Wireless Internet Access Based on UTRA-TDD

E. González Parada, J. M. Cano García, J. A. Aguilar, A. Reyes Lecuona and A. Díaz Estrella.

Departamento de Tecnología Electrónica. Universidad de Málaga.

FAX: +34 95 2131447 E-mail: eva@dte.uma.es

ABSTRACT

The TDD operation mode of the UMTS terrestrial radio access is specially suitable for Internet access. The analysis of different UTRA/TDD protocol stack configurations and the study of interactions between layered protocols (TCP/IP and UTRA/TDD) are key issues in order to optimize wireless Internet access. In this paper, the architecture of a UTRA/TDD system-level versatile simulator meant to examine Internet applications performance is presented. The simulator includes novel application-level traffic models capable of reproducing real traffic features, which are not captured by traditional traffic models. Our aim is to study the system performance when working with several TCP flavours, different UTRA/TDD protocol stack configurations and distinct Internet and radio channel characteristics. We include several TCP versions (Tahoe, Reno, NewReno, SACK/FACK) and TCP wireless enhancements. In addition, we analyse the performance of different RLC modes and compare them to a TCP wireless enhancement such as Snoop.

1 INTRODUCTION

In recent years the popularity of Internet based applications has led to the development and adaptation of new telecommunication networks, being already available even in systems as innovative as mobile networks [1]. Providing an efficient wireless access to the Internet is one of the main goals of third generation (3G) mobile systems such as the Universal Mobile Telecommunication System in Europe.

The UMTS terrestrial radio access (UTRA) consists of two modes, a frequency division duplex (FDD) mode, and a time division duplex (TDD) mode, in order to perform an efficient use of the paired and unpaired band of the allocated spectrum. The FDD mode is intended for applications in macro and microcell environments with medium data rates and high mobility. The TDD mode is particulary well suited for environments with high traffic density and indoor coverage, where the applications require high data rates and tend to create asymmetric traffic (e.g. Web browsing).

Most popular Internet applications such as the Web, file transfer, and e-mail use the reliable transport service provided by TCP. The performance perceived by users of such applications strongly depends on the performance of TCP. TCP has been designed and tuned to perform well in traditional wired networks, since losses are assumed to be a symptom of congestion. However, mobile cellular systems suffer from handover disconnections, corruption losses, and low variable bandwidth, which dramatically affects TCP performance. A growing effort have been devoted in the recent years to solve these problems [2].

This paper provides an overview of the UTRA-TDD system technology and architecture. Further, a simulation based evaluation of the wireless Internet Access through the UTRA-TDD system is presented and the performance of different system configurations are studied. In particular, our analysis focuses on the Web browsing, the most popular Internet application at the present [3].

2 SYSTEM ARCHITECTURE AND PROTOCOLS

2.1 UMTS Network Architecture

The UMTS Network architecture includes the Core Network, the Radio Access Network and the User Equipment (UE), as can be seen in figure 1. This division provides the necessary flexibility by allowing the coexistence of different access techniques and different core network technologies, thus facilitating the migration form 2G to 3G networks [4].

The Core Network (CN) consists of a number of databases and switches. CN connects the Radio Access Network with standard public networks, performs handover and supports charging, accounting and roaming. The CN is divided into the circuit switching domain and the packet switching domain. The Mobile Switching Centers (MSC) form the backbone of the circuit switching domain, inherited from the 2G GSM networks. On the other hand, for data packet traffic the GPRS support nodes (SGSN) are used. The SGSN keep certain location information and are interconnected with the GPRS gateway (GGSN), which is the interface to external packet networks such as Internet.

UTRAN consists of several Radio Network Subsystems (RNS) connected to the Core Network. The RNS can also be interconnected between them. The RNS is divided into the Radio Network Controller (RNC), and several base stations or Nodes B connected to the RNC. The RNC manages the radio resources and handles handovers from one Node B to another, whereas the Nodes B handle the radio communication link.

The UE is connected to the UTRAN through the Uu radio interface, which supports both FDD and TDD mode operation. For both modes the same network architecture and the same protocols are used. Only the physical layer and the Uu interface are specified separately. This paper focuses on the TDD operation mode of the radio interface.

