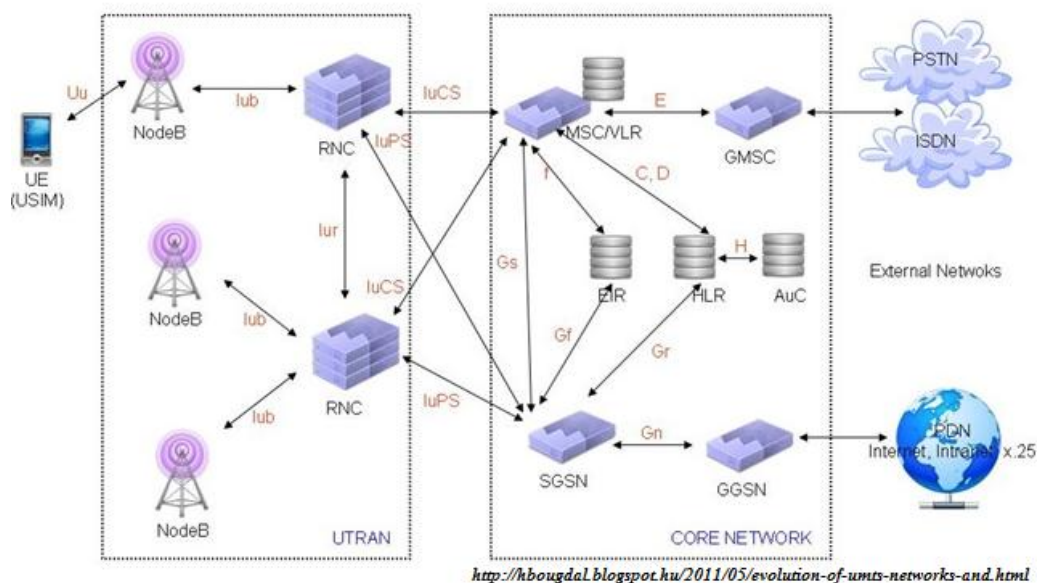
2.2 ábra – GSM/GPRS sávszélesség-menedzsment

A GPRS-nek létezik egy továbbfejlesztése, amely az EDGE (Enhanced Data rates for GPRS Evolution) nevet viseli. Ennél a technológiánál nem történt további változtatás a hálózatban, kizárólag technológiai jellemzőkön változtattak. Ezek közül a legfontosabb, hogy jó minőségű csatornán kevésbé robusztus, ám jobb átvitelt biztosító modulációt alkalmaznak. Ez a 8-PSK (Phase Shift Keying) volt [2]. Ennél a modulációnál a szimbólumok fázisa hordozza az információt és 1 szimbólummal 3 bitet képes átvinni.

Noha ezek a technológiák még mindig nem biztosítottak jelentős adatátviteli sebességet (EDGE esetén a maximális átvitel 384 kbps [2]), miattuk alakult ki az a hálózati struktúra, amely éveken keresztül megfelelően működött és működik ma is. Emellett fontos megjegyezni, hogy a mai hálózatok felépítése is mind visszavezethető a GPRS technológiánál használt hálózati architektúrára.

2.3 UMTS

A GPRS, illetve az EDGE utáni következő lépcsőfok a UMTS (Universal Mobile Telecommunications System) volt. Ez a szabvány már képes 1 Mbps névleges átviteli sebességű link kiépítésére, ezért ez az első 3G-s technológia. A UMTS nem változtatott a GPRS idején kiépített maghálózaton, fejlődés történt azonban a rádióhozzáférési hálózatban, megjelent az UTRAN (Universal Terrestrial Radio Access Network). Ebben két szerkezeti egység szerepel, amik a GPRS rádióhozzáférési hálózatának BTS (Base Transceiver Station) és BSC (Base Station Controller) egységei helyett jelentek meg. Az egyik a NodeB, a másik pedig az RNC (Radio Network Controller) [3]. Az UTRAN-ból a maghálózat felé már titkosított adat kerül továbbításra. Egy UMTS architektúrát mutat a 2.3 ábra.



2.3 ábra – A UMTS architektúra

A UMTS sebességének növekedéséért kis túlzással egyetlen tényező volt felelős. Ez pedig az, hogy az addig használt 200 kHz-es csatornát 5 MHz-re bővítette a szabvány [3]. Erre elsősorban azért volt szükség, mert új közeghozzáférési technológiát alkalmaztak. Az új technológia a CDMA (Code Division Multiple Access), aminek értelmében a jel kis teljesítménnyel, de az egész spektrumra szétszórva kerül kisugárázásra. A GPRS-hez hasonlóan UMTS esetében is használják az FDD technológiát az uplink és a downlink elkülönítésére.

2.4 HSPA

A UMTS továbbfejlesztése a HSPA (High Speed Packet Access), amely még szintén 3G-s szabvány. A HSPA-t két különálló szabvány összegeként alkották meg. A letöltés fejlesztését a HSDPA (High Speed Downlink Packet Access), míg a feltöltés fejlesztését HSUPA (High Speed Uplink Packet Access) szabványokban definiálták. A közeghozzáférés ebben a szabványban is kódosztáson alapult.

2.4.1 HSDPA

A HSPA és ezáltal a HSDPA legfontosabb követelményei a névleges átviteli sebesség növelése és a késleltetés csökkentése. Az átviteli sebességet elsősorban a szóró kódok hosszával tudják befolyásolni, a szóró kódok méretének csökkentése a sebesség növekedésével jár. Emelett új modulációs eljárásokat alkalmaznak. Rossz minőségű csatornán a QPSK (Quadrature Phase Shift Keying) moduláció a mérvadó, viszont a csatorna javulásával 16-QAM (Quadrature Amplitude Modulation), vagy igazán jó minőségű csatornán 64-QAM modulációt használnak [4]. Ez szintén jelentős sebességnövekedéshez vezet. A csatorna minőségét CQI (Channel Quality Indicator) értékekkel jellemzik. A NodeB-k referenciajeleket sugároznak a terminálok felé, abból a csillapításból pedig, amit a jel elszenvedett, a terminál meghatározza a saját csatornája CQI értékét, majd ezt visszaküldi a NodeB-nek.

A késleltetést a hálózat „kilapításával” lehet csökkenteni. Ennek érdekében az RNC számos funkcióját ebben a szabványban már a NodeB látja el.

Fejlesztésként bevezetésre került a HARQ (Hybrid Automatic Repeat Request) [4]. Ennek értelmében az adatot hibajavító bitekkel együtt viszik át. Ezek segítségével a vevő a kis hibákat ki tudja javítani, csak nagyobb problémák esetén van szükség újraküldésre. A vevő azonban minden esetben eltárolja a hibás csomagot is. Erre azért van szükség, mert ha az újraküldött adatok is hibásak, előfordulhat, hogy a kettő „kombinációjából” hibamentes csomagot lehet előállítani. Ezt nevezik növekményes redundanciának.

2.4.2 HSUPA

A HSUPA szabványt azért kellett definiálni, mert a 3GPP a HSDPA-ban egyáltalán nem foglalkozott a feltöltés fejlesztésével. A szabvány legfontosabb eleme egy új, a feltöltéshez szükséges csatorna létrehozása, melyet E-DCH (Enhanced Dedicated Channel) csatornának neveznek [5].

2.5 Összegzés

Az eddigiekben láthattuk, hogy mely szempontok alapján változtatták a hálózat egyégeit, vagy éppen az alkalmazott technológiák paramétereit. Összegzésképpen el lehet mondani, hogy a mobil evolúció négy alapvető pilléren nyugszik. Döntően ez a négy pillér határozza meg a az átviteli sebesség növekedését.

Az első ilyen pillér a sávszélesség növelése. A (2.1) alapján jól látszik, hogy a csatorna kapacitása egyenes arányban áll a sávszélességgel (ellenben a jel-zaj viszonyal csak logaritmikus a viszony) [6]. Emiatt a sávszélesség növelése egy jó módja annak, hogy növeljük a csatorna kapacitását, ezáltal pedig az átvihető bitek számát.

$$C = B \log_2(1 + SNR) \quad (2.1)$$

ahol C a kapacitás, B a hasznos jel sávszélessége, az SNR pedig a jel-zaj viszony.

A második pillér a modulációs módszerek fejlesztése. Minél nagyobb fókuszú egy moduláció, annál több bitet tudunk átvinni egyetlen szimbólum segítségével. Az EDGE technológiánál említettem, hogy a 8-PSK moduláció esetén egy szimbólum 3 bitet hordoz. Ezzel szemben például a 64-QAM modulációnál egy szimbólum már 6 bitet hordoz. Mivel egy technológia esetén meghatározott időnként küldünk egy

szimbólumot, ezért elmondható, hogy ha több bitet hordoz az adott szimbólum, akkor nőni fog a bitsebesség. A szimbólumonként átvihető bitek számát általánosan a (2.2) írja le

$$b = \log_2(M) \quad (2.2)$$

ahol b az átvihető bitek száma, M pedig a moduláció foka.

A harmadik pillér, amely a sebességnövekedéshez járul a hálózat „kilapítása”. Általános megfontolás az, hogy ha a hálózat peremére integrálunk bizonyos funciókat, akkor csökkeni fog a késleltetés. Ennek az az oka, hogy a küldött adatnak nem kell bejárni a hálózat összes szerkezeti egységét, elég csak a peremen lévőket, vagy lévőket. Ennek következtében az adat kevesebb időt tölt a hálózatban, így csökkentve a késleltetést. Ugyan a késleltetés csökkenése nem egyenlő a sebesség növekedésével, mégis azt lehet mondani, hogy bizonyos forgalmak esetén nagyon fontos tényező lehet.

Létezik egy negyedik pillér is, amely csak a HSPA+ szabvány (a HSPA továbbfejlesztése) esetén került bevezetésre, viszont a HSPA+ utáni szabványok mindegyike alkalmazza. Ez a MIMO (Multiple Input Multiple Output) antennarendszerek alkalmazása. A MIMO a SISO (Single Input Single Output) rendszerek utódja. Lényege, hogy az átvinni szándékozott jelet több adóantenna segítségével küldjük és több vevőantenna segítségével vesszük, ezáltal megnöveljük a csatorna kapacitását. Ebben az esetben a kapacitást a (2.3) írja le

$$C = MB \log_2(1 + SNR) \quad (2.3)$$

ahol C a kapacitás, M az átvinni kívánt adatfolyamok száma, B a hasznos jel sávszélessége, az SNR pedig a jel-zaj viszony [6]. Látható tehát, hogy a (2.1) és a (2.3) között egy M szorzó a különbség. M maximális értéke a (2.4) által számítható

$$M = \min(Tx, Rx) \quad (2.4)$$

ahol Tx az adóantennák száma, Rx pedig a vevőantennák száma [12]. Ezek alapján elmondható, hogy például 4x4 MIMO (4 adóantenna és 4 vevőantenna) alkalmazása esetén a csatorna kapacitása négyszeresére nő a SISO esethez képest.

Ebben a fejezetben bemutattam a mobil evolúció fontosabb lépcsőfokait és az ezeknél alkalmazott technológiákat. Fontos azonban megjegyezni, hogy ez a folyamat sokkal összetettebb. Ebben a dolgozatban a célom azoknak a funkcióknak a bemutatása, amelyek fejlődése a felhasználói élmény drasztikus növekedését okozzák, ezért gyakran

pusztán ezeknek a funkcióknak a fejlesztése megfelelő minőségű hálózatot teremt. Helyszűke miatt nem taglalom részletesebben az újgenerációs mobilhálózatokat megelőző hálózatok struktúráját.

3. LTE

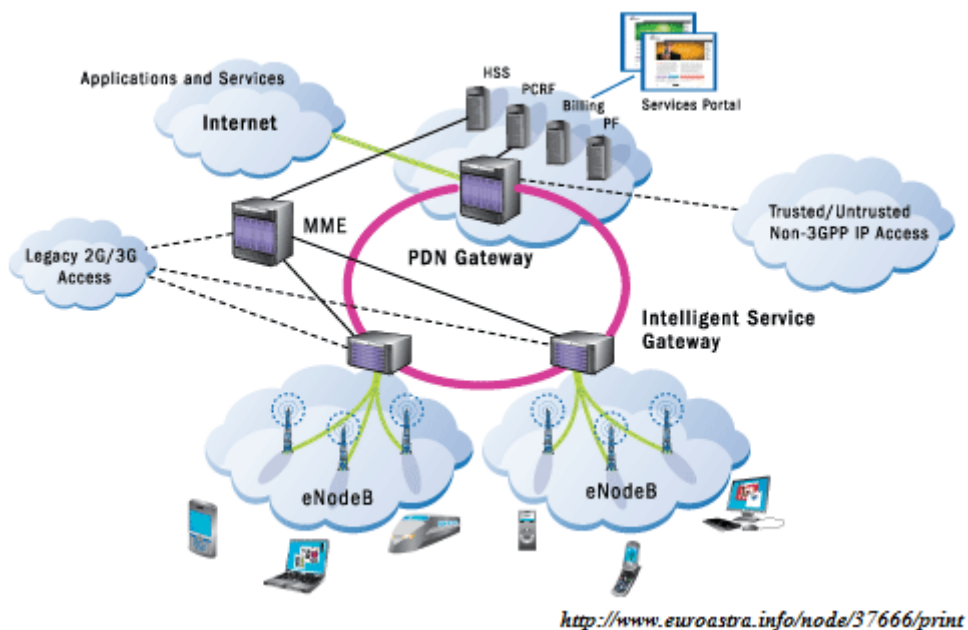
Az LTE (Long Term Evolution) a 3GPP által definiált mobil-kommunikációs szabvány, amely a HSPA hálózatokat hivatott modernizálni. Az LTE nem továbbfejlesztése a HSPA-nak, hanem egy teljesen újszerű, új maghálózattal rendelkező technológia, amely nem is kompatibilis az öt megelőző 2G/3G-s hálózatokkal. Mivel azonban nem lehet gyorsan mindenhol lecserélni a hálózatot, ezért egy alapvető kompatibilitást lehet biztosítani, különböző regisztereken keresztül. Lényege, hogy kizárólag csomagkapcsolt adatokat használ, vagyis a hangátvitel is IP alá kerül. Az LTE tulajdonképpen mindent modernizál a előző hálózatokhoz képest. A megváltozott maghálózaton túl változik a közeghozzáférési technológia, az antennarendszer és a spektrumkihasználás is. A változtatásokra azért volt szükség, hogy az alábbi követelményeket kielégítsék [7]:

- Legalább 100 Mbps downlink és 50 Mbps uplink elméleti maximális sebesség biztosítása 20 MHz-es spektrum használata esetén.
- FDD és TDD támogatása.
- Maximum 5 ms csomagkésleltetés alacsony terhelésnél.
- 5 MHz-en legalább 200, nagyobb spektrumon legalább 400 felhasználó egyidejű kiszolgálása egy cellában.
- Többféle sáv szélesség támogatása.
- 120 km/h-ig nagy teljesítőképesség, de 350 km/h-ig fenn kell tudni tartani a kapcsolatot.
- 30 km-es lefedettség.

