

# TCP-ELN: hatékony kommunikáció rádiós csatornák felett

Buchholcz Gergő, Do Van Tien  
BME Híradástechnikai Tanszék, {buchholcz,do}@hit.bme.hu

Thomas Ziegler  
Telecommunications Research Center Vienna, ziegler@ftw.at

**Kulcsszavak:** vezeték nélküli távközlés, bithibák és csomagvesztés kezelése, TCP protokoll

A vezeték nélküli hálózatok térhódításával egyre nagyobb jelentőséget nyer a rádiós csatornák feletti hatékony kommunikáció. A jelenleg uralkodó TCP verziók nem veszik figyelembe a vezeték nélküli csatornák sajátosságait, így teljesítményük messze elmaradhat a kihasználható kapacitástól. Cikkünkben összefoglaljuk a probléma megoldására tett eddigi javaslatokat és bemutatunk egy új TCP variánst, amely – a megvalósíthatóságot is figyelembe véve – jelentősen javíthatja az adatátvitel sebességét. Megoldásunk hatékonyságát szimulációk felhasználásával mutatjuk be.

## 1. Bevezetés

A TCP protokollt a 80-as években fejlesztették ki azzal a céllal, hogy megbízható adatátvitelt nyújtson úgy, hogy hatékonyan és fair módon használja ki a kommunikáló felek közt levő szabad sáv szélességet. A megbízható adatátvitelt nyugták használatával valószínűsítették meg, a hatékony és fair sáv szélesség kihasználást pedig az egyre újabb folyamszabályozási algoritmusokkal igyekeztek elérni. Ezek az algoritmusok jól teljesítenek vezeték nélküli hálózatokban, azonban fejlesztésük során nem vették figyelembe a vezeték nélküli csatornák sajátosságait. Így a Tahoe, Reno, NewReno, vagy Westwood [5] TCP verziók ugyan kielégítő megoldást jelentenek vezeték nélküli környezetben, a rádiós hálózatokban teljesítményük jelentősen kisebb az elvárnál a rádiós csatornákon tapasztalható magas bithiba-arány miatt.

A probléma bemutatásához tételezzünk fel egy TCP folyamatot, amely 2 peert egy rádiós linket is tartalmazó hálózat felett köt össze. A rádiós bithibák miatt elvesztett csomagokra a folyamszabályozó algoritmus a küldési ablak méretének csökkentésével, azaz a küldési sebesség mérséklésével reagál, annak ellenére, hogy adott esetben nincs torlódás a hálózatban. Ennek következtében a hálózat kihasználtsága messze elmaradhat az optimálistól.

Több javaslat is született a TCP hatékonyságának növelésére. A javaslatok egyik főbb csoportja az IP alatti adatkapcsolati rétegben alkalmazott ARQ módszerre épül (például 802.11, UMTS nyugtázott mód). Az újraküldés hathatós és skálázható megoldást jelenthet, azonban ahhoz, hogy a késleltetés és a buffer méret ne legyen túl nagy, korlátozzák az újraküldések maximális számát. Így elképzelhető, hogy az ARQ alkalmazásával sem szűnik meg a csomagvesztés. A visszamaradó csomagvesztés mértéke [1] alapján akár 6% is lehet egy ipari környezetben alkalmazott WLAN kapcsolat esetén. Ez is azt igazolja, hogy szükség van olyan kiegészítő mechanizmusokra, melyek képesek fokozni a TCP protokoll teljesítményét, akár ARQ alkalmazása

mellett is, kritikus állapotú rádiós csatornák felett. Természetesen az ARQ, illetve a FEC rádiós csatorna feletti alkalmazása nélkül (például UMTS transzparens mód) a TCP folyam még kevésbé védett a rádiós bithiba okozta csomagvesztésekkel szemben, így ekkor az új eljárások alkalmazása még indokoltabbá válik.

A javaslatok egy másik nagy csoportja közvetlenül a transzport rétegben történő módosításokra épül. Ezen módosítások célja, hogy a TCP képes legyen a csomagvesztés okai, a torlódás és a rádiós bithiba (a továbbiakban csak bithiba) között különbséget tenni. Az I-TCP [2] két részre bontja a TCP folyamatokat egy fix és egy mobil peer között úgy, hogy a vezeték nélküli részen egy optimalizált protokollt használ az adatátvitelre. A felbontás elfedi a bithiba okozta csomagvesztéseket a TCP protokoll előtt, így növelve a teljesítményt. A Berkeley SNOOP protokoll [3] a TCP csomagok bázis állapotánál (BS) történő eltárolásával és a visszaérkező nyugták függvényében történő újraküldésükkel próbálja fokozni a TCP teljesítményét. A módszer bonyolultsága és rossz skálázhatósága miatt nem hathatós megoldás, hasonlóan az I-TCP protokollhoz. Bakshi és társai [4] megvizsgálták az MTU méretének hatását a TCP teljesítményére, és javaslatot tettek egy módszerre, mely alapja, hogy a TCP fogadó ICMP csomagok felhasználásával tájékoztatja a TCP forrást a rádiós csatorna állapotáról, és a forrás ezekhez a jelzésekhez igazítja a működését.