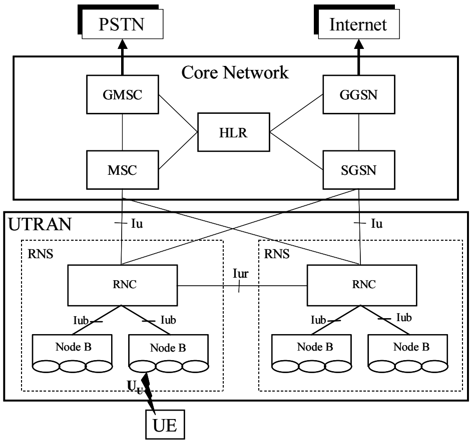

Figure 1: UTRAN Architecture.

2.2 Radio Interface Protocol Architecture

The Radio Interface Protocol Architecture, see figure 2, is divided into three layers: the physical layer (L1), the data link layer (L2) and the network layer (L3) [5]. L2 is further divided into four sublayers: the Medium Access Control (MAC), the Radio Link Control (RLC), the Packet Data Convergence Protocol (PDCP) and the Broadcast-Multicast Control (BMC).

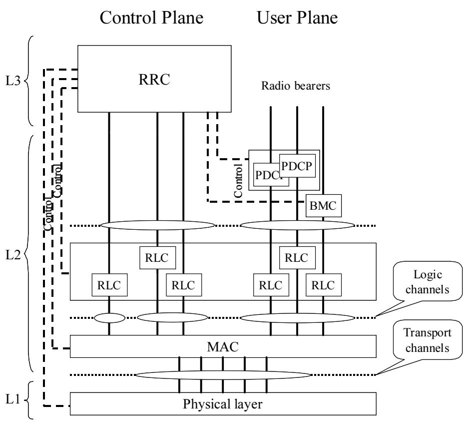

Figure 2: Radio Interface Protocol Architecture

2.2.1 *Physical Layer*

L1 offers information transfer services through the air interface to the MAC sublayer by means of the transport channels. These are described by how and with what characteristics data are transferred over the radio interface. There are two types of transport channels: Common and Dedicated transport channels. A dedicated channel implies the allocation of resources to a single UE, whereas the resources allocated to a common channel are shared between different UE. Both of them can be used to transport either control or user data traffic.

The physical layer is responsible for the forward error correction (channel coding and interleaving) and detection, multiplexing of different transport channels on the same physical resources,rate matching and modulation.

The multiple access scheme divides the available bandwidth into physical channels on which transport channels information is mapped. The multiple access technique proposed for the radio interface is based on DS-CDMA (*Direct Sequence Code Division Multiple Access*). In the UTRA-TDD mode an additional TDMA component is added, so that an hybrid TDMA-CDMA access scheme is used. The TDMA frame has a duration of 10 ms and it is subdivided into 15 time-slots. This frame is divided into downlink and uplink parts, and the switching point can be moved to support asymmetric traffic. In each time-slot a simultaneous transmission of up to 16 burst by means of different spreading sequences is possible.

The physical layer also performs other functions related to the measurement of the radio link quality and power control.

2.2.2 *L2 layer*

MAC The MAC sublayer maps logical channels onto the transport channels offered by the physical layer, providing data transfer services to RLC sublayer. The MAC layer is basically responsive for multiplexing, priority handling between different data flows, and shared channels management. MAC resource allocation and configuration is controlled by the RRC layer. Logic channels are divided into control channels and traffic channels. There are also dedicated and common logical channel, but in the sense that they are point to point or point to multipoint respectively.

RLC The RLC deals with error correction ARQ procedures and data segmentation and reassembly. This sublayer provides transparent, unacknowledged or acknowledged mode data transfer to the upper layers. The acknowledged mode uses a sliding window protocol to assure error free data transmission.

PDCP The PDCP layer transparently carry network layer PDU (e.g. IP packets), adapting user data traffic packets to the wireless link. Additional functions such as header compression can be also applied.

BMC BMC protocol provides a broadcast/multicast user data transport service.

2.2.3 *L3 layer*

RRC The Radio Resource Control (RRC) layer handles the control plane signalling of Layer 3 between the UEs and UTRAN. It is also responsible for adequately configuring and controlling the lower sublayers of the radio interface. The RRC layer establishes the radio resources management policy so that the QoS requirements of the different communications can be satisfied. This includes assignment, reconfiguration and release of radio resources, as well as continuous control of the QoS. In the TDD mode the RRC performs additional functions such as dynamic channel allocation and timing advance.