3.1 Az LTE architektúra

Az LTE teljesen új hálózatot kapott. A legnagyobb változást a maghálózat „szervedte” el, de a RAN (Radio Access Network) is átalakult. A fejlesztett maghálózatot EPC-nek (Evolved Packet Core), a fejlesztett rádió-hozzáférési hálózatot E-UTRAN-nak (Evolved Universal Terrestrial Radio Access Network), míg a kettőt

együtt SAE-nak (System Architecture Evolution) hívják [8]. Az átalakulás abban nyilvánul meg, hogy a HSPA-ban elkezdett késleltetés csökkentést tovább folytatja a 3GPP, ennek értelmében még „laposabb” hálózatot hoztak létre. Emiatt az RNC teljesen eltűnik, a funkcióit a NodeB veszi át. Az így létrejött egység új nevet is kapott, ami az eNodeB (Evolved NodeB). A maghálózat nagyon hasonlít az eddig használatoshoz. Két egységet találhatunk benne, egyet a RAN felőli oldalon, egyet pedig az internet felőli oldalon. A RAN felőli oldalon található egység neve S-GW (Serving Gateway), az internet felőli pedig PDN-GW (Packet Data Network Gateway). Az LTE architektúrában találhatunk még egy fontos egységet, az MME-t (Mobility Management Entity). Az MME a hálózat fő egysége. Ez felelős a terminál helyének lekérdezéséért, az útvonalválasztásért, illetve gyakorlatilag az összes vezérlési funkciót ez látja el. Ez felelős továbbá, azért hogy az LTE mégis kompatibilis legyen az őt megelőző 3GPP szabványokkal. Az MME-hez kapcsolódik a HSS (Home Subscriber Server). Ez tárolja a felhasználók adatait. Hasonló funkciókat lát el, mint a korábbi szabványok HLR-je (Home Location Register) és AuC-je (Authentication Center) együtt. Az LTE hálózat architektúráját a 3.1 ábra mutatja.



3.1 ábra – Az LTE architektúra

3.2 Közeghozzáférés

Az LTE szabványban nem alkalmazzák tovább a kódosztásos technológiát, hanem visszatérnek a korábban is alkalmazott frekvenciaosztásra. Ezen belül is downlink esetben az OFDMA (Orthogonal Frequency Division Multiple Access), míg uplink esetben az SC-FDMA (Single Carrier Frequency Division Multiple Access) technológiát választották [9]. A két technológiát részletesen a következő fejezetben ismertetem.

A modulációs eljárásban nem történt változás, vagyis továbbra is QPSK, 16-QAM, vagy 64-QAM modulációt használnak a CQI-tól függően [9]. Minél gyengébb a csatornából kivehető jel, vagyis minél rosszabb a CQI értéke, annál robosztusabb modulációra van szükség.

Fontos követelménye az LTE-nek, hogy mind az FDD, mind a TDD (Time Division Duplex) átvitelt támogatnia kell. Az alkalmazott sáv szélesség pedig 1.4 MHz-től egészen 20 MHz-ig terjedhet [9]. A lehetséges sáv szélességek: 1.4, 3, 5, 10, 15 és 20 MHz.

3.3 Az LTE antennái

Az előző fejezet végén bemutattam, hogy az átviteli sebesség növelésének egyik fontos, gyakran használt technikája a MIMO antennarendszerek alkalmazása [6]. Ezt a technológiát az LTE is használja, így támogatja a 2x2, illetve a 4x4 MIMO antennarendszereket. Érdeemes megjegyezni azonban azt, hogy LTE esetén SISO antennarendszerek alkalmazásával jellemzően néhányszor 10 Mbps átviteli sebességet lehet elérni, vagyis ennél nagyobb sebesség eléréséhez kifejezetten szükséges MIMO antennákat alkalmazni.

3.4 Összefoglalás

Láttuk tehát, hogy az LTE néhányszor 100 Mbps nagyságrendű elméleti átviteli sebesség biztosítására képes, viszont ennek komoly ára van. A legnagyobb probléma,

hogy mivel az előző szabványok leváltása lenne a cél, ezért az LTE-nek nem kell kompatibilisnek lennie az elődeivel. Az új architektúra kiépítése viszont idő- és pénzigényes.

Az LTE főbb előnyei:

- Korszerű rádióhálózat
- Az elődökhöz képest rugalmas frekvenciahasználat
- Csomagkapcsolt forgalom preferálása
- Az erőforrások hatékony használata

Az LTE hátrányai:

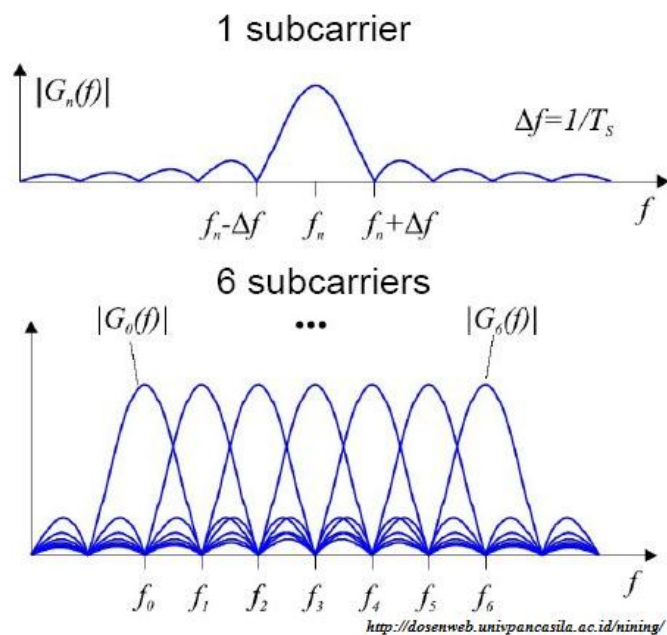
- Nagy beruházás, az egész, már meglévő hálózatot le kell cserélni
- Visszefelé nem feltétlenül kompatibilis

4. Az LTE rádiós interfésze

Az előző fejezetben bemutattam, hogy melyek az LTE szabvány főbb jellemzői, mik azok a fejlesztések, amelyek sebességnövekedést okoznak. Ahhoz azonban, hogy megérthessük az ütemezési eljárások működését, meg kell értenünk a rádiós interfész működését is. Ebben a fejezetben ezt fogom bemutatni. Első lépésként tekintsük át a downlinkben és az uplinkben használatos multiplexálási eljárásokat. Ezek – mint már említettük – az OFDMA és az SC-FDMA [9].

4.1 OFDMA

Az OFDM egy többvivős közegehozzáférési technológia, amelyet többek között az LTE is használ. Lényege, hogy, ha a különböző alvivőket a frekvenciatartományban meghatározott távolságra helyezük egymástól, akkor elkerülhető a alvivők között létrejövő interferencia [10]. Ennek az az oka, hogy adott távolság mellett, ha egy vivő spektruma felveszi a maximumot, akkor az összes többi vivő értéke 0 lesz. Ezt a tulajdonságot a 4.1 ábra demonstrálja.



4.1 ábra – Az OFDM elve

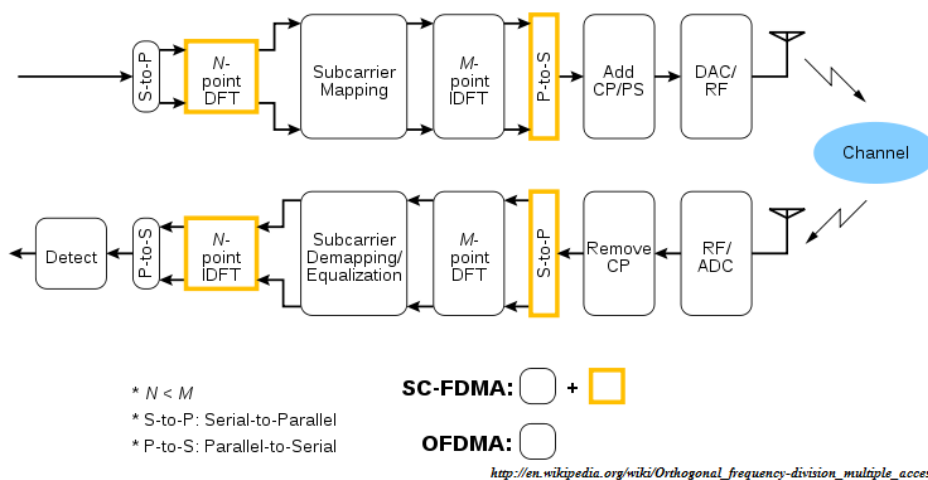
Minden alvivő külön-külön modulálható. Az LTE által használt modulációs eljárások a QPSK, a 16-QAM, illetve a 64-QAM. A modulált jeleket IFFT (Inverse Fast Fourier Transform) alkalmazásával lehet időtartománybeli szimbólummá alakítani.

Az OFDMA az OFDM többfelhasználós változata. Lényegében az ortogonális alvivők az OFDMA esetében különböző felhasználók adatait képesek hordozni, így érhető el az, hogy az adott spektrumban több felhasználó egyidejűleg folytathat kommunikációt.

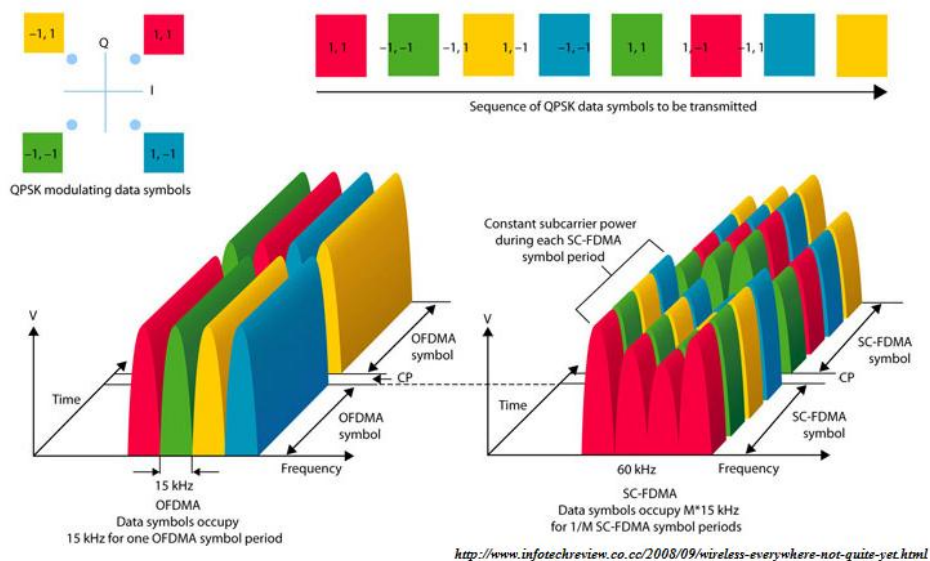
4.2 SC-FDMA

Az OFDM nagy problémája, hogy nagyon nagy a PAPR (Peak-to-Average Power Ratio) értéke, ezért gyengébb minőségű erősítők torzítanak a jelet. Mivel azonban kifejezetten jó minőségű erősítők használata a felhasználói terminálokban nem megoldható, ezért a feltöltést más technológia oldja meg. Ez a technológia az SC-FDMA [9].

Az SC-FDMA két dologban különbözik az OFDMA-tól. Az egyik, hogy a jel küldés előtt még átesik egy DFT-n (Discrete Fourier Transform), a másik, hogy nem párhuzamosan, hanem sorosan történik a küldés. Ezek miatt az igényelt sávszélesség N szimbólum esetén $B = N\Delta f$. Ezzel szemben viszont N -ed részére csökken a szimbólumidő, így az SC-FDMA átvitel körülbelül hasonló teljesítményt képes nyújtani mint az OFDM. A 4.2 ábrán a két jel előállításának közötti különbségek látszanak, míg a 4.3 ábra a két technológia idő-frekvencia képét mutatja, QPSK modulációt feltételezve.



4.2 ábra – Az OFDMA és az SC-FDMA összehasonlítása



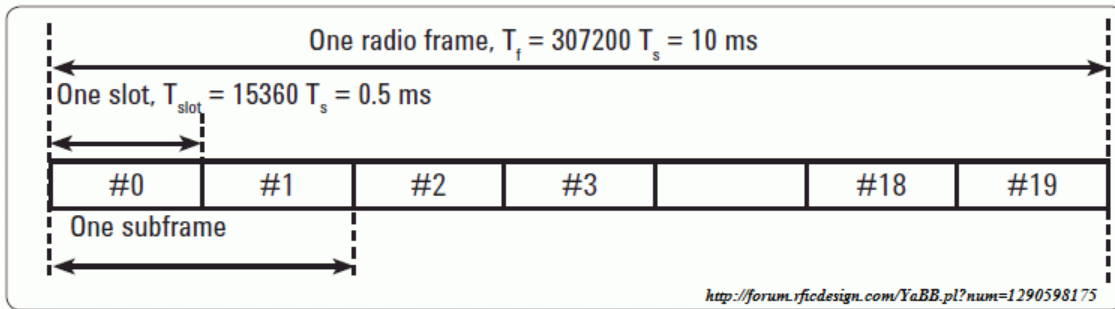
4.3 ábra – Az OFDMA és az SC-FDMA idő-frekvencia diagramja QPSK esetén

4.3 Az LTE ütemezés

Az ütemezés lényege az, hogy egy döntési algoritmus segítségével az ütemező kiosztja a rendszer erőforrásblokkjait. Vagyis az ütemező összeállítja a küldendő keretet. Az LTE-ben a MAC (Medium Access Control) réteg felelős az ütemezés kezeléséért [9]. Nézzük most meg, hogy hogyan áll össze egy LTE keret.

