Silke Heier,
Dirk Heinrichs, Andreas Kemper

ComNets
RWTH Aachen University
Kopernikusstr. 16
D-52074 Aachen
Tel: +49 (0)241 80 279 28
Fax: +49 (0)241 80 222 42
e-mail: she@comnets.rwth-aachen.de
WWW: http://www.comnets.rwth-aachen.de/~she

IP BASED SERVICES AT THE UMTS RADIO INTERFACE

Paper submitted to the Third International Conference on 3G Mobile Communication Technologies

Authors: Silke Heier, Dirk Heinrichs, Andreas Kemper

Keywords: UMTS, UTRA, Radio Interface Protocols, WWW Traffic Model, QoS, TCP/IP, UMTS Radio Interface Simulator

Address: RWTH Aachen
Communication Networks
Kopernikusstr. 16
52074 Aachen, Germany

Tel.: +49 241 80 279 28
Fax.: +49 241 80 222 42
e-mail: she@comnets.rwth-aachen.de

WWW: http://www.comnets.rwth-aachen.de/~she

IP BASED SERVICES AT THE UMTS RADIO INTERFACE

S. Heier, D. Heinrichs, A. Kemper

RWTH-Aachen, Germany

***Abstract** – Universal Mobile Telecommunications System* (UMTS) as a 3rd generation mobile telecommunications system is designed to provide mobile Internet access to the user. In this paper, E-Mail and *World Wide Web* (WWW) browsing sessions as mobile data service applications are examined for performance analysis by simulations. A WWW traffic model is presented as the basis of a WWW load generator. This traffic load generator integrated into an event driven, bit accurate *UMTS Radio Interface Simulator* (URIS) is presented. The simulator is based on a detailed implementation of the standardized UMTS protocol stack extended by the *Transmission Control Protocol* (TCP) and the *Internet Protocol* (IP). Simulation results of quality of service parameters like delay and throughput depict the performance of mobile Internet access over UMTS.2

INTRODUCTION

In the context of exponential growth of the Internet market, growing demand of data services is also expected for mobile users. Third generation mobile radio networks like *Universal Mobile Telecommunications System* (UMTS) will provide data services with higher data rates than presently available in second generation cellular systems. UMTS will support bit rates of up to 144 kbit/s in rural areas, 384 kbit/s in hotspots and up to 2 Mbit/s in indoor scenarios. Hence, the mobile Internet access will be one of the key services which should guarantee a successful start of UMTS operation.

Due to the nature of radio access systems different conditions namely limited bandwidth, higher delays and error rates at the radio interface have to be taken into account compared to fixed network access. Thus, a modelling of the WWW traffic as well as the UMTS radio interface is mandatory for performance evaluation purposes.

First, WWW browsing session traffic characteristics are outlined and mapped with identified parameter sets on a WWW traffic model. Following, the UMTS radio interface simulation environment called *UMTS Radio Interface Simulator* (URIS) is presented. This event driven, bit accurate simulator models the UMTS radio interface protocol stack in accordance with the UMTS protocol specifications of Release 4. The performance evaluation considers the detailed UMTS protocol implementation as well as the traffic load generation for WWW browsing sessions including *Transmission Control Protocol/Internet Protocol* (TCP/IP) modelling. Thus, it is feasible to present the performance of typical TCP applications like WWW and E-Mail in an unreliable mobile environment with higher error rates concerning its packet retransmission interaction on TCP and on radio link control level. Simulation results will show which performance in terms of quality of service parameters a mobile user will experience while surfing the Internet.

The simulation section presents the used simulation methodology and the simulation environment. Performance evaluation, simulation results and discussion is done afterwards. The paper concludes with a summary.

WWW TRAFFIC MODEL

Surfing the Internet is a WWW browsing session which is running on the TCP/IP protocol stack. A WWW traffic model is necessary for simulative examinations of the performance of data services of mobile radio networks. In the following, model parameters of a WWW application and its mathematical description for generating protocol specific traffic are presented.