3 SIMULATION MODEL

Simulation is a vital tool to quickly and inexpensively explore the behaviour of communication networks. In order to determine the end-to-end performance of the UTRA-TDD system from a user perspective, we have developed a system-level simulator based on OPNET 5.1.D.

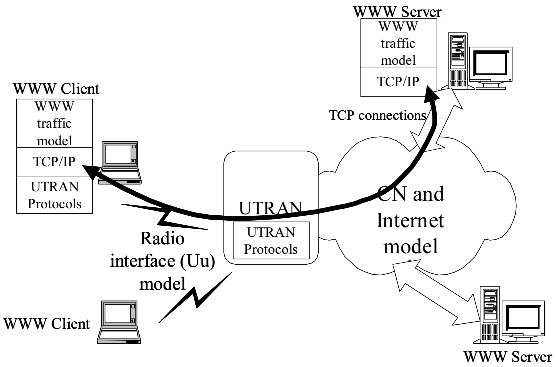

Figure 3: Simulation model

discrete-events simulation kernel. The simulator is also intended to measure the capacity and throughput from the system point of view in order to dimension the radio access network.

As seen in fig. 3, our simulator comprises a model of the radio interface protocol stack, a model of the traffic generated by the WWW clients and a model of the involved Internet protocols. All these simulation models will be detailed in the following sections.

3.1 Traffic Model

In order to evaluate a system performance by means of simulations, traffic models capable of reproducing the features of the traffic generated by the applications are required. In the last decade, several studies [6, 7, 8] have demonstrated that WWW traffic, which is the main Internet traffic source [3], is self-similar. In other words, it presents a significant burstiness on a wide range of time scales. This fractal behaviour has serious implications for the dimensioning of Internet data networks, since aggregating streams of such traffic intensifies the self-similarity instead of smoothing it.

Many proposed models for TCP/IP traffic operate at what might be called the link-level [9], and are capable of emulating the arrival of packets observed over a link. However, Arvidsson and Karlsson [10] postulated that such models can not capture the TCP/IP protocols and network effects (losses, delays, etc.). They proposed the user-level models (or application-level models), which are based on the user actions and need to be integrated with models of the protocols (HTTP, TCP and IP) and the network. Further, it have been recently shown that TCP can be one of the causes of the Internet traffic self-similarity [11, 12].

In our simulator we use an application level model for WWW traffic, the Connection Oriented Model (COM) [13]. This model is implemented into a hierarchical multilevel structure relying on the user behaviour, so that parameters have physical meaning. This structure presents three levels: session, page and connection level (see fig. 4). A session is the period ranging from the moment the user starts the browser until it is switched off. During the session, the user navigates through several web pages and therefore it consists of several periods of page loading separated by periods of page reading. A page transfer consists of several TCP connections working in a parallel way and implying the transmission of a certain amount of data in both uplink and downlink directions.

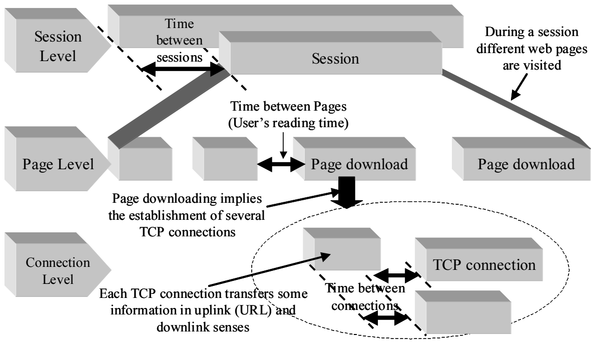

Figure 4: WWW traffic model structure

Variable	Distribution	Parameters	
Time between sessions	Exponential		
Pages per session	Lognormal	$\mu = 14$	$\sigma = 36$
Time between pages	Gamma	$\mu = 46,8$ s	$\sigma = 168,6$ s
Connections per page	Lognormal	$\mu = 5,3$	$\sigma = 12$
Time between connections	Gamma	$\mu = 2,3$ s	$\sigma = 4,5$ s
Connection size (downlink)	Pareto	$\mu = 5616$ Bytes	$\alpha = 1,77$
URL size (uplink)	Lognormal	$\mu = 364$ Bytes	$\sigma = 101$ Bytes

Table 1: WWW traffic model parameters

The parameters involved in this model can be seen in the table 1. A number of them correspond to the user behaviour, whereas other parameters capture application characteristics. A Pareto distribution with $1 < \alpha < 2$ (finite mean and infinite variance) is used to model the connection size, this feature incorporates a self-similar behaviour into the traffic model.