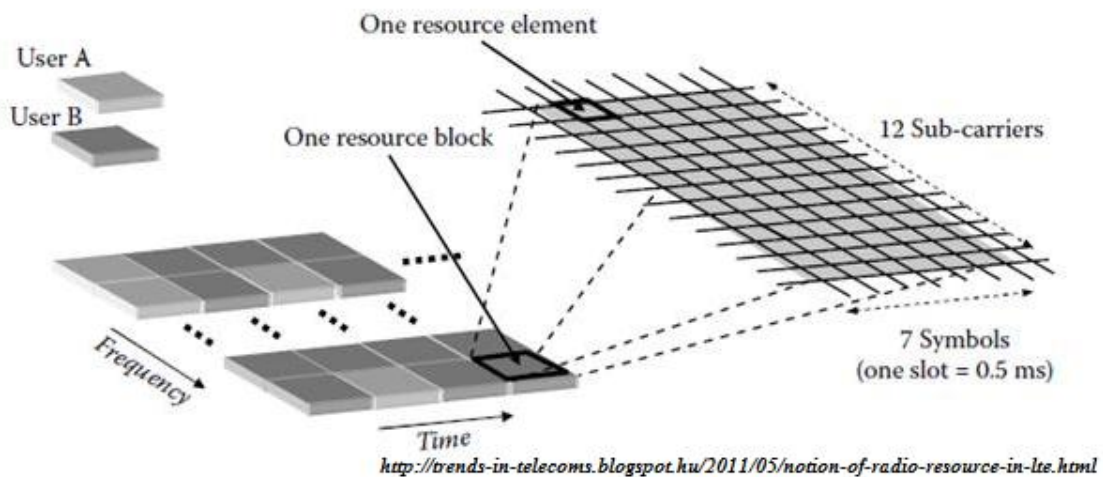
4.3.1 Az LTE keretszerkezete

Egy LTE keret 10 ms hosszúságú és 10 db 1ms hosszúságú alkeretre van felosztva. Az alkereteket is tovább osztják 2db 0.5 ms hosszúságú időrésre, azonban az ütemező mind downlink, mind uplink irányban 1 ms-onként hoz döntéseket, tehát azt lehet mondani, hogy valójában alkereteket oszt ki a felhasználók között az adatküldés, vagy az adatfogadás biztosítása érdekében [9]. Egy LTE keret a 4.4 ábrán látható.



4.4 ábra – LTE keretszerkezet

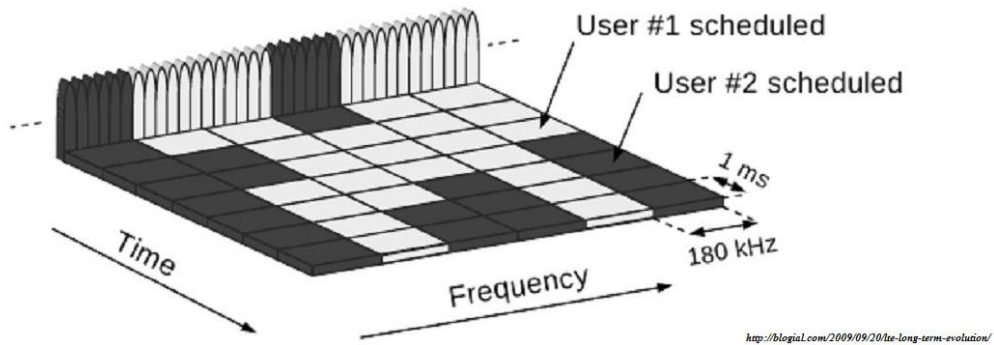
A PRB (Physical Resource Block), vagyis az erőforrásblokk, frekvenciában is megosztott az OFDMA technológiának köszönhetően. Így összeségében azt lehet mondani, hogy egy erőforrásblokk $180 \text{ kHz} * 0.5 \text{ ms}$ területű téglalap. Egyetlen erőforrásblokk tehát 12 alvivőt és egy időrészt tartalmaz [9]. Egy időrésben 7, esetleg 6 szimbólum vihető át, az alkalmazott OFDM paramétereitől függően [9]. Az ütemező minden időpillanatban erőforrásblokkokat oszt ki a felhasználók között, ez tehát a legkisebb egység amit egy felhasználó kaphat. Az ütemezés kérdése tehát abban áll, hogy melyik PRB-t, mikor, melyik felhasználó kapja meg. Egy erőforrásblokk felépítését mutatja a 4.5 ábra.



4.5 ábra – Egy erőforrásblokk

4.3.2 Downlink ütemezés

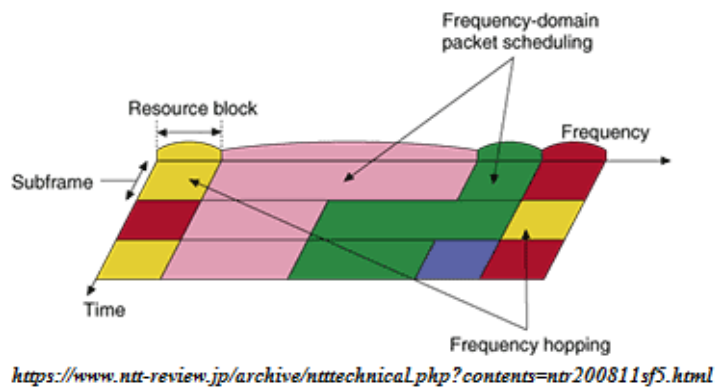
Különbséget kell tenni a downlink és az uplink ütemezés között. Downlink esetben első lépésben a bázisállomás referenciajelet küld a terminálnak, amely ennek segítségével meghatározza a csatornája pillanatnyi minőségét. Ezt a referenciajelen mért csillapítás alapján tudja kiszámolni. A csatorna állapotára jellemző értéket visszaküldi a bázisállomásnak, amely ezek után osztja ki az erőforrásblokkokat [9]. Egy lehetséges erőforrásblokk-allokációt mutat a 4.6 ábra.



4.6 ábra – Egy lehetséges erőforrásblokk-allokáció downlink esetben

4.3.3 Uplink ütemezés

Uplink esetben tulajdonképpen ugyanebből a idő-frekvencia ábrából lehet kiindulni, viszont a downlinkhez képest az uplink egyvívós jellegű, éppen ezért feltöltés esetén csak szomszédos erőforrásblokkok rendelkezhetőek a felhasználóhoz [9]. A 4.7 ábrán látható egy uplink esetben megvalósuló erőforrásblokk-allokáció.



4.7 ábra – Egy lehetséges erőforrásblokk-allokáció uplink esetben

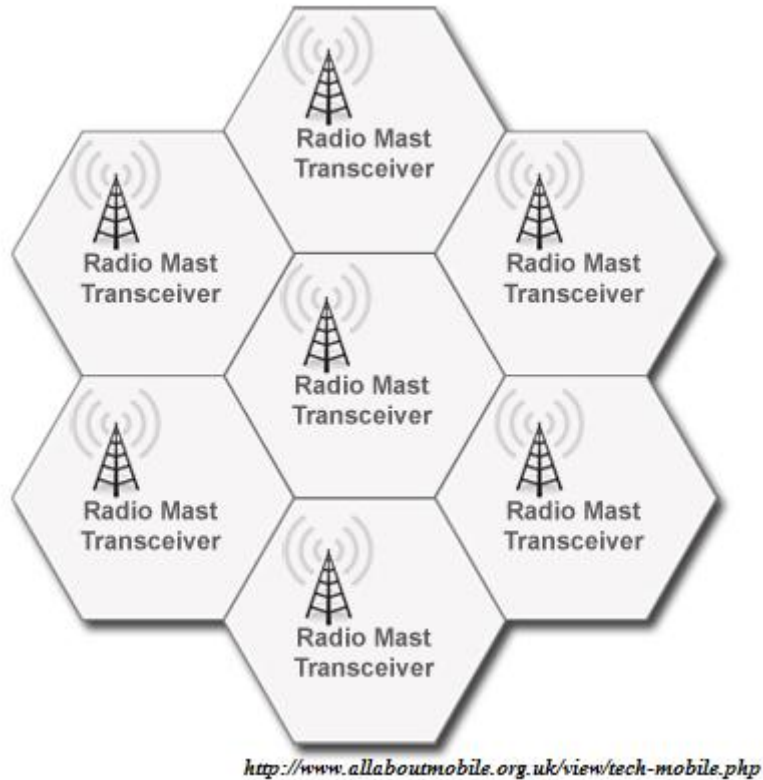
5. Cellás mobil rendszerek

Az előző fejezetekben láthattuk, hogy hogyan is működnek az általunk mai napig használt mobilhálózatok. Bemutattam a különböző szabványok hálózati architektúráját, beszéltem különböző közeg-hozzáférési technológiákról és modulációs eljárásokról, amelyek segítségével egy bázisállomás képes kiszolgálni a felhasználókat. De mi is valójában egy bázisállomás? Hogyan értelmezhetjük ezt a fogalmat egy országos hálózat viszonylatában? Ebben a fejezetben ezekre a kérdésekre keresem a választ.

Amikor az ember mobilhálózatokat tanulmányoz, gyakran találkozik a cellás rendszer kifejezéssel. Ez azt jelenti, hogy a lefedni kívánt területet nem egyetlen antennával fedik le, hanem ún. cellákra osztják. Így lényegesen több antennára van szükség, de több szempontból is kedvező tulajdonságú lesz az így kapott hálózat.

Először is meg kell említeni az adóteljesítmény csökkenését. A mobil távközlésben a frekvenciától függően meg van adva, hogy mekkora lehet az adóteljesítmény. Ennek legfőbb oka az, hogy az elektromágneses sugárzás az emberi szervezetre káros lehet, amennyiben túl nagy dózisban éri a szerveket. Mivel pedig a teljesítmény négyzetesen függ a térerősségtől, ezért meg kell adni egy küszöböt, amely teljesítménynél az elektromos térerősség még ártalmatlan.

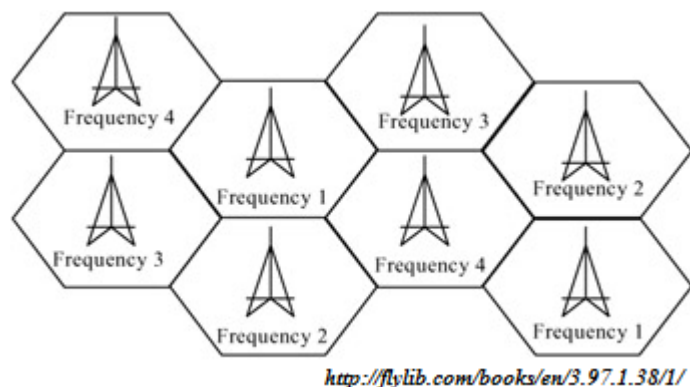
Másodszor kapacitásbeli megfontolásokat érdemes vizsgálni. Minél nagyobb területet fedünk le egy antennával, annál több felhasználó fog az adott antennára kapcsolódva kommunikálni, így az a sáv szélesség, amit az operátor használ, sokfelé oszlik el. Lényegesen hatékonyabb megoldás, ha kisebb méretű cellákra osztjuk a lefedni kívánt területet, így csökken az egy cellában lévő felhasználók száma is, tehát egy felhasználó nagyobb sáv szélességet kaphat. Egy cellás rendszert ábrázol az 5.1 ábra.



5.1 ábra – Egy méhsejt cellás rendszer

Bázisállomásnak a cellákat kiszolgáló adókat nevezzük. A felhasználók ezekhez csatlakoznak, ezeken keresztül kapnak, illetve küldenek adatot a maghálózat felé.

Mivel a frekvencia egy véges erőforrás, ezért az országok megfelelő hatóságai csak véges méretű sávokat osztanak ki a mobil szolgáltatók között. Emiatt a frekvenciát nem lehet pazarolni, nem lehet megoldani azt, hogy minden – az adott terület kiszolgálásában részt vevő – cella különböző sávokat használjon. Ezt a problémát hivatott áthidalni a frekvencia újrafelhasználás, amelynek lényege, hogy az egymástól megfelelő távolságra lévő – nem szomszédos – cellák használhatják ugyanazokat a frekvenciasávokat, hiszen a köztük lévő távolság miatt nem fognak a vevőegységekben interferenciát okozni [11]. A frekvencia újrafelhasználás elve az 5.2 ábrán látható.



5.2 ábra – A frekvencia újrafelhasználás

A cellás hálózatok csoportosításának legegyszerűbb és egyben legelterjedtebb módja, a lefedettség mérete alapján történő csoportosítás. Ezek alapján négy csoportot különböztetünk meg: a makro-, a mikro-, a piko- és a femtocellás rendszereket.

5.1 Makrocellás rendszerek

A legnagyobb lefedettséget biztosító cella a makrocella [12]. Attól függően, hogy az adott területen milyenek a rádiós terjedést befolyásoló paraméterek, a makrocella, egy 2-35 km sugarú cella. Ahhoz azonban, hogy ilyen nagy távolságra is eljusson a jel viszonylag nagy adóteljesítményre van szükség. Egy mobil készülék által vett jel teljesítménye kb. (-80)-(-90) dBm. Szabadtéri terjedést feltételezve, az (5.1) alapján tudjuk kiszámolni a szükséges adóoldali teljesítményt.

$$P_A = P_V \left(\frac{4\pi df}{c} \right)^2 \quad (5.1)$$

Az egyenletben P_A az adó teljesítménye, P_V a vett jel teljesítménye, d az adó és a vevő távolsága, f a használt vivőfrekvencia és c a fénysebesség. Ha kiszámoljuk P_A értékét egy Magyarországon használt vivőfrekvenciára (1800 MHz), akkor néhányszor 10 W-t kapunk. Azt látjuk tehát, hogy egy makrocellás antenna adóteljesítménye néhányszor 10 W kell hogy legyen.

Felmerül a kérdés, hogy a teljesítmény meghatározásakor mennyire jogosan használtuk a szabadtéri jelterjedési modellt. Mivel a makrocellák nagy sugarú terület lefedésére képesek, ezért általában olyan helyekre telepítik őket, ahol a jelet csillapító

tényezők nem meghatározóak (pl.: autópályák mellé, nagyméretű sík terepekre, stb.). Visszafelé gondolkozva, ezeken a területeken éppen azért lehet makrocellákat alkalmazni, mert a jel keveset csillapodik terjedés közben. Ebből a megfontolásból látszik, hogy nem követünk el nagy hibát azzal, hogy makrocellák esetén szabdtéri jelterjedést feltételezünk. Olyan területekre, ahol a szabdtéri terjedés nem alkalmazható, mert a jelet érő csillapító hatások felerősödnek (pl.: városok, sűrűn lakott területek, hegyes-dombos vidékek, stb.) már nem makro-, hanem mikrocellákat alkalmaznak.