Balan és társainak [8] javaslata szerint, a TCP fejléc option mezejébe egy, csak a TCP fejlécét védő ellenőrző összeget kell illeszteni. Ennek felhasználásával ellenőrizni lehet a fejléc épségét, és ha a payload megsérül, de a fejléc ép marad, akkor azonosítani és értesíteni lehet a TCP küldőt, hogy az átviteli ablak méretét ne csökkentse.

Az Explicit Vesztés Jelzés (ELN) [6] alkalmazása lehetővé teszi, hogy a TCP fogadó közvetlen módon értesítse a küldőt a csomagvesztések okáról. A BS a TCP csomagok helyett csupán a sorszámokat tárolja el, majd ezek alapján, az elveszettnek vélt csomagok-

hoz tartozó nyugtáknak (ACK) az ELN bitet átállítja. A módszer hátránya, hogy a BS-nél tárolni és kezelni kell a folyamatokhoz tartozó sorszám listákat, ami IPsec alkalmazása esetén nem lehetséges. További hátrány, hogy a nyugták elvesztése nincs figyelembe véve, és a módszer nem biztosít hatékony átvitelt a jelzés információk számára.

Korábbi munkánk során [7] átértelmeztük az ELN módszert, pusztán az alapelv – a TCP forrás explicit értesítése a rádiós csomagvesztésekről – megtartásával. Javaslatunk a TCP-Reno folyamszabályozási algoritmusára épült, vezeték nélküli hálózatok felett működése megegyezett a TCP-Reno működésével. Vezetéknélküli hálózatokban amennyiben a TCP fogadó képes volt értesíteni a forrást a rádiós linken elvesztett csomagok számáról, jelentős teljesítmény növekedést lehetett elérni alkalmazásával.

Munkánkban vázlatos leírást adunk a veszteségi információ megszerzésének mikéntjéről, amely a protokoll működésének egyik meghatározó eleme. Részletesen leírjuk a veszteségi információ megszerzésének módját, illetve egy olyan folyamszabályozási algoritmust is ismertetünk, mely az előzőleg javasolt protokollnál hatékonyabban használja fel a rendelkezésre álló információkat.

## 2. A TCP-ELN protokoll működése

A protokoll működéséhez elengedhetetlen, hogy a TCP fogadó és küldő képes legyen a veszteségi információkat feldolgozni. Ehhez mind a kliens, mind a szerver oldalon módosításokat kell végrehajtani a transzport rétegben.

### 2.1. Fogadó oldali módosítások

A TCP fogadónak felkészültnek kell lennie, hogy a sértetlen TCP csomagokon kívül, veszteségi információkat (VI) is kaphat az alsóbb rétegektől. Sértetlen TCP csomag érkezése esetén, a fogadó a hagyományos TCP fogadó mintájára viselkedik és generál egy nyugtát. Amennyiben VI érkezik, a fogadó eltárolja, majd a következő nyugtára ráülítve (piggybacking) elküldi a TCP forrásnak. Mivel több csomagvesztés is történhet a rádiós linken, mielőtt egy új nyugta generálna, ezért a VI-eket el kell tárolni, hogy ne vesszenek el.

A protokoll nem használ negatív nyugtákat, így a késleltetett nyugta generálás – bár sok szempontból előnyös lehet –, mégsem ajánlott, mivel túlzottan lecsökkentené a jelzési csatorna sebességét.

Ha egy nyugta elveszik a hálózatban, akkor a ráültetett jelzés sem érkezik meg a TCP forráshoz, ezáltal csökkenhet a folyamszabályozó algoritmus hatékonysága. Ennek elkerülése érdekében a TCP fogadónak redundáns módon kell a jelzéseket továbbítani. Definálni kell, hogy egy nyugtában legfeljebb hány VI-t lehet eltárolni és egy VI hányszor kerül elküldésre. A mi implementációnkban egy nyugta 6 VI eltárolására képes, és minden VI háromszor kerül továbbításra.

A TCP kommunikáció során időzítők járhatnak le a TCP forrásnál. Ekkor a forrás a hálózatban levő nyugtáktól függetlenül újakezdi a csomagok küldését az utolsó nyugtázott csomagtól. A TCP fogadó által eltárolt VI-eket ilyenkor törölni kell, hiszen az általuk hivatkozott veszteségek az időzítő lejáta előtt történtek, így értesítést küldeni róluk fölösleges, sőt inkonzisztens protokoll viselkedést eredményezhet. Ennek elkerülésére a TCP forrás jelzéseket küld a timeout eseményekről a fogadónak. Az időzítőkkal a következő szakasz foglalkozik részletesebben.