3.2 TCP/IP Model

TCP is a reliable transport protocol based on a sliding window mechanism and includes window-based flow control and congestion avoidance techniques. TCP sources slow down the transfer rate when losses are detected, since they are considered as an implicit congestion signal [14]. This fact inflicts a serious damage to TCP performance over lossy links, such as wireless networks, and should be considered in simulations.

TCP is a complex protocol with a wide range of user-configurable parameters [15, 16]. Even worst, there is not a standard version of TCP, as long as variations on the basic congestion control [14] have been continuously proposed and deployed. Since the achievable performance of a TCP connection strongly depend on its configuration, our TCP model implements all relevant mechanisms [17], and some of the most popular improvements proposed in the literature [18, 19, 20]. Thus, it is capable of simulating most widely deployed TCP versions (Tahoe, Reno, NewReno, SACK/FACK) behaviour.

We have also included a model of the Snoop protocol [2], which is a TCP enhancement for wireless links based on a TCP-aware performance-improving proxy capable of retransmitting packet over the wireless link.

We have not included a detailed model of IP in our simulations as long as no routing aspects have been considered. However IP header overhead has been included in the transmitted packet.

3.3 Internet Model

In section 3.1, we justified the need for incorporating the TCP/IP protocols and Internet models and integrating them with the user level traffic models. In spite of being a very difficult task [21], several studies trying to characterize the Internet, most of them centered in the end-to-end delay and in the packet losses [22, 23, 24, 25, 26], are present in the literature.

In order to simulate the Internet, models for delay and losses are discussed in the following sections. Both parameters will be characterized independently and no correlation between them will be considered.

3.3.1 End-to-End Delay

The model assumes that the delay of the packets belonging to a given TCP connection fits a gaussian distribution. It also aims at characterizing the correlation of the delay presented by close packets belonging to the same connection. To do so, a given packet delay is calculated according to the previous one [27]. This dependency exponentially decreases along with time separating both packets:

$$D(t) = a(T)(D(t-T) - \mu) + b(T)w(t) + \mu$$

$$a(T) = e^{\frac{-T}{\tau}} \quad b(T) = \sigma\sqrt{1 - a(T)}$$

Where D is the current packet delay, $D(t-T)$ the previous packet delay, T the time between both packets and $w(t)$ a random gaussian variable presenting zero mean and a standard deviation equal to 1. μ and σ are used to adjust respectively the mean and the standard deviation of the delay distribution, and the exponential decrease of the autocorrelation function can be controlled by means of the τ parameter.

This model adjusts correlation of packets belonging to a same TCP connection at short range time scales. Although some studies show that LRD exists in transmission delay [23], it must be considered that the proposed model is meant for simulating using TCP connections so it would not be necessary to evaluate time scales longer than connections duration. Besides, the condition $D_i < D_{i-1} - T$ should be imposed in order to prevent packet disordering, since it is not realistic and affects TCP performance [27].

3.3.2 Losses

Two models are presented: a simple Bernoulli model and a Loss Burst model. As shown in [27] independent or bursty losses affects TCP performance differently.

- Bernoulli model. This model fixes a loss probability p that is independently applied to each packet belonging to a given connection. Thus, each time a packet goes through the network a random experiment presenting a success probability equal to p is performed. p determines if a packet will be lost or if it will be successfully transmitted through Internet.
- Loss Burst Model. Internet losses are not generally due to errors, but to the congestion of some routers in the path of the sent packets. Thus, losses usually appear in bursts, as proven in several studies [26, 25]. Hence, it might be of interest to take this fact into account in the Internet network model. [25, 28] analyse packet losses by sending equally spaced packets and using, among other possibilities, a two-state Markov chain to model them. According to this approach, a two-states model, where each state presents an exponential duration, will be used.

Loss bursts in Internet are mainly due to the use of *drop-tail* queue management policy. RED (*Random Early Detection*) policy [29], that uses the random discarding of packets to prevent congestion, has been proposed to improve the fairness of Internet and is being currently deployed. This could lead to a different loss model such as the Bernoulli model.