Related documentation can be found in Arlitt and Williamson (3), Frost and Melamed (5), Paxson (8). The main part of the later implementation is based on the work of (3).

A typical WWW browsing session consists of a sequence of page requests. These pages contain a number of objects with a dedicated object size each. During a page request, several packets for each object may be generated which means that the page request constitutes of a bursty sequence of packets. The burstyness during the page request is a characteristic feature of packet transmission. After the page has entirely arrived at the terminal, the user is consuming a certain amount of time for studying the information. This time interval is called reading time. Hence, the following must be modelled:

- Session arrival process: The arrival of session set-ups to the network is modelled as a Poisson process. It is important to note that this process only generates the time instants when a session request begins and is independent of the session termination,
- Session length is modelled implicitly by the number of page requests during the session,
- Number of page requests per session,

TABLE 1- Model Parameters of WWW Browsing Session

WWW Parameter	Distribution	Mean	Variance
Session Arrival Rate [h^{-1}]	negative exponential	2.0	-
Pages per Session	geometric	5.0	-
Reading Time between Pages [s]	negative exponential	12.0	-
Objects per Page	geometric	2.5	-
Inter Arrival Time between Objects [s]	negative exponential	0.5	-
Page Request Size [byte]	Normal	1136.0	-
Object Size [byte]	$\log_2$-Erlang-k	$\log_2 2521 \approx 11.3$	$(\log_2 5)^2 = 5.4$

- Reading time between page requests: The reading time starts when the last packet of the page request was completely received by the user. The reading time ends when the user makes a request for the next page.
- Number of objects within a page request,
- Inter arrival time between objects,
- Size of an object.

TABLE 1 gives an overview of the distributions and their parameters describing the WWW traffic model.

UMTS RADIO INTERFACE MODEL

For this study, the simulation tool URIS was developed. This simulation environment is used to investigate, optimise and develop features of the radio interface protocol stack. In addition, it offers the opportunity of capacity and quality of service evaluation by simulations of various scenarios. The simulator is a pure software solution in the programming language C++. The simulation model is implemented with the help of a powerful C++ class library which was developed by the Chair of Communication Networks and is called the *SDL Performance Evaluation Tool Class Library* (SPEETCL), Steppler and Lott (9). This generic, object oriented library is well suited for telecommunication network simulation purposes and can be used in event driven, bit accurate simulation environments. The UMTS protocols at the radio interface enhanced by a TCP/IP protocol stack were specified with the *Specification and Description Language* (SDL).

The software architecture of the URIS simulator is shown in FIGURE 1. Up to now, the simulator consists of various traffic generators and a traffic load mixture unit which is used to adjust scenarios with desired load mixtures. The physical channel module models transmission of bursts in radio frames on the radio interface. This includes discarding of erroneous bursts depending on the error model.

The core of the simulator are the modules *User Equipment* (UE) and *UMTS Terrestrial Radio Access* (UTRA) which are built formally similar. Each single UE and UTRA is represented as an SDL system which contains the protocol implementation of the layers.

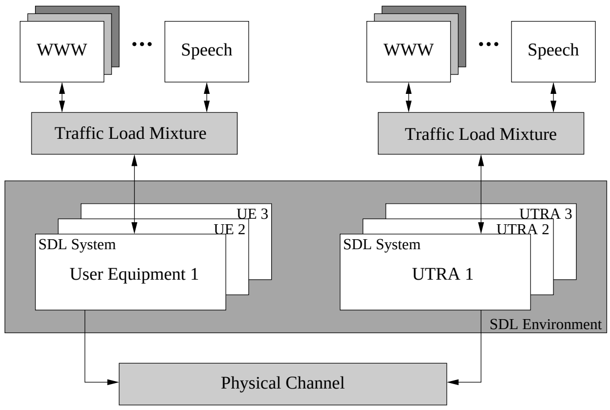

FIGURE 1 – Structure of the URIS Simulator

FIGURE 2 gives an overview of the protocol structure in a generic SDL system. The complex protocols like *Medium Access Control* (MAC), RLC, *Packet Data Convergence Protocol* (PDCP) and *Radio Resource Control* (RRC) based on UMTS Release 4 and TCP, *User Datagram Protocol* (UDP) and IP are specified in SDL. The modern Internet is mainly based on TCP and IP. In order to provide Internet services to the mobile user, the TCP/IP protocol suite has to be adapted to the UMTS protocols. The UMTS radio interface protocol stack offers this functionality by providing the PDCP.