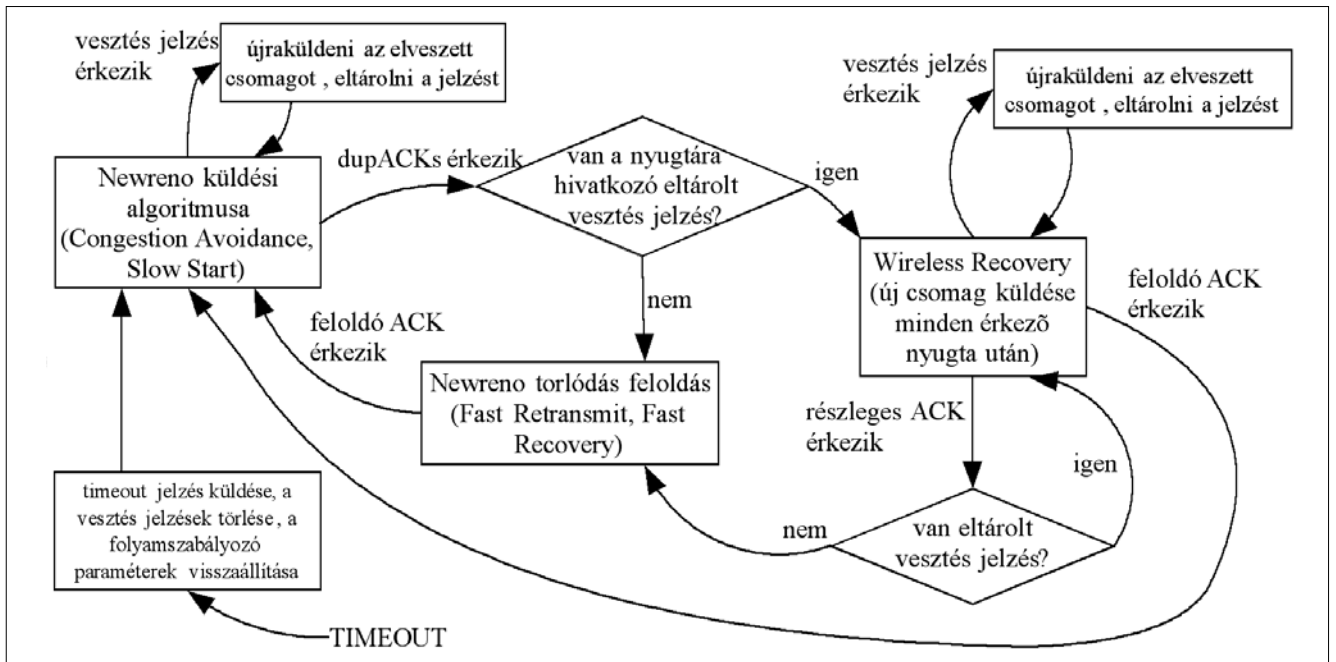
### 2.2. Küldő oldali módosítások

A TCP-ELN működését meghatározó folyamszabályozási algoritmus a TCP-NewReno-ra épül. Amennyiben nem érkezik visszajelzés rádiós csomagvesztésről, a TCP küldő a TCP-NewReno-val megegyező módon viselkedik. Vesztes jelzések (VJ) érkezésének hatására a TCP forrás a bithiba miatt elvesztett csomagokat újraküldi és eltárolja a VJ-eket. A csomagvesztések következményeként érkező harmadik duplikált nyugtáig a forrás a Slow Start vagy a Congestion Avoidance algoritmus szerint működik. A harmadik duplikált nyugtát a forrás a csomagvesztés jelének tekinti. Ilyenkor megvizsgálja a vesztes okát, ami torlódás vagy bithiba lehet, és ennek megfelelően dönt a vesztes feloldás módjáról. Ha az eltárolt VJ-k közt nem szerepel hivatkozás az elvesztett csomagra, akkor a forrás torlódást feltételez, és ennek megfelelően – a NewReno mintájára – meghívja a Fast Retransmit és Fast Recovery algoritmusokat. Ellenkező esetben, azaz ha egy eltárolt VJ hivatkozik az elvesztett csomagra, akkor a TCP küldő egy új állapotba lép (Wireless Recovery) a bithibából eredő csomagvesztés feloldására.

Wireless Recovery állapotban a TCP-ELN forrás a feloldó nyugtára (recovery ACK) vár, ami nyugtázza az összes rádiós csatornán elvesztett csomagot. A feloldó nyugta hatására a folyamszabályozó algoritmus törli a VJ-k listáját, és a harmadik duplikált nyugta érkezése előtti állapottól folytatja a működését. Az állapottól függően a küldési ablak mérete vagy a Congestion Recovery, vagy a Slow Start szerint változik tovább.