3.4 Radio Access Network Model

3.4.1 RLC model

We have included in our simulator a model of the acknowledged mode of the RLC. According to this model the RLC divides upper layer PDUs into smaller RLC blocks, adds them a sequence number and transmit them following a sliding-window based mechanism. Corrupted blocks are rejected by the receiving RLC and retransmitted. Communication stall is avoided by mean of a *watchdog* timer.

3.4.2 RRC/MAC model

The UTRA-TDD system is highly flexible and configurable, transport services through the air interface (radio bearer) can rely on dedicated or shared resources. Since we are considering a data packet traffic application, a demand assignment resource sharing mechanism seems to be more appropriate. Therefore, we have considered that users accessing the Internet through the system are using Dedicated Traffic Channels mapped into Uplink and Downlink Shared Transport Channels (USCH and DSCH). Shared transport channels ease resource sharing between different UEs, but require the use of the SHCCH logical channel in order to control resource allocation. The uplink operation mode is different than the downlink (see fig. 5 and 6).

- Before a UE can transmit information in a USCH, some resources should be allocated to this channel. Thus, when the RRC layer of the UE detects packet waiting in the RLC buffer to be transmitted, it sends a Capacity Request to the network using the SHCCH logical channel. The network RRC scheduling function will decide if physical resources will be allocated to the USCH transport channel and sends a resource allocation message to its peer entity in the UE, specifying the allocated resources and the period of time they can be used. Once the physical resources are allocated, both RRC in the network and the UE configure their respective Layer 1 and MAC for the data transfer on the USCH, and at the specified time MAC in the UE conveys the data using the

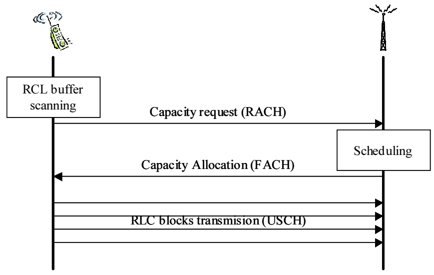

Figure 5: UL operation model

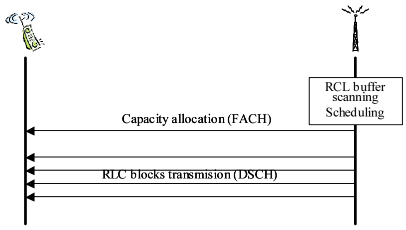

Figure 6: DL operation model

specified USCH resources. When the transmission of the RLC blocks finishes, allocated resources are released.

- The operation of the DSCH is simpler. The network RRC scheduling function will decide to allocate physical resources to the DSCH when it infers that there is some information to transmit to the UE. In order to configure the UE the SHCCH is used to send a capacity allocation message which designates the physical resources assigned to the DSCH. Once downlink resource units are assigned to the UE, the network transmit the information through the DSCH.

We have considered that the uplink SHCCH logical channel is mapped onto the RACH transport channel and the downlink SHCCH is mapped onto the FACH transport channel. The Random Access Channel is an uplink contention-based common transport channel used for transmitting small amounts of data such as control information. One or several time slot may be reserved to RACH and are used by the UEs to transmit information bursts without previous resource reservation. When two UE using the same code attempts to transmit simultaneously in the RACH time-slot a collision ocurrs and the information should be retransmitted. Thus the RACH can be modelled as multiple slotted-ALOHA contention channels. In order to stabilize the access to the RACH a backoff algorithm should be used.

Thus the considered resource allocation policy in the uplink is a demand assignment mechanism which implies that entities must obtain permission by sending a request through a slotted-ALOHA channel before gaining access to transmission resources. This strategy is known as MD-PRMA++ in the literature [30].

3.4.3 Physical layer model

We have not integrated an accurate model of the physical layer in our system-level simulator due to the required computational complexity. Therefore, separate link layer simulation should be performed in order to determine the performance of the physical layer coding, spreading and modulation schemes.

We have considered the Physical link as a number of resource units (RU) representing the combination of one 16 chip DS-CDMA code in one time-slot. Since the spreading codes are OVSF, the allocation of two 16-chip codes on one time-slot, the allocation of one 16-chip code on two time-slot and the allocation of one 8-chip code on one time-slot is considered to be equal to two RU allocation. The useful information per RU depends on the coding and rate matching scheme. We have considered 30, 20 and 10 bytes per RU for uncoded, 1/2 and 1/3 coding schemes respectively. The number of available resource units for uplink and downlink depends on the number of time-slot assigned to each link, and are handled by the radio resource management model.