5.2 Mikrocellás rendszerek

Kisebb lefedettségigényű területek ellátásához mikrocellákat alkalmaznak [12]. A makrocellákhoz képest a mikrocellák kisebb, jellemzően néhány kilométer sugarú cellát képesek lefedni. Természetesen a megfelelő minőségű kiszolgálás érdekében egy mikrocella kevesebb felhasználó ellátására képes, mint egy makrocella. Ezzel szemben a kisebb lefedettség két pozitív járulékot is jelenthet:

- Hasonló rádiós paraméterekkel jellemezhető makrocellával szemben a mikrocella kisebb teljesítménnyel kell, hogy adjon. Ennek oka az, hogy kisebb távolságon kevésbé csillapodik a jel. Vagyis kisebb adóteljesítmény is elégséges jelszintet eredményez a vevőben, kisebb távolságokon.
- Egy makrocellához hasonló adóteljesítményű mikrocella lényegesen rosszabb rádiós paraméterek mellett is megfelelő jelszinteket képes az vevőig eljuttatni.

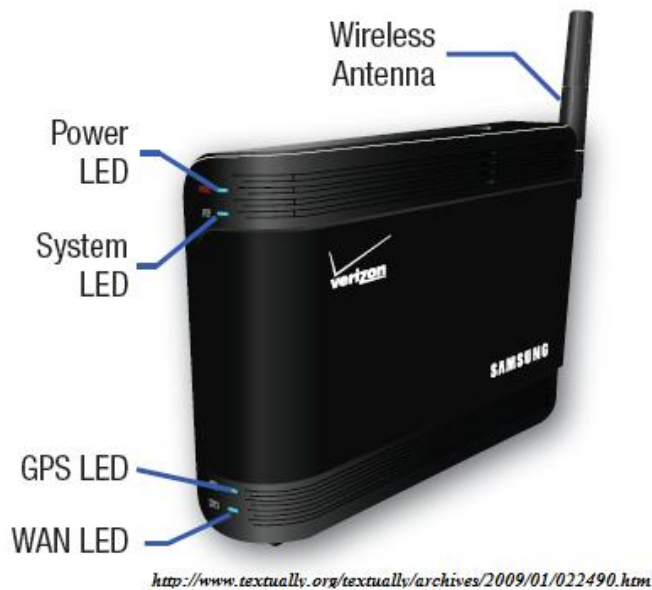
Az utóbbi következmény miatt alakult ki az, hogy sűrűn lakott városi környezetben – ahol a többutas terjedés, valamint az épületek csillapítása miatt sokkal nagyobb jeltorzulással kell kalkulálni – jellemzően kisebb cellaméreteket alkalmaznak, vagyis mikrocellák segítségével fedik le a területet.

5.3 Pikocellás rendszerek

Amennyiben még a mikorcelláknál is kisebb lefedettségre van szükség, pikocellák alkalmazására kerül sor [12]. A pikocellák jellemzően néhány száz méter sugarú lefedettséget képesek biztosítani és általában mobil hálózatok beltéri kiterjesztésére használják. A kisebb lefedettség ebben az esetben is kisebb adóteljesítménnyel jár, ami ebben az esetben már csupán néhány száz és néhány ezer mW között mozog. Ahogyan a mikorcellák esetében is, úgy itt is igaz az, hogy a cellaméret csökkenése magában hordozza a maximálisan kiszolgálható felhasználók számának csökkenését is. A pikocellák tulajdonképpen értelmezhetőek nagyobb terület lefedésére képes femtocellákként is.

5.4 Femtocellás rendszerek

A femtocellák – ahogy azt a „femto-” prefix is mutatja – nagyon kis lefedettség biztosítására alkalmas cellák [12]. Egy femtocella jellemzően egy szoba, jobb esetben egy háztartás ellátását tudja biztosítani és körülbelül 5, maximum 15 körüli felhasználó számára tud megfelelő minőségű átvitelt biztosítani (ezutóbbi inkább vállalati környezetben fordul elő). Ahogy azt az eddigiekben láttuk, a cellák adóteljesítménye is csökken a lefedettség csökkenésével. Az adóteljesítmény femtocellák esetében csupán néhány száz tíz mW, ami elsőre meglepőnek tűnhet, de ilyen kis lefedettség eléréséhez elegendő. Egy tipikus femtocella, melyet LTE esetben „Home eNodeB”-nek nevez a szabvány a 5.3 ábrán látható.



5.3 ábra – Egy femtocella (Verizon-Samsung)

Ahogy az ábrán is jól látszik, egy femtocella ránézésre nem sokban különbözik egy otthon használt 802.11 (Wi-Fi) routertól, ráadásul működésben is hasonlóak, hiszen mindkettő működéséhez szükség van már meglévő szélessávú kapcsolatra. Fontos különbség azonban az, hogy míg egy Wi-Fi router a 2.4 GHz-es ISM (Industrial, Scientific and Medical) sávban működik, addig a femtocella a szolgáltató dedikált sávját használja. Ez az jelenti, hogy Wi-Fi router alkalmazása esetén lényegesen nagyobb interferenciával kell számolni, mintha femtocellát használunk.

Fontos különbség mutatkozik még az adóteljesítményekben is. Mint már említettük, egy femtocella körülbelül néhányszor tíz mW teljesítménnyel ad, ezzel szemben egy Wi-Fi router adóteljesítménye a 100 mW-t is elérheti. Ez akár a femtocellák adóteljesítményének az ötszöröse is lehet (kisteljesítményű femtocella esetén). A XXI. században pedig egyre nagyobb figyelmet kap a „green computing”, így ezt a teljesítménykülönbséget nem lehet figyelmen kívül hagyni.

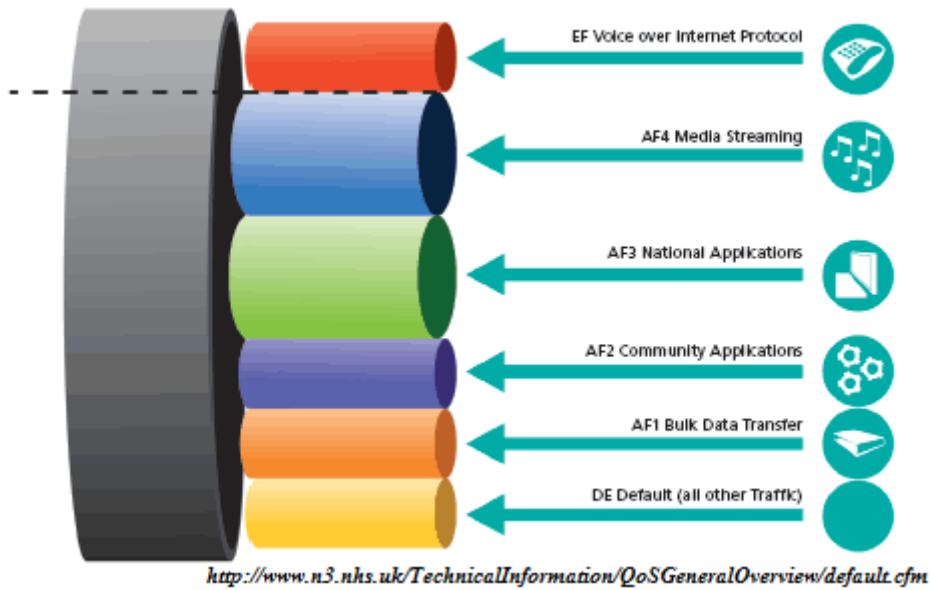
A femtocellák éppen a kis lefedettség, valamint a kevés felhasználó miatt nagyon hatékonyan, gyors le-, illetve feltöltési sebességeket tudnak produkálni. Nem tűnik tehát rossz ötletnek az, hogy kombináljuk a femtocellák, valamint a jelenleg leggyorsabb átvitelre képes telekommunikációs szabvány előnyeit és vizsgáljuk hogyan fog működni egy LTE alapú femtocella. Előtte azonban tekintsük át a garantált minőségű szolgáltatás nyújtásához szükséges feltételeket.

6. Quality of Service

Az eddig átgondoltak alapján nyugodtan kijelenthetjük, hogy egy komolyan kigondolt és nagyon jól kidolgozott infrastruktúrát építettünk ki annak érdekében, hogy a megnövekedett felhasználói igényeknek eleget tudjunk tenni. Nem vizsgáltuk azonban még azt, hogy milyen jellegű felhasználói igényekről is van szó. Az internet rohamos fejlődésének következtében mára olyan összetett forgalom alakult ki, amit már nem lehet olyan egyszerűen kezelni, mint tettük azt az internet kialakulásakor. Ahogy azt korábban említettem, az újgenerációs hálózatok nagyrészt IP alapúak, ezáltal egy felhasználó által generált összes forgalmat tulajdonképpen az internet segítségével továbbítunk. Ez VoIP (Voice Over IP) különböző minőségű hang-, illetve videó-adatfolyamok átvitele, vagy egyszerű adatátvitel (pl.: e-mail). Ahhoz, hogy ezeket a forgalmakat megfelelő minőséggel tudjuk továbbítani, át kell gondolnunk a jellemzőiket [13]. Tulajdonképpen erre épít a QoS technológia, amely garantálni képes a különböző típusú adatfolyamok megfelelő minőségű átvitelét. A QoS négy alapelven nyugszik, amelyek a következők:

1. Csomagok osztályozása
2. Forgalomfelügyelet
3. Hatékony erőforráskihasználás
4. Hívásengedélyezés

Minden forgalomtípus különböző jellemzőkkel rendelkezik. A lassú adatátviteli sebesség, a bithibák, a csomagvesztés, vagy éppen a késleltetés mind negatívan befolyásolják a felhasználói élményt, ezért ezek alapján a jellemzők alapján csoportosítani kell a különböző forgalmakból származó csomagokat [13]. Ezek után meg lehet határozni valamilyen prioritást, amely alapján a forgalmakat érdemes kiszolgálni. Amennyiben a szétválasztott forgalmakat megfelelő módon ütemezzük, megvalósítható, hogy minden felhasználó megfelelő minőségű kapcsolatot kapjon. Egy ilyen jellegű forgalomszétválasztásra mutat példát a 6.1 ábra.



6.1 ábra – Forgalmosztválasztás

Az ábrán egy 6 rétegű QoS megvalósítás látszik, vagyis a forgalmat 6 különböző osztályba sorolják. Ez egy fajta megvalósítás, viszont az, hogy milyen környezetben milyen QoS-re van szükség nagyban függ a környezet típusától. Például egy otthoni felhasználó igényei nem egyeznek meg egy üzleti felhasználó igényeivel, ezért a különböző forgalmosztályok meghatározásakor figyelembe kell venni a felhasználási környezetet.

7. A szimulációs környezet

Az előző részekben bemutatam azokat a szabványokat és technológiákat, amelyek ismerete elengedhetetlen a szimulátor működésének megértéséhez. Ismertettem az LTE hálózatok főbb paramétereit és keretszerkeztét, betekintést nyújtottam a femtocellák működésébe és vázoltam a garantált minőségű szolgáltatás nyújtásának főbb szempontjait. Ebben a fejeztben azonban már a szimulátorról lesz szó. Bemutatom a szimulátor felépítését, valamint ismertetem a szimulátor működését.

7.1 A szimulátor felépítése

A szimulátor megalkotása során figyelembe kellett venni, hogy nem elhanyagolható méretű adatstruktúrákkal kell dolgozni, azért különösen fontos szempont volt a gyors működés. Elsősorban emiatt, a szimulátort C++ nyelven fejlesztettem. Felépítését tekintve a szimulátor alapvetően két rétegből áll. Az első réteg a felhasználói adatok bekéréséért és azok tárolásáért, a második réteg pedig az ütemezési algoritmusok lefuttatásáért és az eredmények kiírásáért felelős.

A program számtalan paraméterrel dolgozik. Ezek a következők:

- a kiszolgáló cellában lévő felhasználók száma,
- a cella sugara,
- az alkalmazott sávszélesség,
- a forgalommodellezéshez szükséges paraméterek,
- valamint bizonyos ütemezések paraméterei.

A szimulátor többféle forgalomtípus figyelembevételével építi fel a szimulációt. A forgalomtípusok kiválasztásánál figyelembe vettük a jelenlegi internetes trendeket, illetve a telekommunikációban várhatóan bekövetkező változásokat, így az alábbi forgalomtípusok adódtak:

- magas bitsebességű videó (2,25 Mbit/s),
- alacsony bitsebességű videó (768 kbit/s),
- VoIP (8 kbit/s),
- magas bitsebességű audió (320 kbit/s),
- alacsony bitsebességű audió (128 kbit/s),
- általános adatforgalom.

A különböző típusok mellett az általunk beállított kodekekre kiszámított sávszélességigények láthatóak. Ezeket a felhasználó a saját igényei szerint megváltoztathatja, erre a szimulátor lehetőséget ad.

Mint az később látni fogjuk, bizonyos ütemezések esetén szükséges, hogy a forgalomtípusok valamilyen hierarchiába rendeződjenek. Ez a hierarchia megegyezik a fenti felsorolással, vagyis a magas bitsebességű videó a legmagasabb prioritású, míg az általános adatforgalom a legalacsonyabb. A forgalmak generálásakor „worst case” esetet feltételezünk. Ez azt jelenti, hogy minden felhasználó, minden időpillanatban, mindegyik típusú forgalmat használja, ezáltal a hálózat terhelése maximális. Mivel a szimulátor egy 1 perces időtartamot vizsgál, ezért a forgalmakat úgy generáltuk, hogy a szimuláció végéig a kezdeti érték ne változzon. Ez azt jelenti, hogy mindenki használni fogja az összes forgalomtípust a szimuláció teljes ideje alatt.