A feloldási nyugtára várva további duplikált nyugták is érkehetnek, melyek hatására a folyamszabályozási algoritmus tovább növeli a küldési ablakát egy szegmenssel (a maximális méretig), azaz minden nyugta beérkezésekor egy új szegmenst küld ki, így fenntartva a küldési sebességet. Amennyiben részleges nyugtát (partial ACK) kap a forrás, két lehetőség áll elő attól függően, hogy a nyugtázott TCP csomagot követő szegmensre van-e hivatkozás az eltárolt VJ-k közt. Ha van, akkor a forrás lecsökkenti az adási ablak méretét az újonnan nyugtázott szegmensek össz méretével, és tovább vár a feloldó nyugtára. Ha nincs hivatkozás, a NewReno torlódás feloldó algoritmusával folytatódik a folyamat.

A nyugtákra ültetve új VJ-k is érkehetnek. Ezeket a jelzéseket a TCP küldő eltárolja, és az általuk hivatkozott csomagokat újraküldi.



1. ábra A TCP-ELN forrás folyamatszabályozási algoritmus

Az algoritmus biztosítja, hogy rádiós csomagvesztések esetén a feloldás ideje alatt ne csökkenjen le a küldési sebesség, illetve ha torlódás lépne fel a feloldás alatt, akkor a NewReno algoritmus által biztosított torlódás kezelő algoritmusok lépjenek életbe. A TCP-ELN működése az 1. ábrán látható.

Ha a TCP küldőnél lejár egy időzítő, akkor a paraméterek újra inicializálódnak a TCP-NewReno mintájára, törölődik a VJ-k listája és az eseményről jelzés indul a TCP fogadó felé a TCP szegmensekre ültetve. Mivel a szegmensek elveszhetnek, ezért a forrásnak redundánsan kell elküldenie a jelzéseket. Egy lehetséges – általunk is implementált – megoldás, ha a forrás a folyam keletkezése óta fellépett timeout események számát minden egyes szegmens fejlécébe beilleszti. Ekkor – függetlenül a csomagvesztésektől –, az első TCP fogadóhoz megérkező szegmensből kiolvasható, ha lejárt egy időzítő a küldőnél.

### 3. Veszteségi információk kinyerése

Az ELN módszer működéséhez elengedhetetlen, hogy a rádiós csatornán megsérült csomagokból veszteségi információt lehessen kinyerni. Egy VI-t három adat határoz meg, ami elég egy TCP folyam, illetve szegmens azonosításához. A három adat: a forrás IP címe, a TCP célportja és a szegmens sorszáma.

Az információ megszerzésének több módja is van, de ezek a lehetőségek erősen függenek a hálózat topológiájától, illetve konfigurációjától. A módszerek bemutatására használt topológia egy elterjedt, és a gyakorlati életben is jelentős elrendezés, melyben egy vezeték nélküli link köti össze a hálózat hozzáférési pontját és a klienst.

A topológia a 2. ábrán látható.

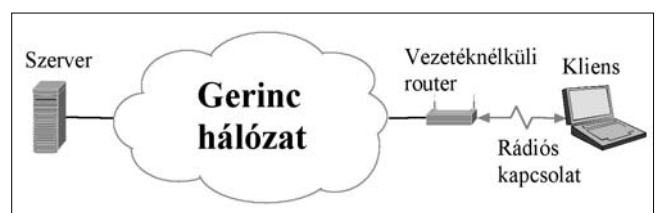
A következőkben két módszert ismertetünk ezen adatok kinyerésére olyan csomagokból, melyek megsérültek a rádiós csatornán.

#### 3.1. Módosítás az adatkapsolati rétegben

Az első módszer arra az észrevételre épül, hogy a rádiós linken bekövetkező hibák hatására nem elveszik egy MAC keret, hanem bitek átállítódásával megérkezik a fogadó félhez. Ebből a sérült keretből a TCP/IP fejléccet kiemelve azonosítható a szegmens és a folyam, amihez a szegmens tartozott. A fejléc felhasználásához ellenőrizni kell a fejléc épségét, de ezt a TCP protokoll – az IP-vel ellentétben – alapértelmezett esetben nem támogatja. Így szükséges egy új, dedikált védelem, ami csak a TCP fejléccet védi [8]. Egy alkalmas védelmi eljárás lehet az ellenőrző összeg használata, amit a TCP fejléc „option” mezőjébe lehet elhelyezni.

Ha a kliens MAC rétege egy sérült keretet kap a rádiós csatornán, ahelyett hogy eldobná, kinyeri és továbbítja a felsőbb rétegnek szánt adatokat (payload). A további rétegek hasonló módon ellenőrzik a fejléc épségét, és továbbítják az adatokat. Ha minden alsóbb réteghez tartozó fejléc sértetlen, akkor a csomag eléri a TCP réteget, ahol a VI-k kinyerése megtörténik. A módszer hatékonyságát az biztosítja, hogy a payload mérete sokkal nagyobb, mint a fejléceké, így a bithibák nagyobb valószínűséggel esnek az adat-részbe, mint a fejlécekbe.