In addition, a 2-state Markov modulated Bernoulli process have been considered to model RLC block corruption due to wireless transmission. This model can reasonably emulate error bursts as long as a sustained BER in the radio channel.

4 SIMULATION RESULTS

4.1 Simulation Scenarios

As long as several parameters can be adjusted for each of the model included in our simulator, the total number of simulation parameters is high. In this section we will describe the main simulation parameters used in the simulations corresponding to the results shown in the next section.

The system has been considered to be serving only data users, so that 13 time-slot are allocated to the service. As long as the traffic generated is asymmetric, a total of 11 time-slot are reserved to the downlink access and 2 time-slot are reserved to the uplink. An additional uplink time-slot is assigned to the RACH, and up to 12 codes may be used for random access. The Pseudo Bayesian Broadcast [30] back-off mechanism is applied to stabilize the random access. The coding scheme rate is 1/3, so that the number of available bytes per resource units is 10. Radio channel errors have been characterized as a Bernoulli process with a variable bit error rate (BER).

In order to determine which RLC mode should be used we have compared the benefits obtained using the RLC acknowledged mode with standard TCP and the unacknowledged mode with the Snoop enhancement. The size of the RLC block in the acknowledged mode has been set to 20 bytes and the block header size is considered to be 3 bytes.

As previously mentioned, TCP configuration can strongly determine the achieved performance. The TCP version considered in these simulations is TCP NewReno with Limited transmit [19]. The maximum segment size (MSS) is 536 bytes, the maximum congestion window size is 100 MSS, and the timer granularity is 200 ms.

According to [27] losses and mean delay are the main parameters of the Internet model that affects TCP performance. In the simulations shown in this paper no losses and a constant delay have been considered in the Internet. In order to determine the effect of the mean Internet delay on TCP performance in the presence of wireless losses, different mean delays have been simulated.

In this paper a finite population of web users is considered. Each connected user is represented by a session which last the whole simulation. A fixed number of simultaneous web sessions have been simulated. The total number of web users have been set to 140.

4.2 Performance Analysis

Figures 7, 8, 9 and 10 show the performance of the different RLC modes and the Snoop enhancement for different Internet delays. The solid line correspond to the RLC acknowledged mode performance, the dotted line correspond to the RLC transparent mode performance, and the dashed one represents the Snoop protocol performance. The performance is analysed in terms of connection *Goodput*, defined as the ratio of connections size to connections duration (bytes per second).

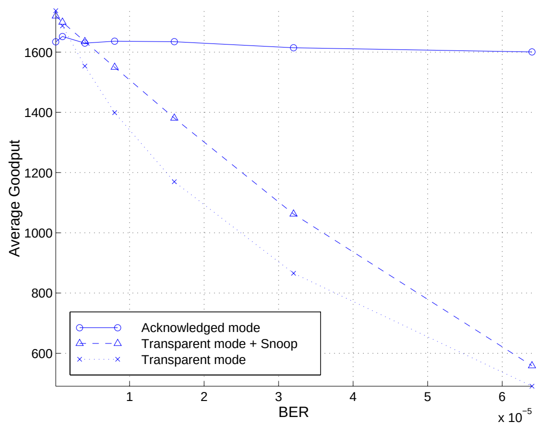

Figure 7: Connection Goodput vs error rate for 200 ms Internet mean delay

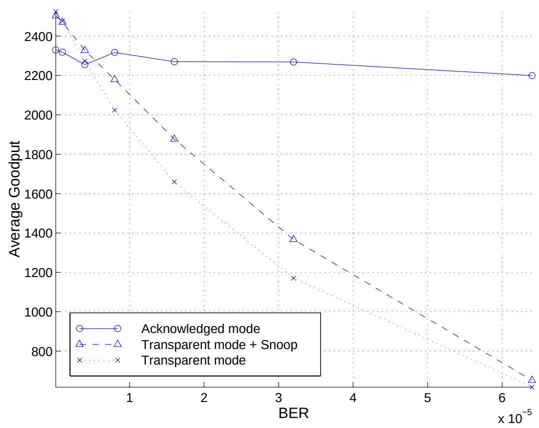

Figure 8: Connection Goodput vs error rate for 100 ms Internet mean delay

Simulations show how corruption losses in the radio link affect TCP performance when the transparent mode of RLC is used. The acknowledged mode is capable of maintaining TCP performance at higher error rates, but