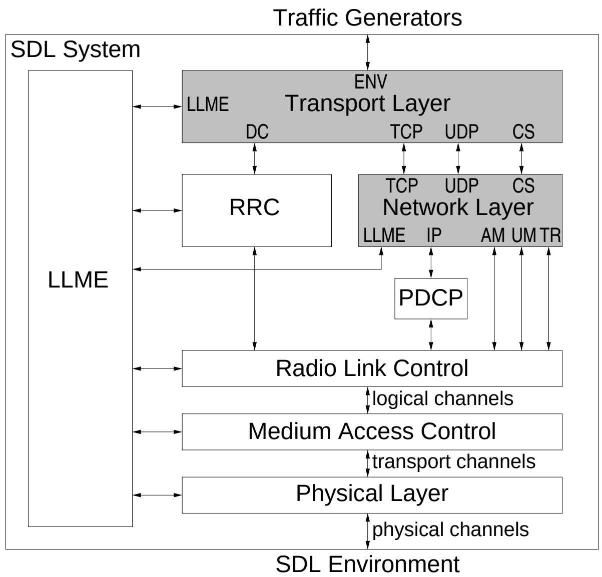

FIGURE 2 – URIS SDL System Structure

TCP provides overlying protocols with a reliable connection. The following services are offered:

- Retransmission of lost packets,
- Recovery from out-of-order delivery,
- Flow control.

The TCP implementation realized in URIS is based on the so called ”Reno” TCP stack and “SACK TCP”, Fall and Floyd (4), Mathis et al (7).

IP is a connectionless and unreliable protocol that operates on a best-effort basis. This means that IP packets may be lost, out-of-order or even duplicated without IP handling these situations.

The RLC realizes segmentation and retransmission services for both user and control data. The RLC protocol provides three different data transfer services to the adjacent layer:

- *Transparent data transfer service mode* (TR),
- *Unacknowledged data transfer service mode* (UM),
- *Acknowledged data transfer service mode* (AM).

The AM transmits higher layer data packets with guaranteed delivery to the peer entity. Therefore, it is mainly used for traffic of interactive class or background class type like WWW or E-Mail sessions. *Automatic Repeat Request* (ARQ) mechanisms are applied for correction of transmission errors. A Selective-Repeat ARQ has been implemented for AM. Segmentation and reassembling as well as concatenation are fully implemented as specified in 3GPP25.322 (1).

Usual simulator approaches model protocols and functions on basis of abstractions and simplifications. The aim of URIS is a detailed, bit accurate implementation of the standardized protocols. This offers the opportunity to determine the performance of UMTS in a realistic manner.

SIMULATION

The main parameters concerning *Quality of Service* (QoS) which are affected by RLC and TCP/IP, are delay and throughput. Two measurements were made during the simulations:

1. Packet Delay: Time from sending a traffic load packet to the TCP layer until correct reception of the packet by the traffic load receiver,
2. Packet Throughput: Size of the traffic load packet divided by the packet delay. The throughput is given in bits per second [bit/s].

Simulations of E-Mail and WWW browsing sessions running on TCP/IP in RLC AM were executed. The assumption has been made that a *Dedicated Channel* (DCH) of 64 kbit/s for E-Mail/WWW traffic resp. 128 kbit/s for WWW traffic is dedicated to the mobile user.

TABLE 2 - Simulation Parameter

Parameter	**Value**
TTI Length [s]	0.02
DCH Capacity [kbit/s]	64 / 128
RLC Tx Window Size [PDU]	1024
Status Prohibit Timer [s]	0.1
Poll Prohibit Timer [s]	0.13
Poll Timer [s]	0.1
Max. TCP Segment Size [byte]	512
Max. TCP Window [kbyte]	16
Min./Max. TCP Retrans. Timeout [s]	3 / 64
Block Error Rate [%]	0, 1, 2, 5, 10

The corresponding RLC configuration for this channel capacity is taken from the technical specification document 3GPP34.108 (2) and is shown together with the TCP parameters in TABLE 2.