2. ábra Tipikus rádiós hálózatelérési topológia



A MAC módosítás módszerének legnagyobb előnye az, hogy csak a TCP folyam két végpontjánál kell átalakításokat végezni, a hálózat belső elemei érintetlenek maradhatnak. Ennek következményeként viszont csak a letöltési irányba lehet az átvitel hatékonyságát növelni, hisz csak a kliens képes monitorozni a rádiós csatornát.

Ha a MAC réteg átalakítása, vagy a sérült MAC csomagokból az adatok kinyerése nem lehetséges, akkor a következőkben ismertetett módszer alkalmazható.

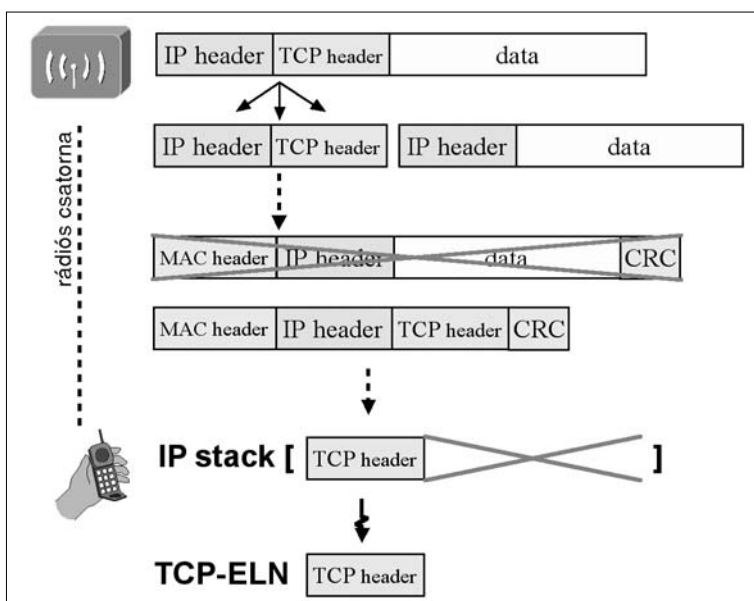
### 3.2. Módosítás az IP rétegben

A módszer arra a tényre épül, hogy a kis méretű csomagok elvesztése kisebb valószínűséggel következik be, mint a nagy méretű csomagok elvesztése. Így ha egy TCP szegmens egy IP csomagba ágyazva éri el a vezeték nélküli routert, akkor az IP csomag két részre bontásával – ahol a kisebb rész a TCP fejléccel, a nagyobb rész a payloadot tartalmazza – a VI-k nagy valószínűséggel épen érik el a kliens.

A módszer alkalmazásának feltétele, hogy a TCP szegmens egy IP csomagba ágyazva haladjon végig a hálózaton egészen a vezeték nélküli routerig. Ekkor ha a kliens IP rétegéhez nem érkezik meg egy a wireless routernél feldarabolt IP csomag valamelyik része, akkor az a rádiós linken történt hiba nyilvánvaló jele. Az IP réteg megvizsgálja a megkapott csomag részleteket, és ha köztük van az első – a TCP fejléccel tartalmazó – részlet, akkor a beágyazott TCP fejléccel megfelelő jelzésekkel továbbítja a TCP rétegnek, ahol a fejléc birtokában a TCP réteg kiolvashatja a veszteségi információt. A módszer nem teszi lehetővé, hogy torlódás miatt elveszett csomagokból VI-t lehessen kinyerni, hiszen torlódás esetén egy egész IP csomag, illetve TCP szegmens elveszik. Az eljárás működése a 3. ábrán látható.

Ha a folyam mentén valamelyik router darabolást végez az IP rétegben, akkor a módszer nem alkalmazható, mivel egy IP részlet elvesztése esetén nem lehet a veszteség okát a kliensnél meghatározni.

3. ábra IP darabolás

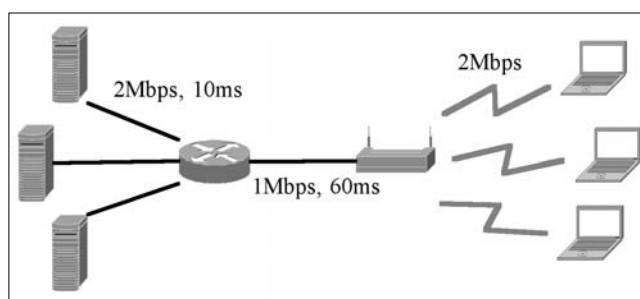


Az IP darabolás módszerének hátránya, hogy nem csak a folyam két végpontját, de a vezeték nélküli routert is módosítani kell. Mindazonáltal a vázolt eljárás egyrészt sokkal hatékonyabb, illetve skálázhatóbb más, a router átalakítását igénylő módszernél (például Snop). Másrészt az átalakítások függetlenek az alkalmazott MAC rétegtől és az IP réteg szolgáltatásainak megfelelően történnek (darabolás).

### 4. Szimulációs környezet

A TCP-ELN protokoll teljesítményét szimulációk futtatásával teszteltük.