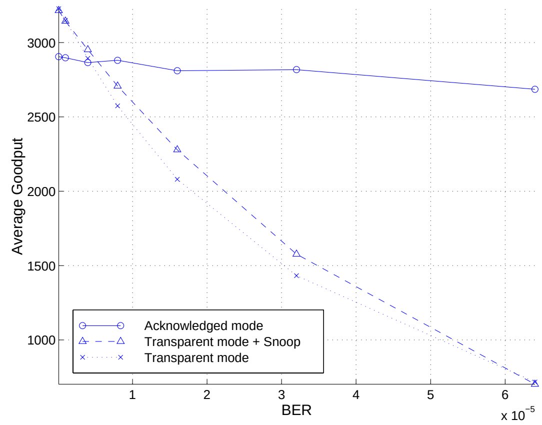

Figure 9: Connection Goodput vs error rate for 50 ms Internet mean delay

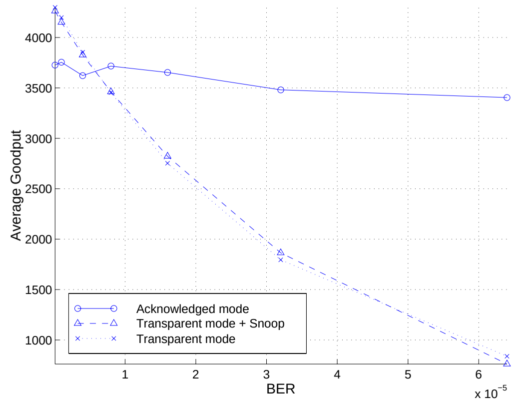

Figure 10: Connection Goodput vs error rate for no Internet delay

obtains a lower performance at low error rates. However, the RLC acknowledged mode overhead is greater, which can led to a loss of capacity at higher system loads.

On the other hand, Snoop protocol improves TCP and achieve a better performance than the RLC acknowledged mode at certain error rates. The benefits of Snoop in TCP performance are greater as the Internet latency is higher, because packets are retransmitted locally by the Snoop agent whereas packet retransmitted by the sender must traverse the Internet

5 Conclusions and Future Work

Simulations are a crucial tool to test communication networks performance before deploying them. This paper discusses the modelling of the different protocol layers of a wireless Internet access system in order to carry out end-to-end performance estimations. The discussion includes the radio interface protocols modelling as well as the relevant Internet protocols and the involved application. The use of a realistic traffic model, capable of reproducing the main features of the application traffic is a key issue to achieve reliable simulations. TCP is a complex protocol which interacts both with the network and with the application, so that traffic sources are affected by the network conditions. In order to capture this effect, a TCP model is included in the traffic generator. Internet delay and losses are also considered, since they affect

TCP performance.

Further, several simulation that show how corruption losses degrade TCP performance have been carried out. We have compared the Snoop TCP enhancement to the RLC acknowledged mode. The results demonstrate that Snoop protocol improves TCP performance and achieves a better performance than the RLC acknowledged mode only at low error-rates. This enhancement is higher when internet delay increases. On the other hand, the RLC acknowledged is capable of maintaining TCP performance at high BER.

The developed simulator allows the evaluation of the UTRA/TDD system performance from the user point of view. A detailed physical layer model has not been used due to the heavy computational effort required to integrate it with intensive protocol simulations. Separate physical layer simulations should be performed to evaluate the performance of the coding, interleaving and modulation schemes at different conditions. As a future work, the results of such physical layer simulations should be integrated in our simulator to improve physical layer modelling. Thus, the benefits of adapting the different coding schemes to the channel conditions may be explored.

6 ACKNOWLEDGEMENTS

This paper has been supported by the Spanish CICYT, under contract TIC2000-1120-C03-01, and Alcatel España.

REFERENCES

[1] R. Kalden, I. Meirick, and M. Meyer. Wireless internet access based on GPRS. *IEEE Personal Communications*, pages 8–18, April 2000.

[2] H. Balakrishnan, V. N. Padmanabhan, S. Seshan, and R. H. Katz. A comparison of mechanisms for improving TCP performance over wireless links. *IEEE/ACM Transactions on Networking*, 5(6):756–769, 1997.