The cumulative distribution function of the measured packet delay and the complementary cumulative distribution function of the measured packet throughput have been calculated. Measurements have been made for increasing transport block error rates from 0 % up to 10 %. A transport block error is a missing RLC PDU caused by not recoverable errors after error detection, correction and checking by the physical layer.

E-Mail Simulation Results

The simulations with traffic load packets of fixed size were made to give an idea of the time a user has to wait for downloading an E-Mail. Packet sizes of 10 kbyte were used to model typical text E-Mails. Packet sizes of 100 kbyte or 1 Mbyte model the download of E-Mails with bigger attachments. A channel capacity of 64 kbit/s was used.

The results given in TABLE 3 show the delay and throughput dependent on packet size and block error rate. Included into this measurements is the TCP connection establishment, since it is part of the necessary protocol overhead using these services. As can be seen the necessary transmission time for small traffic load packets of 10 kbytes is neglectable. Delays of 2 s can be easily accepted. The throughput is relatively low compared to the reached throughput with bigger packets because a fixed time for the TCP connection establishment is necessary, which lowers the effective throughput for the whole transfer. Higher error rates of 10 % result in increasing delays up to doubled values. With bigger load packets the throughput increases continuously because the time for TCP connection establishment is getting less important for the results. With big load packets, higher block error rates have less impact on the results than with small ones, but with a block error rate of 10 % the throughput decreases by approximately 20 %. The maximum throughput reaches 52.98 kbit/s of the 64 kbit/s channel.

TABLE 3 - E-Mail Simulation Results

Packet Size [kbyte]	BLER [%]	Delay [s]	Throughput [kbit/s]
10	0	1.70	47.06
10	1	1.70	47.06
10	5	2.20	36.36
10	10	2.14	37.38
100	0	15.20	52.63
100	1	15.26	52.42
100	5	16.14	49.56
100	10	19.16	41.75
1000	0	151.00	52.98
1000	1	151.04	52.97
1000	5	165.42	48.36
1000	10	182.26	43.89

WWW Browsing Simulation Results

First, simulations results for two WWW browsing sessions per hour on a 64 kbit/s channel are presented. Since the total amount of data of a requested page is characterized by a high variance, this traffic load leads to high delays (FIGURE 3), which possibly reach a scale of 10 s to 100 s. The minimum delay for user data packets is 0.04 s.

This minimum delay can only be reached by small packets, which can be transferred within one *Transmission Time Interval* (TTI) and without queuing time. But since most of the user data packets are too big being transferred during one TTI, the probability for such low delays is very small. The vast majority of data has a certain queuing time, leading to higher delays and lower throughput. 50 % of all page objects will have a delay of less than 1.5 to 3.5 seconds.

FIGURE 4 shows that a maximum throughput of 53.6 kbit/s is reached, but only 20 % of all data packets reach a throughput of 40 kbit/s and above. The lowest throughput values of up to 10 kbit/s are caused by very small data packets and by data packets with a long queuing time.