A szimulációkhoz felhasznált topológia 8 hálózati berendezést tartalmaz: 3 szervert, 3 mobil klienst, egy vezeték nélküli és egy vezeték nélküli routert, ahogy az a 4. ábrán is látható.



4. ábra Szimulációs topológia

A szerverek és a router között 2 Mbit/s-os, 10 ms késleltetésű linkek vannak. A két routert egy 1 Mbit/s-os, 60 ms késleltetésű link köti össze. A vezeték nélküli router minden mobil állomásnak 2 Mbit/s sávszélességet szolgáltat. Minden kapcsolat full duplex. A wireless routerben nem használtunk semmilyen specifikus MAC réteget, hiszen a bemutatott módszerek általános érvényűek, így sokkal kevésbé függenek a MAC rétegtől, mint a topológiától.

Két különböző forgalom került felhasználásra: FTP és web forgalom. Míg az FTP forgalom a protokoll állandósult állapotbeli viselkedésének, a web forgalom a protokoll dinamikus viselkedésének tanulmányozására megfelelő. A futtatások során minden kliens-szerver pár között egy végtelen FTP letöltést indítottunk a 3 szerveren ugyanazzal a TCP típusal (ELN, NewReno). A két protokoll nyújtotta átviteli sebességet (throughput) hasonlítottuk össze.

Web forgalom esetén minden kliens az egyik szerverhez kapcsolódó, böngésző felhasználót szimulált. A web forgalom paramétereit a SURGE [9] modellből származtak, mely valós mérésekre épül és széleskörben alkalmazott realisztikus terhelések generálására. A futtatások során az oldalak letöltésének sebességét (oldal méret/letöltési idő) mértük az egyes TCP verziók felhasználása mellett.

A rádiós linken történő TCP szintű csomagvesztések szimulálására két modellt használtunk: uniform és egy markovi modellt. Az uniform modellnél az egyenletes eloszlás várható értékét 0 és 0,2 között, a Markov-modellnél a paramétereket a [10] tanulmány szerint változtattuk. A Markov-modell előnye, hogy alkalmas a különböző sebességű felhasználók által tapasztalt különböző csomagvesztési valószínűséggel rendelkező rádiós linkek szimulálására (1. táblázat).

Modell szám	Sebesség	Átlagos hiba vsz.	Átlagos börszt hossz
1	gyalogos	0.001	1.4913
2		0.01	4.0701
3		0.1	13.6708
4	közepes	0.001	1.0083
5		0.01	1.0838
6		0.1	1.8629
7	közúti	0.001	1.0024
8		0.01	1.012
9		0.1	1.1317

1. táblázat A Markov-hibamodell paramétere

Mivel a TCP-ELN folyamatszabályozó algoritmus erősen függ a TCP szegmensek meghibásodásának módjától (a fejléc is megsérül-e, vagy sem), ezért a futtatások során a sikeres VI kinyerésének valószínűségét 95%-ra állítottuk, figyelembe véve a fejlécek és a payload méretének gyakorlatban tapasztalható arányát. A beállítás következménye, hogy a sérült TCP szegmensek 5%-ból nem lehet veszteségi információt kinyerni.

### 5. Eredmények

Az 5. ábrán látható eredmények alapján megállapítható, hogy FTP forgalom mellett a TCP-ELN protokoll teljesítménye jelentősen nagyobb a NewReno-énál. A csomagsérülés valószínűségével nő a javulás mértéke, ami a vizsgált tartomány határánál akár a 400%-ot is elérheti. A rádiós csatorna ideális állapota esetén – nem sérülnek meg csomagok – a TCP-ELN protokoll nyújtotta átviteli sebesség kis mértékben (0,5%) alacsonyabb a NewReno sebességénél. Ennek oka a TCP-ELN által használt fejléc, aminek a mérete nagyobb a NewReno által használt fejlécnél, hogy képes legyen a vesztesé jelek eltárolására. A megnövekedett csomagméret miatt, a csomagok átviteli ideje megnő, de ez nem jelent problémát, hisz más elterjedt és használt TCP verzióknál is fellép ez a jelenség (TCP-SACK).

Az ábrán jól megfigyelhető, hogy a TCP-ELN teljesítménye sokkal finomabban törik le a hibavalószínűség növelésével, mint a NewReno teljesítménye. Míg a TCP-ELN által nyújtott sebesség 14%-os hibavalószínűségnél esik le 90%-ra, addig a NewReno-t használva ugyanez a sebességcsökkenés már 4%-os hibavalószínűség mellett bekövetkezik. A Markov-hibamodell mellett a javulás mértéke elérheti a 185%-ot a nagy hibavalószínűségű esetekben (10%-os hiba valószínűség, 3-as, 6-os, 9-es modell) a börsztösség függvényében. A kis hibavalószínűségű esetekben (1-es, 4-es, 5-