7.2 A szimulátor működése

A szimulátor elindítása után adatbekérésre kerül sor. A felhasználó megadja azokat a paramétereket, amelyeket a program a működése során használni fog. Miután ez megtörtént, megkezdődik a forgalomgenerálás. Mivel az ütemező időrezenként dönt a kiosztható erőforrásblokkok sorsáról, ezért szükséges, hogy mind az egy erőforrásblokkban maximálisan továbbítható bitek számát, mind a különböző forgalmak egy időrésre redukált erőforrásigényét ismerjük. Az egy időrésben maximálisan továbbítható bitek számát (B) a (7.1) segítségével határozhatjuk meg:

$$B = 12 (\log_2 M) S. \quad (7.1)$$

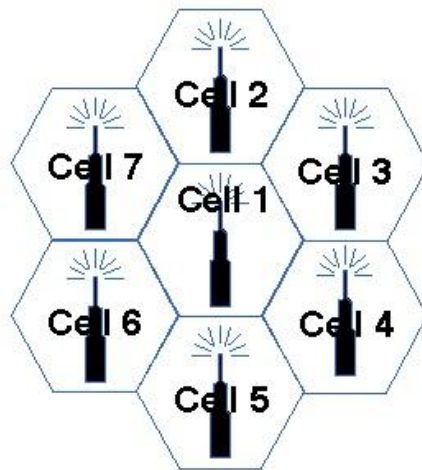
A képletben M az alkalmazott moduláció állapotainak a száma, S pedig az egy erőforrásblokkban átvihető szimbólumok száma. Ez az alkalmazott OFDM paramétereitől függően 6 vagy 7. Én, a szimulációban 64-QAM modulációt használok, és egy időrésben 7 szimbólumot viszek át, így B -re 504 bit/erőforrásblokk adódott.

A forgalmak időrezenkénti erőforrásigényének (F) meghatározásában a (7.2) lehet segítségünkre:

$$F \left[\frac{\text{bits}}{\text{timeslot}} \right] = \frac{F \left[\frac{\text{bits}}{\text{seconds}} \right]}{2000}. \quad (7.2)$$

A fenti képletben kihasználtuk azt, hogy egy másodpercben 2000 db 0,5 ms-os időrés található.

Így megkaptuk az igényelt és a kiosztható erőforrások számát is, ugyanakkor a hatékony ütemezés megvalósításához szükség van rá, hogy ismerjük a különböző felhasználók és a bázisállomás közötti csatorna minőségét. Ezt a vevőben meghatározott jel-zaj viszony értékével jellemezhetjük. A vizsgált cellánk elhelyekedését az 7.1 ábra szemlélteti.



7.1 ábra – A vizsgált cellák elhelyezkedése

Az ábrán Cell 1 az általunk vizsgált, kiszolgáló cella, Cell 2 – Cell 7 pedig a vele szomszédos interferenciát okozó cellák. Ezek teljesítményjellemzői megegyeznek az általunk vizsgált celláéval. Így meghatározhatjuk a Cell 1 által a vevőben létrehozott jelteljesítményt, illetve a többi cella által létrehozott jelteljesítményt. A kiszolgáló cella adóteljesítményét (50 mW) a szabadtéri csillapítással csillapítva, megkaptam a hasznos jelet. Az interferenciát okozó antennák esetében a teljesítmény ugyanannyi, viszont a csillapítás ebben az esetben logaritmikus csillapítás. Emiatt az interferenciát okozó jelek jobban csillapodnak, ami reálisnak mondható, hiszen femtocellák esetében joggal élhetünk azzal a feltevéssel, hogy a két cella között (kiszolgáló és interferáló) valamilyen fal található. A fehér Gauss-zajtól ebben a szimulációban eltekintettem, hiszen általánosan elmondható az, hogy mobil rendszerekben a Gauss-zaj elhanyagolható a saját rendszerből származó interferencia mellett. Emiatt valójában nem jel-zaj viszonyt, hanem jel-interferencia viszonyt kapunk. A jel-interferencia viszony meghatározása után meghatároztam egy BER értéket (Bit Error Rate), amelyet

beleépítettem a sebesség meghatározásába. Ezt úgy tettem meg, hogy az átküldött bitekből levontam annyit, amennyi hibásan érkezett meg a BER alapján és a sebességet az így kapott – hibátlanul átvitt – bitek számából határoztam meg. A BER meghatározását a (7.3) alapján végeztem. A képletben SNR a jel-interferencia viszonyt jelöli.

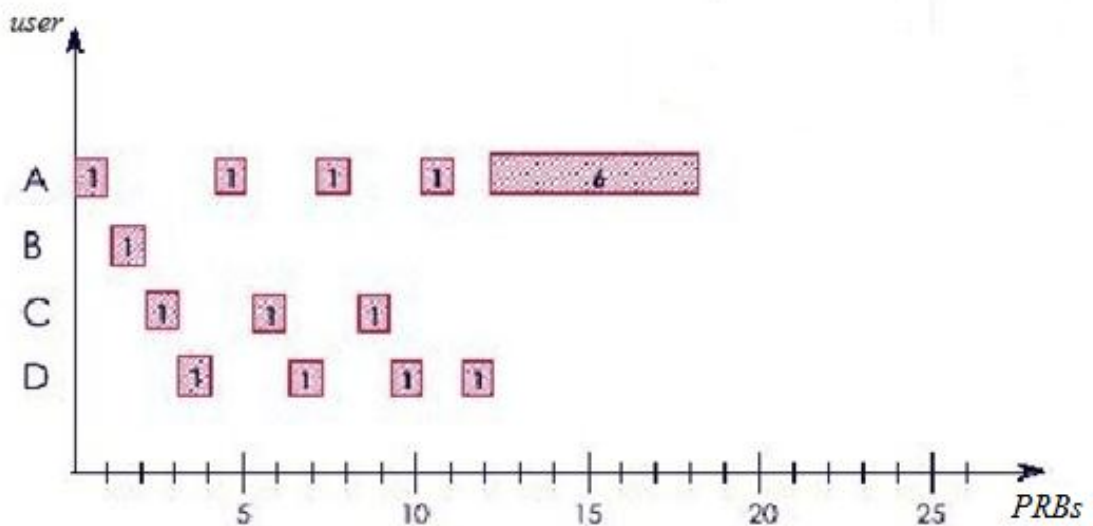
$$P_{b,64QAM} = \frac{1,75 \operatorname{erfc}\left(\sqrt{\frac{8}{126} SNR}\right) - 0,765625 \operatorname{erfc}^2\left(\sqrt{\frac{8}{126} SNR}\right)}{6} \quad (7.3)$$

8. A szimuláció kiértékelése

Az előző fejezetben áttekintettem a szimulátor felépítését és működését, ebben a fejezetben már a szimulátor által kapott eredményeket ismertetem. Négy ütemezési algoritmust fogok vizsgálni átvitel szempontjából. A szimulációt 10 méter sugarú cellát feltételezve, 15 felhasználóra végeztem el. Ez a beállítás – az adatgeneráláshoz hasonlóan – meglehetősen túlterhelt cellát feltételez, ezzel is segítve a „worst case” elemzést.

8.1 Round Robin algoritmus

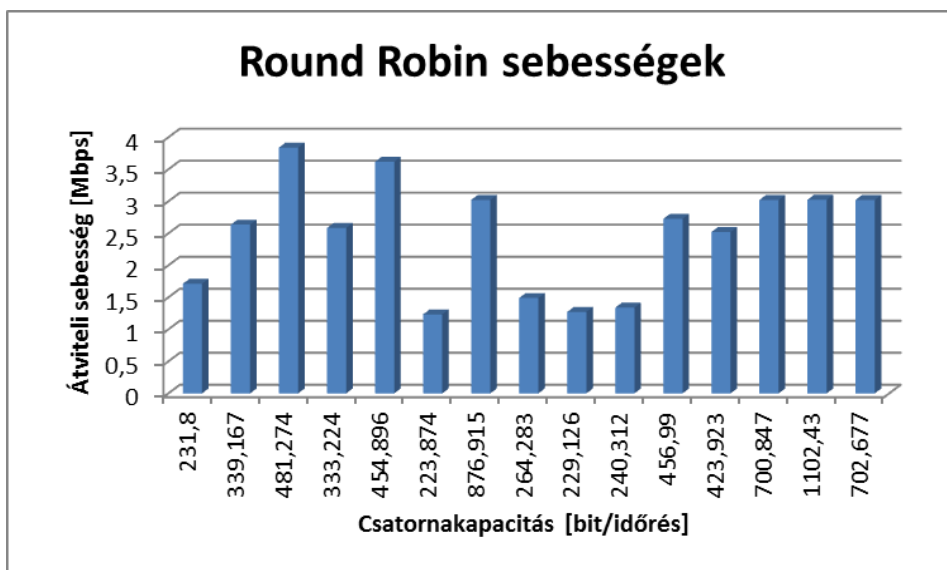
A Round Robin algoritmus talán a legegyszerűbb létező döntési modell. Az a lényege, hogy az ütemező minden döntési pillanatban más felhasználó számára biztosít erőforrást. Ennek köszönhetően senki nem fog prioritást élvezni, viszont – mivel a csatorna minőségétől függetlenül osztjuk ki az erőforrásokat – lassabb lesz az átvitel. A RR működését a 8.1 ábra szemlélteti.



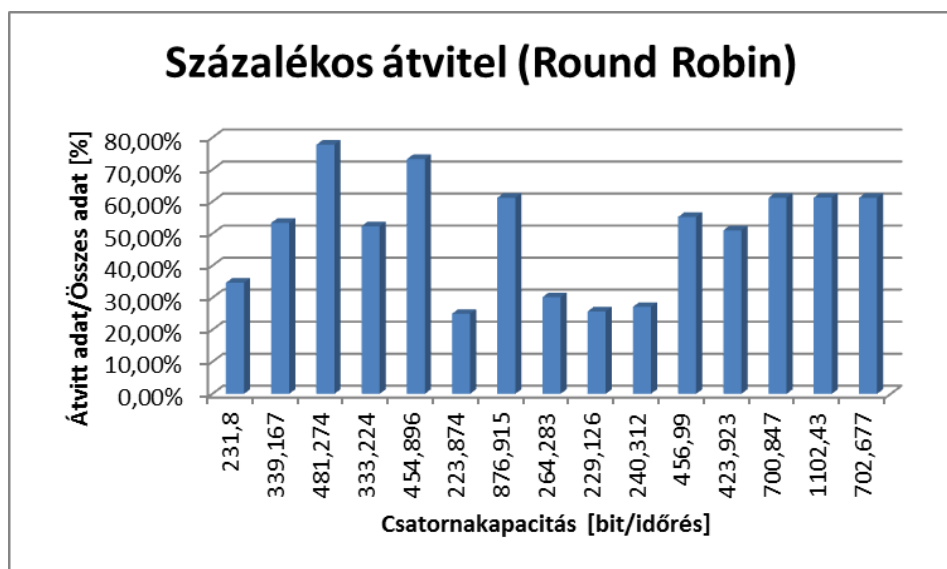
8.1 ábra - Round Robin algoritmus

A fenti ábrán jól látható, hogy minden döntési pillanatban más kap erőforrást, egészen addig, amíg szüksége van rá. Ha már minden adata továbbításra került, akkor nem kap több erőforrást.

A szimulációban a felhasználók között történik az ütemezés. Minden egyes felhasználóra meghatároztam, hogy összesen mennyi forgalmat generáltak (a forgalmak típusától függetlenül), és ezután indítottam az algoritmust. Minden felhasználó egy erőforrásblokkot kapott egy ütemezési ciklusban. A kapott eredményeket a 8.2 és a 8.3 ábrák szemléltetik.



8.2 ábra – Átviteli sebességek Round Robin esetén



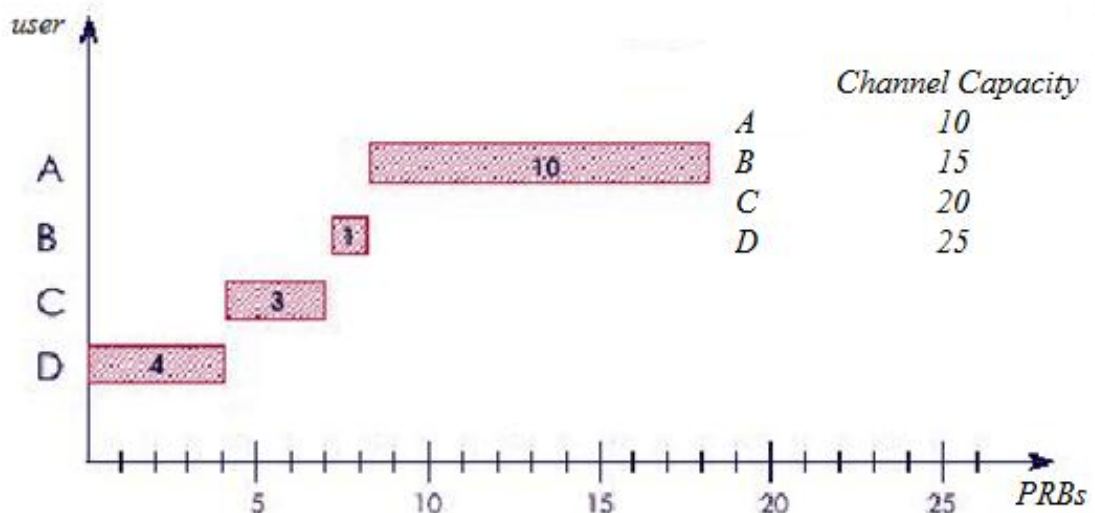
8.3 ábra – Százalékos átvitel Round Robin esetén

Az ábrákon a vízszintes tengely mindkét esetben a felhasználók csatornkapacitását jelöli bitben (időrésenként). A 8.2 ábrán a függőleges tengely az átviteli sebességeket jelöli Mbps-ban, míg a 8.3 ábrán az összes átvitt adat és az összes átvinni kívánt adat hányadosa látszik százalékban.

Az ábrákon jól lehet látni a Round Robin természetét. Minden felhasználó át tud vinni valamennyi adatot, ez azonban az átviteli sebesség rovására megy. Azt is jól lehet látni, hogy azok a felhasználók, akiknek rosszabb a csatornkapacitása, kevesebb adatot tudnak átvinni. Ez is érthető, hiszen az algoritmus azt garantálja, hogy minden felhasználó nagyjából azonos mennyiségű erőforrásblokkot kap. Azt, hogy ezeket ki milyen százalékban tudja kihasználni, nem tudja garantálni a rendszer.