[3] K. Claffy, G. Miller, and K. Thompson. The Nature of The Beast: RecentTraffic Measurements from an Internet Traffic Backbone. In *INET 98*, 1998.

[4] Network architecture. Technical Specification 25401, 3GPP, 2000.

[5] Radio interface protocol architecture. Technical Specification 25301, 3GPP, 2000.

[6] W.E. Leland, M. S. Taqqu, W. Willinger, and V. Wilson. On the Self-Similar Nature of Ethernet Traffic (extended version). *IEEE/ACM Transactions on Networking*, 2:1–15, 1994.

[7] M. Crovella and A. Bestavros. Self -Similarity in World Wide Web traffic: Evidence and possible causes. Philadelphia, Mayo 1996.

[8] Jon M. Peha. Protocols can make traffic appear Self-Similar, 1997.

[9] ETSI. Selection procedures for the choice of radio transmission technologies of the UMTS. Technical Report TR 101 112 v3.1.0, ETSI, 1997.

[10] A. Arvidsson and P. Karlsson. On traffic models for TCP/IP. *ITC*, 16, 1999.

[11] L. Guo, M. Crovella, and I. Matta. TCP Congestion Control and Heavy Tails. Technical Report BUCSTR -2000-017, Computer Science Dept., 2000.

[12] A. Veres and M. Boda. The Chaotic Nature of TCP Congestion Control. In *INFOCOM (3)*, pages 1715–1723, 2000.

[13] A. Reyes-Lecuona. *Modelado de Tráfico de Clientes WWW*. PhD thesis, E.T.S.I.T. Universidad de Málaga, 2001.

[14] V. Jacobson. Congestion avoidance and control. *ACM SIGCOMM-88*, Augus 1988.

[15] A. Falk and M. Allman. On the effective evaluation of TCP. *ACM Computer Communication Review*, 5(29), 1999.

[16] J. Padhye and S. Floyd. Identifying the TCP behavior of web servers. Technical Report TR-01-002, International Computer Science Institute, 2001.

[17] M. Allman, V. Paxson, and W. Stevens. TCP congestion control. RFC 2581, IETF, 1999.

[18] J. C. Hoe. Improving the start-up behavior of a congestion control sheme for TCP. 26,4:270–280, 26–30 1996.

[19] S. Floyd and T. Henderson. The NewReno modification to TCP's fast recovery algorithm. RFC 2582, Abril 1999.

[20] M. Mathis and J. Mahdavi. Forward acknowledgement: Refining TCP congestion control. pages 281–291, 1996.

[21] S. Floyd and V. Paxson. Difficulties in simulating the internet. *to appear in IEEE/ACM Transactions on Networking*, 2001.

[22] J. Bolot. Characterizing End-to-End Packet Delay and Loss in the Internet. *Journal of High-Speed Networks*, 2:305–323, 1993.

[23] Q. Li and D. L. Mills. On the long-range dependence of packet round-trip delays in internet. *Proc. of IEEE ICC'98*, 2:1185–1191, 1998.

[24] V. Paxson. *Measurement and Analysis of End-to-End Internet Dynamics*. PhD thesis, University of California, Berkeley, Abril 1997.

[25] M. Yajnik, S. Moon, J. Kurose, and D. Towsley. Measurements and modelling of the temporal dependence in packet loss. *Infocom99.*, 1999.

[26] Y. Zhang, V. Paxson, and S. Shenker. The stationarity of internet path properties: Routing, loss, and throughput. Technical report, ACIRI, 2000.

[27] E. González-Parada, J. Cano, J. Aguilar, A. Reyes-Lecuona, and A. Diaz-Estrella. Modelado de Internet para simulación con modelos de tráfico que incorporan el protocolo TCP. In *III Jornadas de Ingeniería Telemática*, 2001.

[28] S. Moon, J. Kurose, P. Skelly, and D. Towsley. Correlation of packet delay and loss in the internet. Technical report, Department of Computer Sciencie, University of Massachusetts, 1998.

[29] S. Floyd and V. Jacobson. Random Early Detection gateways for congestion avoidance. *IEEE/ACM Transactions on Networking*, 1:397–413, 1993.

[30] A. Brand and A. H. Aghvami. Multidimensional PRMA with prioritized bayesian broadcast: A MAC strategy for multiservice traffic over UMTS. *IEEE Transactions on Vehicular Technology*, 47(4):1148–1160, 1998.