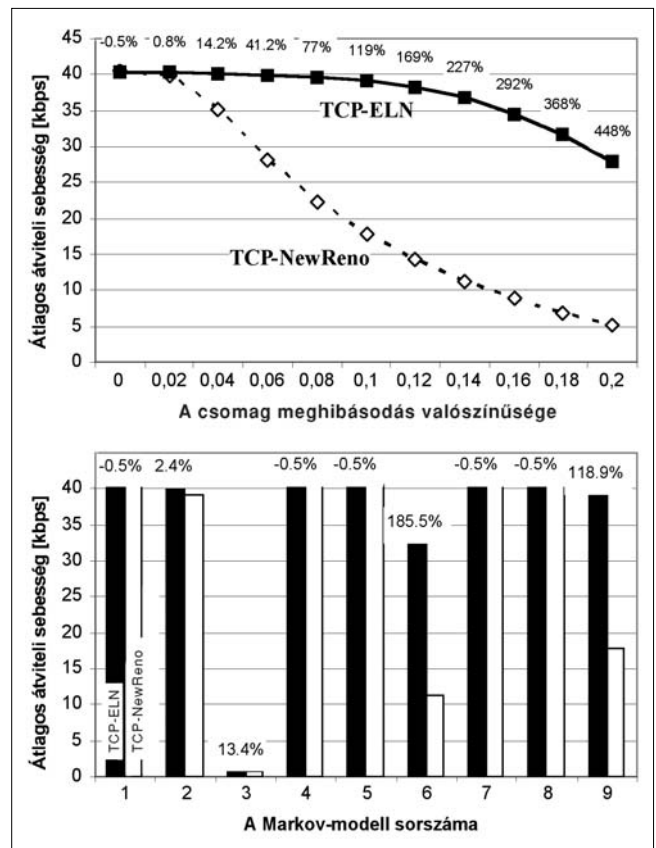
ös, 7-es, 8-as modell) egy kisebb sebességcsökkenés tapasztalható az előbb részletezett ok miatt. A 2-es modell esetén a javulás 2%-os.

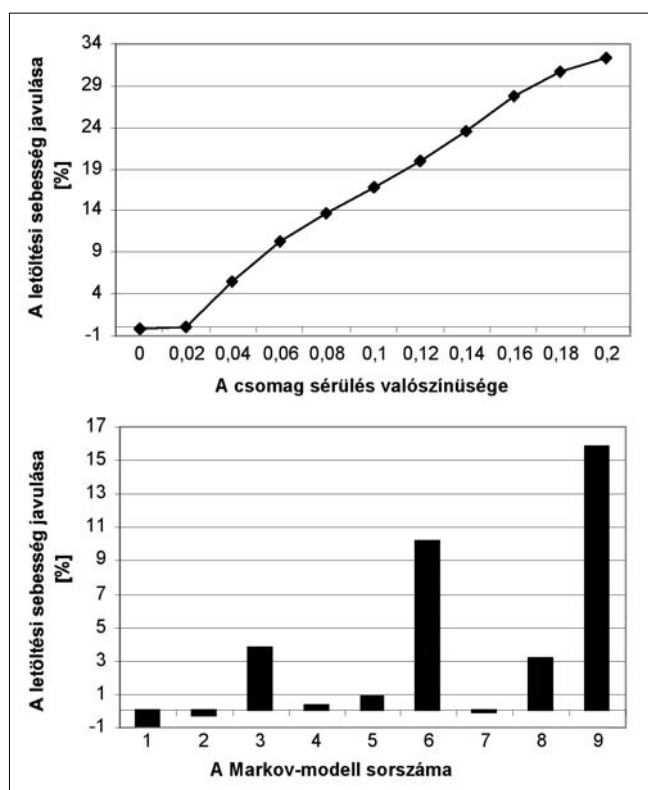
Az uniform és a Markov-modell melletti relatív javulás értékeit összehasonlítva fontos összefüggés állapítható meg. A Markov-modell sebesség paraméterének megnövelése lecsökkenti a hibabörsztök hosszát. Így ezekben az esetekben, ahogy a csomagvesztések közt csökken a korreláció, a Markov-modell szolgáltatja eredmények az uniform modell eredményeihez konvergálnak.

Ahogy a 6. ábrán látható, web forgalom esetén az eredmények kevésbé kiugróak, de még így is jelentősek. Uniform hibamodell esetén a TCP-ELN teljesítménye közel azonos a NewReno-éval 2%-os hibavalószínűségig, majd a javulás mértéke 33%-ig fut fel nagyobb hibavalószínűségekre. Ha nincs csomag sérülés a rádiós csatornán, a TCP-ELN teljesítménye 1%-ot esik a fejléc mérete miatt. Markov-modell esetén a javulás mértéke 4% és 16% közé esik a nagy hibavalószínűségű modellekre, a kis hibavalószínűségű modelleknél a relatív javulás kevesebb.