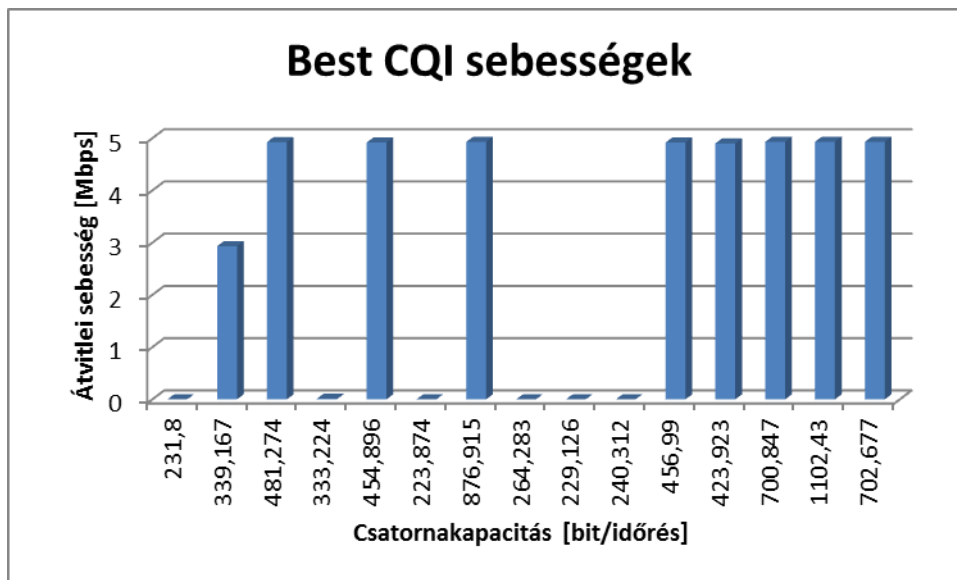
8.2 Best CQI algoritmus

A második vizsgált algoritmus a Best CQI. Ez az algoritmus gyakorlatilag a RR teljes ellentéte. A felhasználók prioritást élveznek az alapján, hogy milyen minőségű csatornával rendelkeznek. Értelemszerűen, minnél jobb a felhasználó CQI értéke, annál hamarabb fogják kiszolgálni. Amíg RR esetben minden felhasználó egy erőforrásblokkot kap, addig a Best CQI algoritmus értelmében annyit, amennyire szüksége van. Az algoritmus működését a 8.4 ábra szemlélteti.

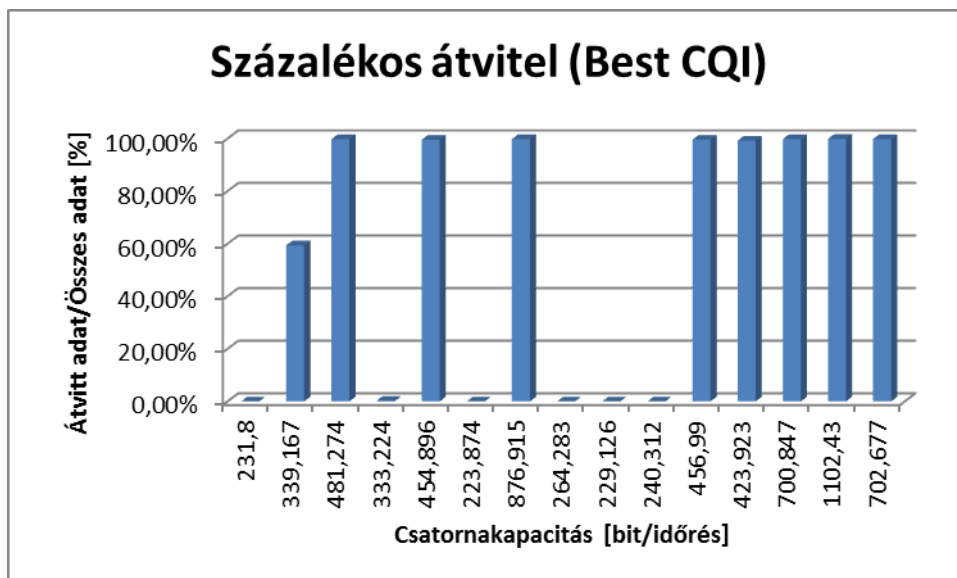


8.4 ábra – Best CQI algoritmus

Best CQI algoritmus abban az esetben akár hasznos is lehet, hogyha szándékosan korlátozni szeretnénk azokat a felhasználókat, akiknek gyenge minőségű a csatornája. Ez az algoritmusra vonatkozó szimulációs eredményekből is látszik, melyeket a 8.5 és 8.6 ábrák demonstrálnak.



8.5 ábra – Átviteli sebességek Best CQI esetén

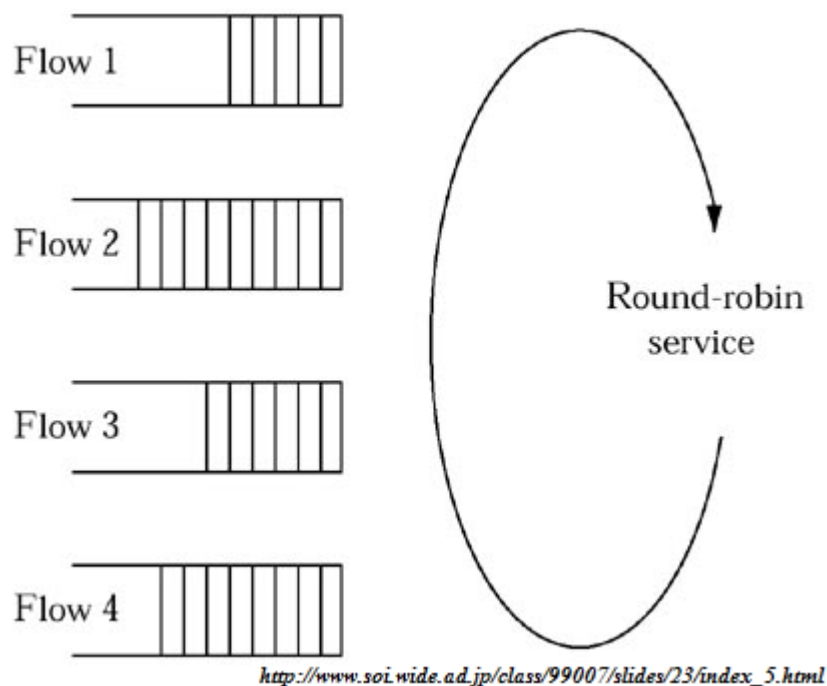


8.6 ábra – Százalékos átvitel Best CQI esetén

Az ábrákról ebben az esetben is leolvashatjuk az eredményeket. Látható, hogy ameddig van szabad időrés, azt mindig az a felhasználó kapja, aki a legjobb csatornával rendelkezik és van küldendő csomagja. Emiatt a jó csatornával rendelkező felhasználók közel 100%-ban át tudják vinni a forgalmukat, míg a gyenge csatornával rendelkezők egyáltalán nem kapnak erőforrásblokkot. Emiatt viszont a jó csatornával rendelkezők átviteli sebessége magas lesz.

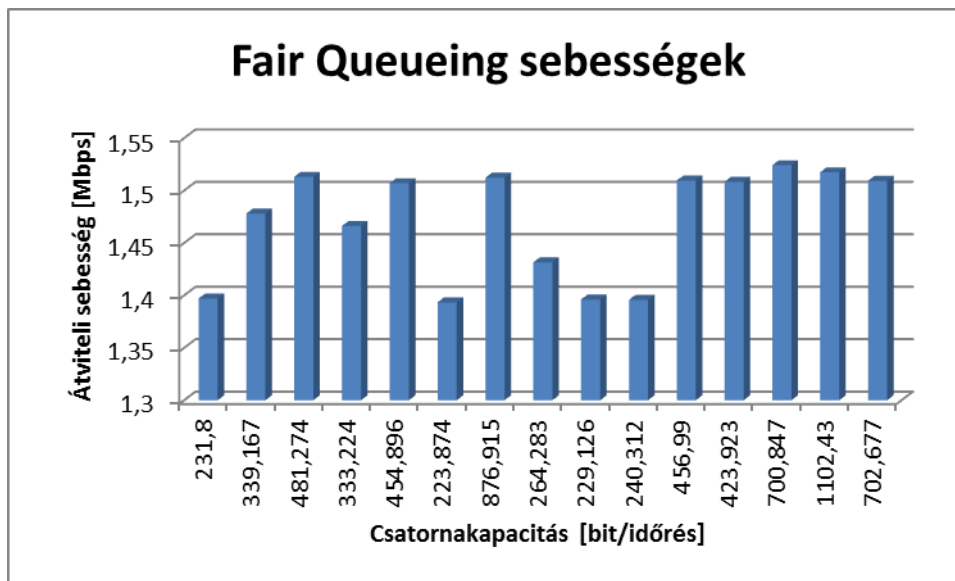
8.3 Fair Queueing algoritmus

Ahogy azt az előző fejezetben említettem, néhány algoritmus működéséhez elengedhetetlen, hogy különválasszuk a különböző típusú forgalmakat. Nézzük most meg az első ilyen algoritmusunkat, ami a Fair Queueing. A Fair Queueing algoritmus lényege, hogy szétválasztjuk a különböző forgalmakat és ezek között a forgalomosztályok között valósítunk meg Round Robin ütemezést. A működést a 8.7 ábra szemlélteti.

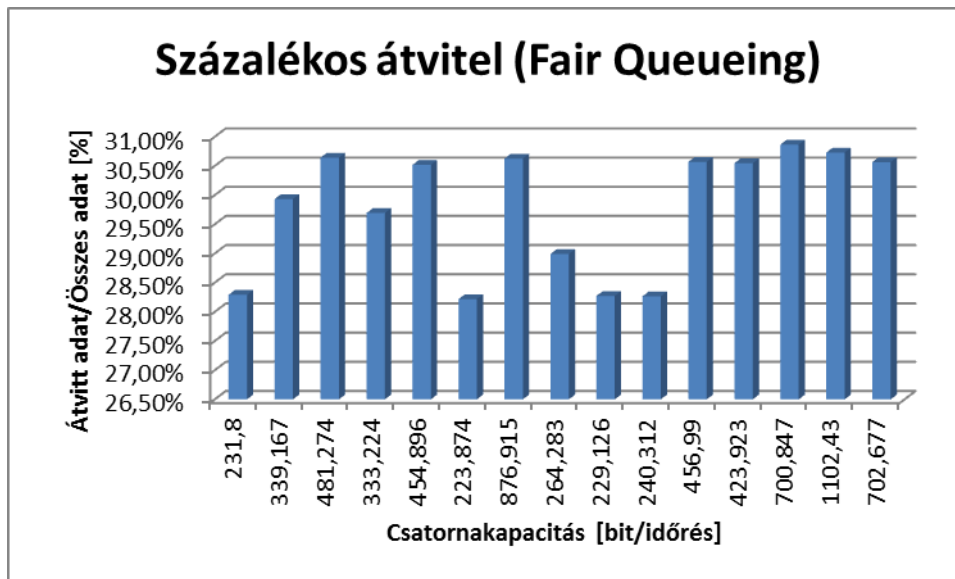


8.7 ábra – Fair Queueing algoritmus

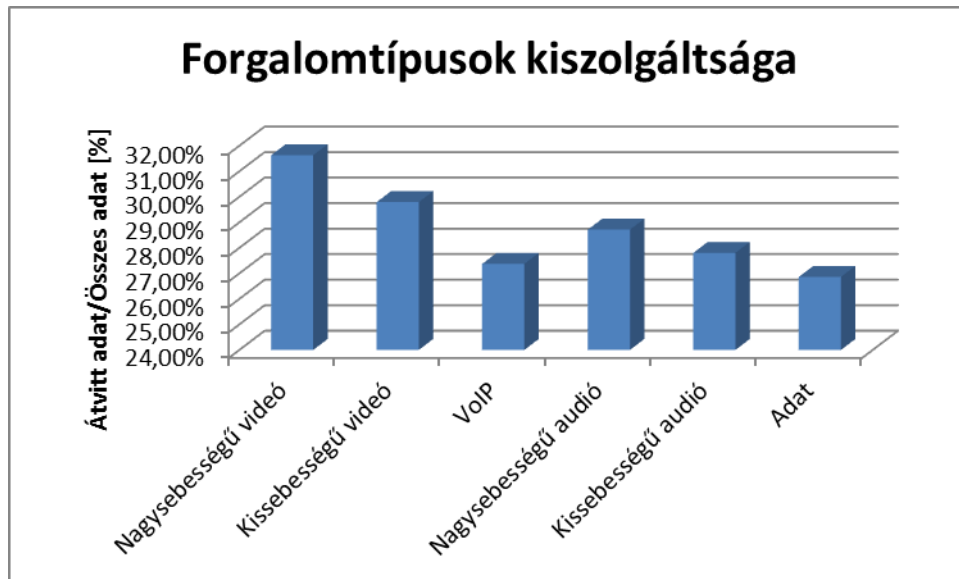
A mi esetünkben hat különböző forgalomsztály van, amik az előző fejezetben kifejtett forgalmakat azonosítják. A szimulációval kapott eredmények a 8.8, a 8.9 és a 8.10 ábrákon látszanak.