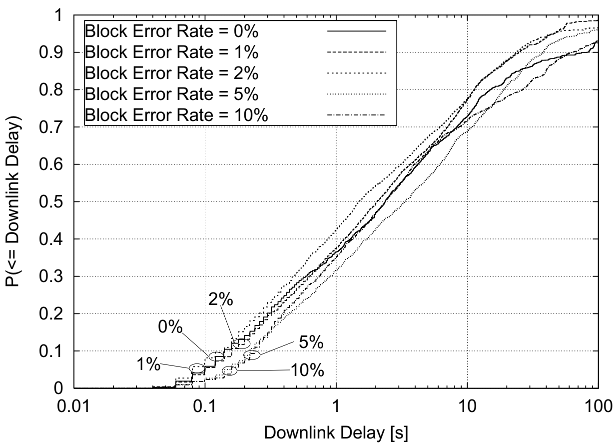

FIGURE 3 - WWW Downlink Delay, 64 kbit/s

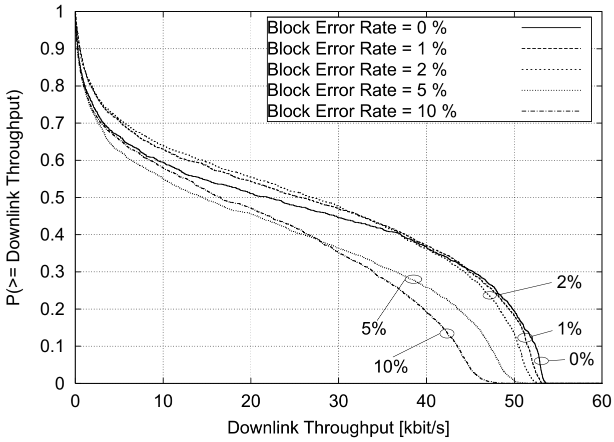

FIGURE 4 - WWW Downlink Throughput, 64 kbit/s

The maximum throughput reaches 84 % of the available channel capacity, the rest is overhead and padding generated by TCP/IP and RLC layer. Related to the maximum reachable performance considering RLC and TCP header, a channel usage of 97.6 % is reached.

Simulations on a 128 kbit/s DCH were executed as well. Again, maximum packet delays reach a scale of up to 100 s but 90 % of all user data packets are transmitted within 7 – 18 s or less, dependent on the error rate. Compared to the range of 25 – 90 s at a channel capacity of 64 kbit/s, this is a remarkable improvement.

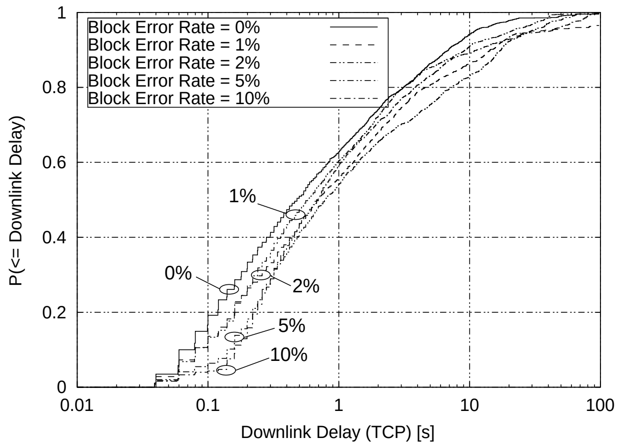

FIGURE 5 – WWW Downlink Delay, 128 kbit/s

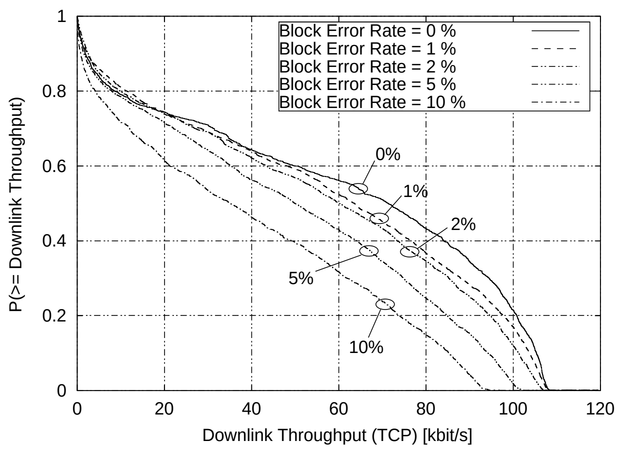

FIGURE 6 – WWW Downlink Throughput, 128 kbit/s

FIGURE 6 shows, that a maximum throughput of 109 kbit/s is reached, which is equivalent to 85.5 % of the available channel capacity. This maximum throughput is only slightly better compared to the 64 kbit/s scenario.