Összehasonlítva a web és az FTP forgalom által szolgáltatott eredményeket, megfigyelhető, hogy FTP forgalom esetén a TCP-ELN relatív teljesítménye jóval nagyobb. Ennek oka az, hogy a felhasznált HTTP-szerű web forgalom esetén a TCP folyamat többségének mérete kicsi, csak pár csomag. A kis folyamat pedig nem teszik lehetővé, hogy a küldési ablak mérete

5. ábra Az adatátviteli sebesség és a TCP-ELN által nyújtott relatív javulás mértéke FTP forgalom esetén, uniform és Markov-hibamodell mellett





6. ábra

A TCP-ELN által nyújtott relatív javulás a letöltési sebességben uniform és Markov-hibamodell mellett

nagyra nyílnon, így az ELN folyamatszabályozási algoritmus nem képes olyan mértékben megnövelni az átviteli sebességet az ablak felezések elkerülésével, mint nagy folyamok, illetve küldési ablakok esetén.

Elmondható, hogy p2p forgalom esetén a TCP-ELN teljesítmény növekedése az FTP és web forgalom nyújtotta eredmények közé várható, az FTP esetét felső, a web esetét alsó korlátnak tekintve.

## 6. Összefoglalás

Cikkünkben bemutatunk egy új módszert, a TCP kapcsolatok rádiós csatornák feletti teljesítményének növelésére. Módszerünket az explicit vesztes jelzés (ELN) technikájára alapoztuk, azzal a céllal, hogy képessé tegyük a TCP forrást a csomagvesztések okainak: a torlódásnak és a rádiós bithibáknak a megkülönböztetésére. Ezáltal a TCP folyamatszabályozó algoritmus képes elkerülni a rádiós csomagvesztésekből eredő fölösleges ablakcsökkentést, aminek következtében a teljesítménye jelentősen jobb lesz összehasonlítva az elterjedt TCP verziókkal.

Két módszert is kidolgoztunk a veszteségi információk megszerzésére, ami elengedhetetlen az ELN működéséhez. Az első módszer a mobil kliens adatkapcsolati rétegének módosítását igényli, a második eljárás egy speciális IP darabolási technikára épül.

Szintén bemutatásra került egy új, általunk fejlesztett TCP variáns, a TCP-ELN, mely integrálja a veszteségi információk megszerzésének módszerét a TCP-New-

Reno folyamatszabályozási algoritmusával. A küldő és fogadó oldali módosítások részletes tárgyalásra kerültek.

A protokollt különböző forgalmak (FTP, web) és rádiós hibamodellek (uniform, Markov) mellett is teszteltük. Szimulációs eredményeink azt mutatták, hogy a TCP-ELN alkalmazása igen hatékony a hálózati környezetek széles spektrumán. Magas csomagvesztési valószínűségű rádiós csatornán, FTP forgalom mellett a relatív javulás mértéke akár a 400%-ot is elérheti, web forgalom mellett a javulás 30%-ot ért el.

## Irodalom

- [1] A. Willig, M. Kubisch, C. Hoene, A. Wolisz, Measurements of a Wireless Link in an Industrial Environment Using an IEEE 802.11 – Compliant Physical Layer. IEEE Transactions on industrial electronics, Vol. 49, No.6., December 2002.
- [2] A. Bakre, B. R. Badrinath, I-TCP: Indirect TCP for Mobile Hosts. In: Proc. 15th International Conf. on Distributed Computing Systems (ICDCS), May 1995.
- [3] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, R.H. Katz, A Comparison of Mechanisms for Improving TCP Performance over Wireless Links, IEEE/ACM Trans. on Networking, 5(6), December 1997.
- [4] B. Bakshi, P. Krishna, N. H. Vaidya, D. K. Pradhan, Improving Performance of TCP over Wireless Networks, 17th Int. Conf. Distributed Computing Systems, Baltimore, May 1997.
- [5] C. Casetti, M. Gerla, S. Mascolo, M. Y. Sanadidi, R. Wang, TCP Westwood: Bandwidth Estimation for Enhanced Transport over Wireless Links, In: Proc. ACM Mobicom'2001, pp.287–297., Rome, Italy, July 16-21, 2001.
- [6] Hari Balakrishnan, Randy Katz, Explicit Loss Notification and Wireless Web Perform., Proc. IEEE GLOBECOM Global Internet Conf., Sydney, Australia, November 1998.
- [7] G. Buchholz, A. Gricser, T. Ziegler, T. Van Do, Explicit Loss Notification to Improve TCP Performance over Wireless Networks, 6th IEEE High-Speed Networks and Multimedia Communications (HSNMC), 2003.
- [8] R. K. Balan et. al, TCP Hack: TCP Header Checksum Option to Improve Performance over Lossy Links, IEEE Infocom 2001.
- [9] P. Barford, M.E. Crovella, Generating Representative Web Workloads for Network and Server Performance Evaluation, In: Proc. of Performance, ACM SIGMETRICS'98, pp.151–160., Madison WI.
- [10] A. Chockalingam, M. Zorzi, Rames R. Rao, Performance of TCP on Wireless Fading Links with Memory, ICC 1998.