8.8 ábra – Átviteli sebességek Fair Queueing esetén



8.9 ábra – Százalékos átvitel Fair Queueing esetén

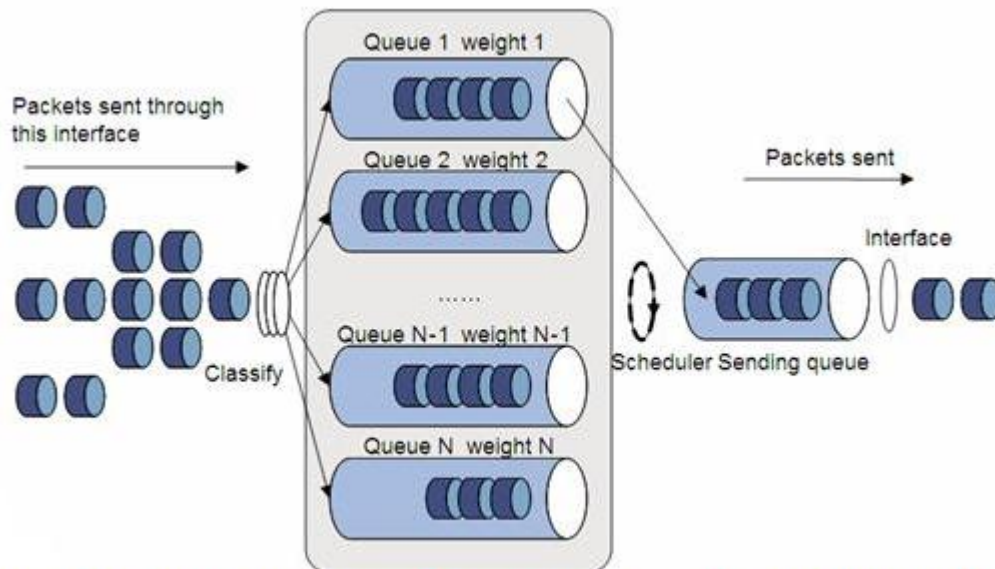


8.10 ábra – Forgalomtípusok kiszolgáltsága Fair Queueing esetén

Az ábráról jól leolvasható a Fair Queueing algoritmus működése. Látható, hogy a felhasználók az adataik közel azonos hányadát tudták átvinni, így az átviteli sebességeik közel megegyeznek. Látható az is, hogy a különböző forgalmak között is közel azonos a kiszolgálás mértéke, ami nem meglepő, hiszen a forgalomtípusok között Round Robin ütemezést végeztem. Van azonban egy dolog, ami a szemünkbe tűnhet. Az átviteli sebességek ugyan közel megegyeznek, de önmagukban nem magasak, legjobb esetben is 1.5 Mbps körül vannak. Felmerül tehát a kérdés, hogy rosszabb-e ez az algoritmus, mint a Round Robin volt? Nem, nem rosszabb. A kisebb átviteli sebességnek egyetlen oka van, mégpedig az, hogy a Round Robin ütemezésnél burst-ös forgalmat vizsgáltam. Ez alapján összeadtuk, az összes átvinni kívánt bitet, majd ezek közül egy erőforrásblokkban átvittem, amennyit lehet. Ebben az esetben viszont a forgalmakat külön kezelem. Így egy erőforrásblokkban csak egyféle forgalom lehet. Emiatt, sok erőforrásblokk gyakorlatilag alig kitöltve kerül továbbításra, ez pedig lassú forgalmat eredményez. Amennyiben Fair Queueingnél is alkalmaznám a burst-ös forgalmat, akkor lényegesen magasabb sebességeket érnék el. A szimuláció szempontjából azonban ez is megfelel, hiszen az algoritmus működésének jellege jól látható. Ez igaz a következő vizsgált algoritmusra is.

8.4 Weighted Fair Queueing algoritmus

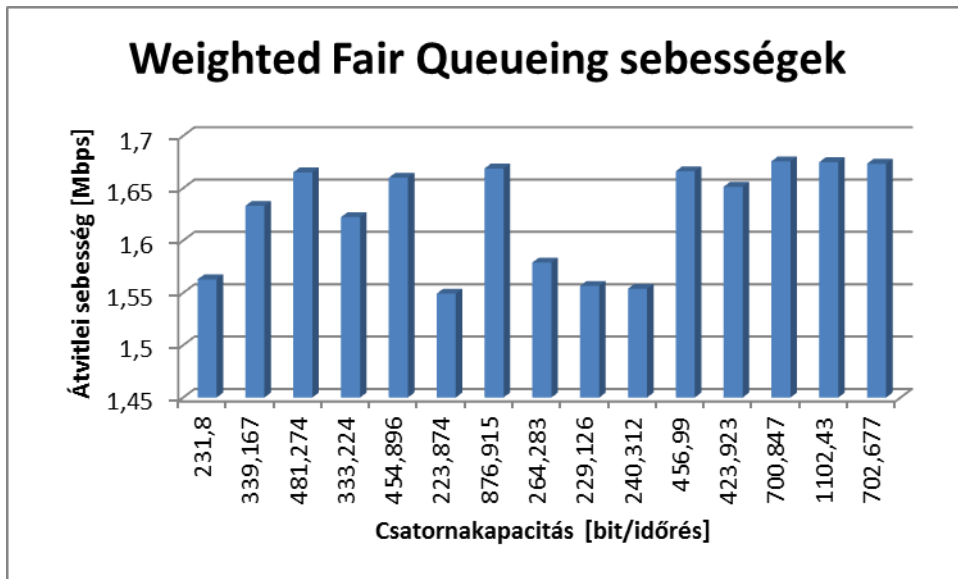
A következő és egyben utolsó ütemezési algoritmus a Weighted Fair Queueing. A vizsgált algoritmusok közül ez az egyetlen, amely QoS biztosítására képes. A működés ugyanúgy néz ki, mint a Fair Queueing esetén, azzal a különbséggel, hogy a forgalmakhoz súlyokat rendeltem. Ezek a súlyok meghatározták azt, hogy egy forgalomtípusból hány csomagot szolgálunk ki egy ütemezési periódusban. Az ütemezés működését a 8.11 ábra mutatja.



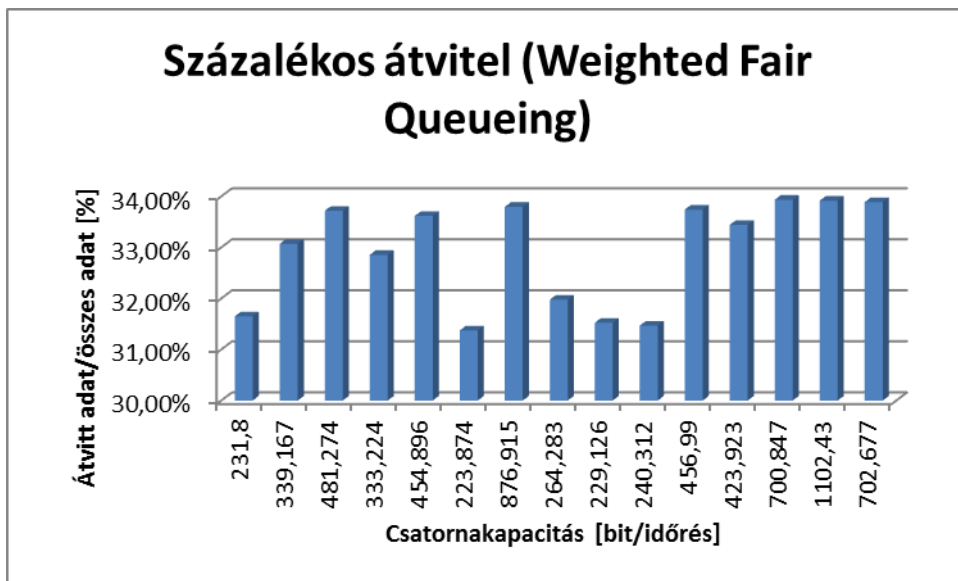
http://www.h3c.com/portal/Products_Solutions/Technology/QoS/Technology_Introduction/200701/195599_57_0.htm

8.11 ábra – Weighted Fair Queueing működés

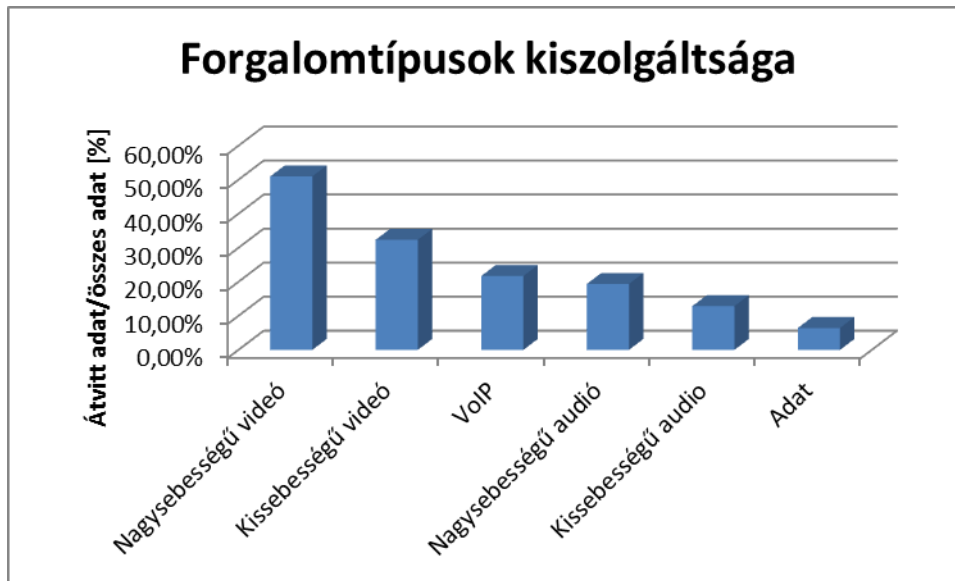
A szimulátorban be lehet állítani, hogy melyik forgalomtípusból mennyi kerüljön továbbításra, vagyis gyakorlatilag be lehet állítani a forgalmak súlyát. Én a szimulációt úgy végeztem el, hogy a legnagyobb prioritású forgalomból (nagysebességű videó) egy ciklusban 6 db csomag kerüljön kiszolgálásra, míg a legkisebből (Adat) 1 db. Közöttük a csomagok száma lineárisan csökken, vagyis 6-5-4-3-2-1 csomag kerül kiszolgálásra a különböző prioritású forgalmakból ütemezési periódusonként. Az így kapott eredményeket a 8.12, a 8.13 és a 8.14 ábrák mutatják.



8.12 ábra – Átviteli sebességek Weighted Fair Queueing esetén



8.13 ábra – Százalékos átvitel Weighted Fair Queueing esetén

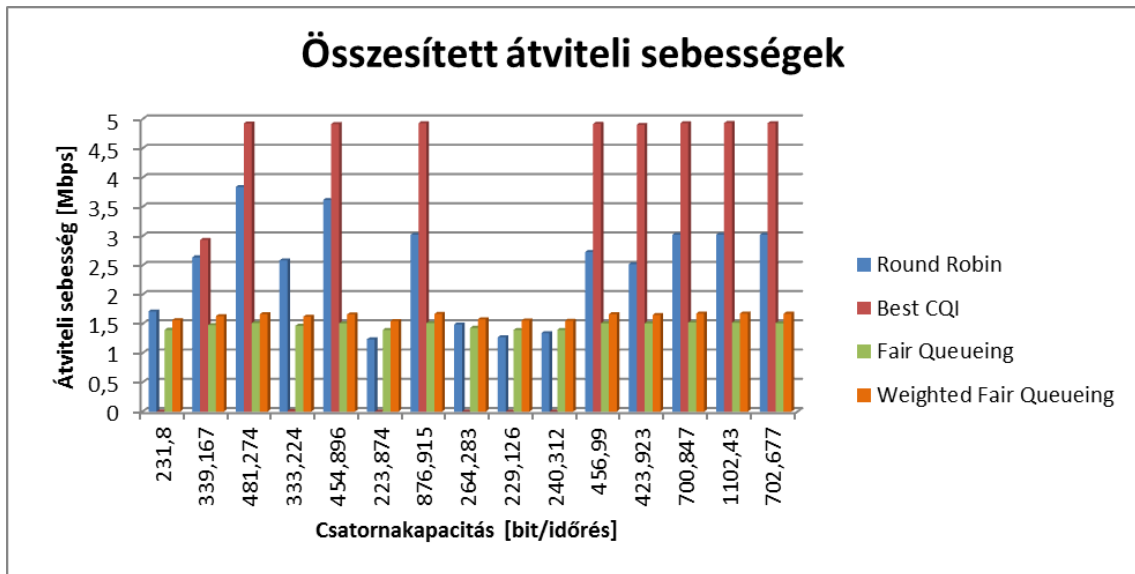


8.14 ábra – Forgalomtípusok kiszolgáltsága Weighted Fair Queueing esetén

Az átviteli sebességek a Fair Queueing esetén elmondottakhoz hasonlóan a burst-ös jelleg hiánya miatt alacsony. Ez burst alkalmazása esetén lényegesen jobb lenne. Ami azonban ennek az ütemezésnek a lényegét adja, az a 8.14 ábrán látszik. Látható, hogy a Fair Queueing-hoz képest megváltozik a forgalomtípusok kiszolgáltsága, köszönhetően annak, hogy a nagyobb prioritású forgalomból több csomagot viszünk át egy ütemezési ciklusban. Jól lehet látni, hogy a legnagyobb prioritású forgalom kiszolgáltsága majdnem a duplájára nőtt. Ezzel pedig sikerült QoS-t megvalósítani. Természetesen az 50%-os kiszolgáltság még korántsem jó, azonban ne felejtjük el, hogy a szimulált hálózat jelentősen túl van terhelve, így igazán jó átvitelt csak akkor tudunk elérni, ha a többi forgalomtípust „hanyagoljuk el”. Látható, hogy a súlyok beállításával ez megvalósítható. Innentől már a felhasználó igénye szerint lehet beállítani az ütemezést. Amennyiben a felhasználó csak egy forgalmat szeretne kapni, de azt igazán jó minőséggel, akkor a többi forgalom rovására ezt nyugodtan megteheti.

8.5 Az algoritmusok összefoglalása

Áttekintettük a vizsgálni kívánt algoritmusok működését, és a szimulátor segítségével láthattuk, hogy milyen teljesítménnyel képesek kiszolgálni a felhasználókat. A 8.15 ábrán összefoglaltam a különböző algoritmusok esetén kapott átviteli sebességeket.



8.15 ábra – Összesített átviteli sebességek

Az ábrán nagyon jól látható, hogy a legmagasabb sebességeket a Best CQI algoritmus alkalmazásával lehet elérni viszont ebben az esetben lényegesen kevesebb felhasználót szolgál ki a rendszer, mint az összes többi algoritmus esetén. Ez érthető is, hiszen a Best CQI algoritmus a csatornkapacitások alapján állít fel prioritásokat, így azok a felhasználók, akik gyenge minőségű csatornát használnak, egyáltalán nem kapnak erőforrást.

A RR átviteli sebességek esetén látható, hogy a felhasználók (közel) azonos mennyiségű erőforrásblokkot kapnak, ezért már minden felhasználó kiszolgálásra kerül. Észre lehet venni azonban, hogy ahol rosszabb a csatornkapacitás, ott rosszabb az átviteli sebesség is. Ennek oka, hogy rossz minőségű csatornán egy erőforrásblokk kevesebb bitet képes átvinni, mint egy jó minőségű csatornán, ezért hiába osztunk ki a felhasználók között nagyjából ugyanannyi erőforrásblokkot, a rossz csatornával rendelkező felhasználók ebben az esetben is rosszul járnak.