For block error rates of 5 % and 10 % delays are higher and the corresponding throughput is lower since every page object is more often affected by errors which results in time consuming retransmissions at the RLC layer.

In both WWW simulation scenarios even more delay is caused by queue waiting times. While the transmission of the first object of a WWW page is started, an immediately following object has to wait until the previous one has been transmitted.

CONCLUSION

This paper presents a simulator which can be used to model the protocol stack and service behaviour at the UMTS radio interface in a realistic manner. Due to the detailed and standard related implementation, the simulator can be used to evaluate the performance to be expected by real systems. In a second step the simulator can be used to optimise performance or to develop advanced protocol algorithms.

Running Internet applications over TCP and RLC AM will be an ordinary scenario during the introduction of UMTS. First terminals will rely on standard applications which will be adopted from the fixed world. Hence, TCP will run end-to-end including the radio interface.

This paper shows that this solution is applicable to satisfy the mobile user. The simulations have shown, that delays of WWW objects can be reduced by ⅔ when the channel capacity will be doubled from 64 kbit/s to 128 kbit/s. During these simulations, TCP retrans-missions have rarely occurred. The Selective-Repeat ARQ error correction of the RLC layer is sufficient to provide a fast and error free data transmission to the TCP layer. Therefore, the TCP protocol is able to operate with full performance. No congestion is assumed, so there is no need for the TCP protocol to reduce the size of the transmission window. Nevertheless the mobile user will not experience the same performance like surfing at a fixed access. Due to the unreliable radio link and its higher error rates, the channel capacity is reduced by a huge amount of necessary retransmissions on the RLC layer. A higher delay of a WWW object and a longer download time for a whole WWW page is the result due to longer queue waiting times.

It has to be mentioned that effects of the core network are not included in our simulations. The overall performance will get worse since TCP as a end-to-end protocol will experience additional delays and congestion in the core network. The mobile user will face a reduced throughput and longer waiting times for WWW page downloads than in fixed networks.

ACKNOWLEDGEMENT

Authors would like to thank Professor B. Walke (RWTH-Aachen, Germany) for advice and colleagues at ComNets for improving the UMTS simulator.

REFERENCES

1. 3GPP TS 25.322, June 2001,
   Radio Link Control (RLC) Protocol Specification,
   Technical Specification V4.1.0, Release 4,
   3rd Generation Partnerschip Project,
   Technical Specification Group Radio Access Network.

2. 3GPP TS 34.108, September 2001,
   Common Test Environments for User Equipment (UE) Conformance Testing,
   Technical Specification V4.0.0, Release 4,
   3rd Generation Partnerschip Project,
   Technical Specification Group Terminals.

3. Martin F. Arlitt, Carey L. Williamson, July 1995,
   A Synthetic Workload Model for Internet Mosaic Traffic, Proceedings of the 1995 Summer Computer Simulation Conference, Ottawa, Canada, pp. 24-26.

4. K. Fall, S. Floyd, July 1996,
   Simulation-based Comparisons of Tahoe, Reno, and SACK TCP, Lawrence Berkeley National Laboratory.

5. Victor S. Frost, Benjamin Melamed. March 1994,
   Traffic Modeling For Telecommunications Networks, IEEE Communications Magazine, pp. 70-81.

6. Harri Holma, Antti Toskala, March 2001,
   WCDMA for UMTS: Radio Access for Third Generation Mobile Communications, John Wiley & Sons Ltd..

7. M. Mathis, J. Mahdavi, S. Floyd, A. Romanow, April 1996, Selective Acknowledgement Options, RFC2018, Internet Engeneering Task Force.

8. Vern Paxson, August 1994, Empirically-Derived Analytic Models of Wide-Area TCP Connections, IEEE/ACM Transactions on Networking, 2 (4), pp. 316-33.

9. Martin Steppler, Matthias Lott, September 1997,
   SPEET - SDL Performance Evaluation Tool,
   8th SDL Forum '97.

10. B. Walke, Mobile Radio Networks, 2001,
    John Wiley & Sons Ltd., Chichester, England, 2nd Edition.