A két leghatékonyabb algoritmus a FQ és a WFQ, amelyek esetén megfigyelhetjük, hogy közel azonos sebességekkel szolgálják ki a felhasználókat. Emellett megfigyelhetjük a – már korábban említett – burts-ös jelleg hiányát is, ami miatt kisebb átviteli sebességeket tapasztalunk.

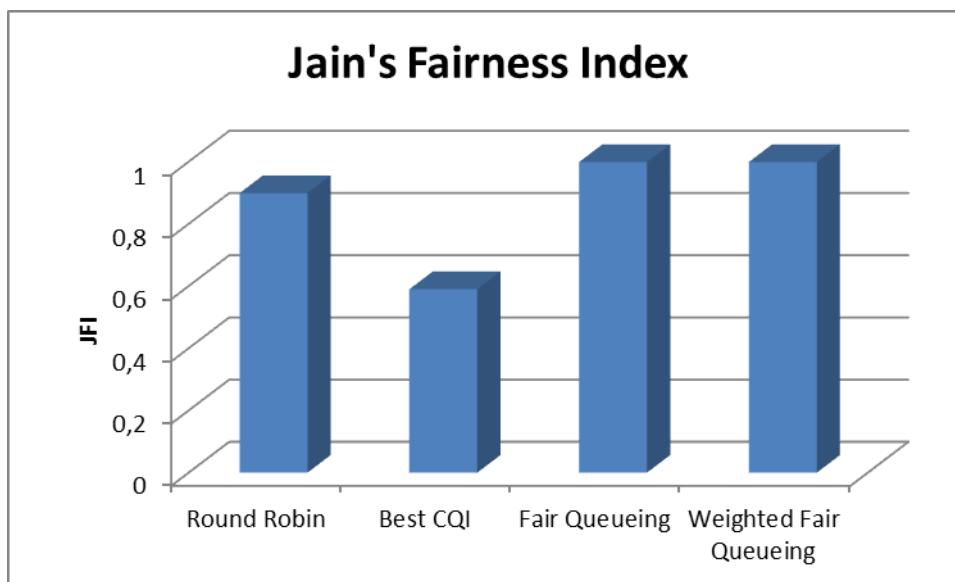
8.6 A fairness meghatározása

Algoritmusok vizsgálata során szokás beszélni arról, hogy egy-egy algoritmus mennyire fair. A fairness arról ad egy képet, hogy egy algoritmus mennyire egyenletesen osztja ki a rendszer erőforrásait a felhasználók között. Ezt a szimulációs eredmények birtokában megbecsülhetjük, létezik azonban erre egy sokkal precízebb módszer, ami a Jain's Fairness index. A Jain's Fairness Indexet Raj Jain definiálta és a (8.1) segítségével lehet meghatározni

$$JFI = \frac{(\sum_{i=1}^n x_i)^2}{n \sum_{i=1}^n x_i^2} \quad (8.1)$$

A képletben n a felhasználók száma, x_i pedig az i . felhasználóhoz tartozó átvitel értéke [14].

Egy algoritmus akkor maximálisan fair, ha minden felhasználója ugyanolyan átvitelre képes. Ebből és a 8.1 képletből látszik, hogy egy maximálisan fair algoritmus JFI értéke 1, illetve, hogy a JFI minél jobban megközelíti a 0-t annál kevésbé fair az adott algoritmus. Kiszámíthatjuk ezt az általunk szimulált ütemezésekre. Az eredményt a 8.16 ábra szemlélteti.



8.16 ábra – Jain's Fairness Index

Az ábrán jól látható, amit az átviteli sebességek alapján már megelőlegeztem, vagyis, hogy a Fair Queueing és a Weighted Fair Queueing a két leginkább fair

algoritmus, ezek JFI értéke 0.99 körül van. Ehhez képest kevésbé fair, de még mindig annak mondható a Round Robin a maga 0.85 körüli JFI értékével és – ahogyan azt a szimulációban is láthattuk – a legkevésbé fair a Best CQI algoritmus, aminek a JFI-je alig több 0.5-nél.

9. Összefoglalás

Az LTE szabvány napjaink telekommunikációs iparának egyik legfontosabb kérdése. A folyamatos számítástechnikai fejlődésnek köszönhető elvárásnövekedés egyre gyorsabb, minőségi hálózatok fejlesztését vonja maga után. A régóta üzemelő 3G hálózatokhoz képest nagy ugrást jelenthet az LTE, de még nagyobb az LTE-Advanced, ami a maga akár 5 Gbps-es névleges átviteli sebességével véleményem szerint a vezetékes hálózatok végét jelentheti. Megfelelő QoS biztosítása, illetve a sebesség maximalizálása érdekében azonban körültekintően, sok szimuláció és teszt után kell az ütemezési lehetőségek közül választani.

A dolgozatban igyekeztem bemutatni a mobil távközlés főbb sajátosságait. Az első fejezetben megemlítettem a mobilinternet kialakulásának és folyamatos fejlődésének okait, valamint bemutattam azokat a fontosabb lépcsőfokokat, amik végül az LTE-hez vezettek.

A második szakasz az LTE alapvető jellemzőiről szólt. Körvonalaztam azokat a változásokat, amelyek az LTE-ben – az elődeihez képest – történtek. Ilyenek voltak az architektúrában, közeghozzáférésben, valamint az antennahasználatban bekövetkezett változások.

A dolgozat harmadik részében részletesen bemutattam az LTE rádiós működését. Megvizsgáltam a downlink, illetve az uplink ütemezés során alkalmazott közeghozzáférési eljárásokat, vagyis az OFDMA-t és az SC-FDMA-t. Leírtam továbbá egy LTE keret szerepét és bemutattam a felépítését.

Ezek után vázoltam a cellás rendszerek működését. Ismertettem az ilyen rendszerek felépítését, valamint a manapság használt rendszereket. Különös figyelmet fordítottam a femtocellák működésének bemutatására, hiszen – az LTE-hez hasonlóan – ez is egy új technológia, így mindenképpen érdekes kutatási terület.

A cellás rendszerek bemutatása után rátértem a QoS technológia ismertetésére. Vázoltam, hogy melyek a QoS alapelvei, illetve, hogy hogyan lehet QoS-t megvalósítani. Bemutattam a forgalmak szétválsztásának módszerét, valamint hogy ez miért vezethet QoS biztosításhoz.

Az utolsó fejezetben ütemezési eljárásokat vizsgáltam saját fejlesztésű szimulációs környezetben. Leírtam az algoritmusok jellemzőit, valamint bemutattam a szimulációs környezetet. Végezetül pedig garfikonok segítségével értékeltem a kapott eredményeket.

Hangsúlyoznom kell azonban, hogy a dolgozatban szereplő technológiák folyamatosan fejlődnek és akár egy-egy hónap alatt komoly változások történhetnek a technológiai jellemzőkben, a szabványokban. Éppen ezért nagyon fontos, hogy folyamatosan figyelemmel kövessük az aktuális változásokat és ezeket a szimulátorba is implementáljuk. Azon túl, hogy más algoritmusokat is célszerű vizsgálni, más irányba is lehet fejleszteni a szimulátort. Ilyen például a puffér alkalmazása az osztályozott forgalmakra, hiszen így burst-ösen lehet továbbítani az adatokat, aminek következtében jobb átvitelt kapunk. Törekedni lehet például arra, hogy az erőforrásblokkokat minden esetben 100%-osan használjuk ki, szintén az átviteli sebesség növelésének céljából.

Ahhoz, hogy a szimulátort még szélesebb körben lehessen alkalmazni további fejlesztésre van szükség. A dolgozatban bemutatott eredmények azonban azt bizonyítják, hogy a szoftver a jövőben akár az iparban is hasznosan alkalmazható elemző programmá válhat.

Rövidítésjegyzék

3GPP	3rd Generation Partnership Project
AuC	Authentication Center
BER	Bit Error Rate
BSC	Base Station Controller
BTS	Base Transceiver Station
CDMA	Code Division Multiple Access
CQI	Channel Quality Indicator
CSD	Circuit Switched Data
DFT	Discrete Fourier Transform
E-DCH	Enhanced Dedicated Channel
EDGE	Enhanced Data rates for GSM Evolution
eNodeB	Evolved NodeB
EPC	Evolved Packet Core
E-UTRAN	Evolved UTRAN
FDD	Frequency Division Duplex
FDMA	Frequency Division Multiple Access
FQ	Fair Queueing
GGSN	Gateway GPRS Support Node
GMSK	Gaussian Minimum Shift Keying
GPRS	General Packet Radio Service
HARQ	Hybrid Automatic Repeat Request
HLR	Home Location Register
HSCSD	High Speed Circuit Switched Data
HSPA	High Speed Packet Access
HSDPA	High Speed Downlink Packet Access
HSUPA	High Speed Uplink Packet Access
HSS	Home Subscriber Server
IFFT	Inverse Fast Fourier Transform
IP	Internet Protocol
ISM	Industrial, Scientific, Medical
JFI	Jain's Fairness Index
kbps	kilobit per second

LTE	Long Term Evolution
MAC	Medium Access Controll
Mbps	Megabit per second
MIMO	Multiple In Multiple Out
MME	Mobility Management Entity
OFDM	Orthogonal Frequency Division Multiplexing
OFDMA	Orthogonal Frequency Divison Multiple Access
PAPR	Peak to Average Power Ratio
PCU	Packet Controll Unit
PDN-GW	Packet Data Network Gateway
PRB	Physical Resource Block
PSK	Phase Shift Keying
QAM	Quadrature Amplitude Modulation
QoS	Quality of Service
QPSK	Quadrature Phase Shift Keying
RAN	Radio Access Network
RLC	Radio Link Controll
RNC	Radio Network Controller
RR	Round Robin
SAE	System Architecture Evolution
SC-FDMA	Single Carrier Frequency Division Multiple Access
SGSN	Serving GPRS Support Node
S-GW	Serving Gateway
SISO	Single In Single Out
SJF	Shortest Job First
TBF	Temporary Block Flow
TDD	Time Division Duplex
TDMA	Time Division Multiple Access
UMTS	Universal Mobile Telecommunications System
UTRAN	UMTS Terrestrial Radio Access Network
VoIP	Voice Over IP
WFQ	Weighted Fair Queueing

Ábrajegyzék

2.1 ábra: Egy GPRS hálózat felépítése.....	8
2.2 ábra: GSM/GPRS sávszélesség-menedzsment.....	9
2.3 ábra: A UMTS architektúra.....	10
3.1 ábra: Az LTE architektúra.....	16
4.1 ábra: Az OFDM elve.....	19
4.2 ábra: Az OFDMA és az SC-FDMA összehasonlítása.....	20
4.3 ábra: Az OFDMA és az SC-FDMA idő-frekvencia diagramja QPSK esetén.....	21
4.4 ábra: LTE keretszerkezet.....	22
4.5 ábra: Egy erőforrásblokk.....	22
4.6 ábra: Egy lehetséges erőforrásblokk-allokáció downlink esetben.....	23
4.7 ábra: Egy lehetséges erőforrásblokk-allokáció uplink esetben.....	24
5.1 ábra: Egy méhsejt cellás rendszer.....	26
5.2 ábra: A frekvencia újrafelhasználás.....	27
5.3 ábra: Egy femtocella (Verizon-Samsung).....	30
6.1 ábra: Forgalomszétválasztás.....	32
7.1 ábra: A vizsgált cellák elhelyezkedése.....	35
8.1 ábra: Round Robin algoritmus.....	37
8.2 ábra: Átviteli sebességek Round Robin esetén.....	38
8.3 ábra: Százalékos átvitel Round Robin esetén.....	38
8.4 ábra: Best CQI algoritmus.....	39
8.5 ábra: Átviteli sebességek Best CQI esetén.....	40
8.6 ábra: Százalékos átvitel Best CQI esetén.....	40
8.7 ábra: Fair Queueing algoritmus.....	41
8.8 ábra: Átviteli sebességek Fair Queueing esetén.....	42

8.9 ábra: Százalékos átvitel Fair Queueing esetén.....	42
8.10 ábra: Forgalmotípusok kiszolgáltsága Fair Queueing esetén.....	43
8.11 ábra: Weighted Fair Queueing működés.....	44
8.12 ábra: Átviteli sebességek Weighted Fair Queueing esetén.....	45
8.13 ábra: Százalékos átvitel Weighted Fair Queueing esetén.....	45
8.14 ábra: Forgalmotípusok kiszolgáltsága Weighted Fair Queueing esetén.....	46
8.15 ábra: Összesített átviteli sebességek.....	47
8.16 ábra: Jain's Fairness Index.....	48

Irodalomjegyzék

- [1] *Nokia 9210 Communicator High Speed Circuit Switched Data*, Nokia Mobile Phones Ltd.
- [2] Brahim Ghribi, Luigi Logrippo: *Understanding GPRS: the GSM packet radio service*, Elsevier Science B.V., 2000
- [3] David Tipper: *UMTS overview*, University of Pittsburgh
- [4] Juha Korhonen: *HSDPA – An Introduction*, A TTPCom White Paper, 2004
- [5] Pete Goldsack: *HSUPA Overview*, Agilent Technologies, 2009
- [6] Schulz Schindler: *Introduction to MIMO (Application Note)*, Rohde & Schwarz GmbH & Co. KG
- [7] Fazekas Péter: *LTE: A rendszer felépítése, működése és a rádiós interfész*, BME-HIT
- [8] *3G Long-Term Evolution (LTE) and System Architecture Evolution (SAE)*, Ilmenau University of Technology, 2011
- [9] Eric Dahlman, Stefan Parkvall, Johan Sköld, Per Beming: *3G Evolution: HSPA and LTE for Mobile Broadband, First Edition*, Academic Press, 2007
- [10] Dr. Dárdai Árpád: *Ortogonalis frekvenciaosztású többszörös hozzáférés*, *Híradástechnika LIX. évfolyam 2004/4.*
- [11] Jingyuan Zhang, Ivan Stojmenovic: *Cellular Networks*, University of Alabama, University of Ottawa, 2005
- [12] Madocsai Judit: *Femtocella*, BME-VIK, 2010
- [13] Aaron Balchunas: *Introduction to QoS v1.21*, 2010
- [14] Raj Jain, Arjan Duresi, Gojko Babic: *Throughput Fairness Index: An Explanation*, The Ohio State University