A Demonstrator for Real-time Multimedia Sessions over 3rd Generation Wireless Networks

S.Gruhl*, A.Echihabi #, T.Rachidi #, M.Link*, and M.Söllner*

▪* Bell Laboratories, Lucent Technologies
Thurn-und-Taxis-Strasse 10
90411 Nuremberg, Germany

#Alakhawayn University
Ifrane, 53000
Morocco

Abstract -**This paper presents a simulation tool used to demonstrate the effects of 3rd generation wireless network access on IP-based multimedia applications. In 3rd generation wireless networks like, e.g., the Universal Mobile Telecommunications System (UMTS), complex control and transport mechanisms influence the link quality for a given application, while the application itself influences the control mechanisms by generating load on the wireless link. To study the effects of such a dynamic system on the user-perceived Quality of Service (QoS) of a real-time multimedia application it is necessary to simulate the wireless link in real-time. Our tool allows to interface standard IP-based multimedia applications with an UMTS-specific link simulator. The system comprises essential layer 2 and 3 protocol functionality of the UMTS Terrestrial Radio Access Network (UTRAN) for Terminal Equipment (TE) and Base Station (BS). We motivate and describe the concept, system architecture and implementation in terms of timing, QoS mapping, packet- and circuit-switched (CS) services integration, and physical link parameters.**

I. Introduction

UMTS, as a 3rd generation wireless network, will provide a wide range of multimedia services at data rates up to 2 Mbit/s [6]. In addition to the traditional circuit-switched services found in second-generation networks for voice transmission, UMTS will offer packet switched data services. The statistical gain of packet switching results from increased link utilization due to non-continuous bandwidth requirements from applications. Such mechanism is well investigated in wireline networks such as ATM, X25, or WAN/Internet. It is also currently being introduced in packet service extensions of second generation wireless networks such as (E-)GPRS for GSM networks [10]. These are expected to interface seamlessly with traditional higher level protocols, namely the TCP/IP suite. This will also hold for 3rd generation wireless networks, while there will be increased demand for QoS guaranties, which are necessary for multimedia applications [4] in particular. However, provisioning for CS-like QoS guaranties using packet switched services may turn out to be a challenging task in an environment where link capacity changes frequently and where users move unpredictably. It is rather unsound to make assumptions based on the study of wireline networks and to use them as a basis for predicting the behavior of wireless networks such as UMTS. Major differences exist between wireline and wireless networks networks, since the radio link constitutes a massive bottleneck:

- Power limitation, interference and altering radio link conditions due to mobile terminal position cause the link capacity to change rapidly.
- Handoff calls lead to additional and unpredictable load in a cell.
- High bit error rates (BER) are encountered in wireless communications. Forward error correction (FEC) becomes important and retransmissions are more frequent compared to fixed networks. Consequently, data rates are affected.

It is difficult to predict the impact of the aforementioned differences between wireline and wireless networks on performance and on QoS. We identify a set of challenging tasks in the field of wireless network performance, which to address we have implemented a real-time testbed:

- How successful does a certain QoS enabling technology perform with a given application?
- What will user-perceived QoS be like over a future wireless link?
- What are the traffic arrival characteristics of a certain type of future service, e.g., a video real-time session?

A. *Control Loops*

To illustrate what we consider as protocol performance in this context, we reference some studies of dynamic TCP/IP behavior and its performance over erroneous/slow links. TCP has TCP-flow-control as an adaptation mechanism for network congestion situations. It acts on packet loss and adapts its sending rate. For lossy links, as experienced in wireless transmission, this mechanism can show unexpected behavior and perform poorly [11][12]. We view these investigations as one example of undesired control loop interaction in a multi-protocol-layer wireless environment. We are interested in control loops in general, e.g., bandwidth adaptation methods in a scalable video codec. Our approach will also allow us to address TCP performance investigations, but is not restricted to this.

In the worst case we expect to find oscillation effects that may arise due to the various dynamic controls that will be applied simultaneously by higher and lower level protocols. These mechanisms are found in lower layers to overcome lossy and variable capacity links, and in higher layers specifically to adapt to changing network capacity due to congestion situations.

In particular, we have looked at adaptive video codecs on top of the Real-time Transport Protocol (RTP), which run their independent control loops. This can negatively interact with other adaptive control loops from a wireless network, in a way comparable to the TCP phenomenon. There can be many control loops as, e.g., Power Control to adjust the transmit power on the radio link which effects error rates or control performed by the Radio Resource Manager (RRM) on Resource Allocation. These interactions, if not investigated and understood, can hinder the many benefits expected from future wireless communications networks.

B. *Control Loops and QoS*

Future packet switched networks will have to incorporate QoS enabling techniques to address specific application requirements. Imagine a guaranteed bandwidth link with a low error rate. Then you will find most of the TCP-flow control problems to be softened by a large degree. How does this work for other control loops? What are the QoS requirements that should be satisfied? How sensitive is this approach to wireless intrinsic link variation or actions like a handover to another BS?

With our approach we will be able to investigate the benefits of a QoS enabled system. As we offer a real-time real-application interface, these effects will be accessible for measurements. Furthermore, our system will allow demonstrating user perceived QoS over a simulated UMTS network.

C. Real-time Simulation

Traditional off-line simulators are not suited for our type of investigation unless all protocol layers including the adoption mechanisms of the application, e.g., an adaptive video codec, are available in a simulation model. Furthermore, traffic models are required which are difficult to define for future services.

For many higher layer protocols and applications we lack the full description of internal control mechanisms, but often we find them as implemented applications. Most applications today communicate via IP traffic. We use this transparent standard IP-interface to transport application data seamlessly in and out of our real time simulation of a third generation network. Thus standard IP-based applications such as FTP, web browsing, video applications or Microsoft Netmeeting™ can be simulated concurrently over the simulated UMTS protocol stack and radio link (see Fig. 1). Our approach links existing applications during run-time with our prototype implementation of a UMTS protocol stack and with our PHY-layer simulator. Thus we can run our investigations without the need to model the application/higher layer part. By using the real application we do not need any traffic models assumptions, but use the real dynamic behavior - including user interaction - for our investigations. Furthermore, we can trace these scenarios for off-line investigations. The main advantage is that our traces incorporate many dynamic effects of a real system.

D. Evaluation of Today's non QoS-aware Applications

The testbed is capable of accepting an arbitrary number of concurrent TCP/UDP flows. In future wireless systems these flows will have individual QoS-requirements.

Although there are QoS protocols for IP networks, such as the Resource Reservation Protocol (RSVP), these QoS negotiation protocols are rarely found in present applications. Therefore we have developed the concept of individually assigned QoS for individual flows. A flow is distinguished by its sending/receiving port number. This allows us to do investigations with a simulation of QoS enabled networks. The QoS requirements are for example for reliability, delay, jitter, and guaranteed bandwidth.

The main advantage of an on-line simulation system is the possibility to demonstrate the user-perceived quality of service. We can investigate the effects of the various control mechanisms, altered radio conditions or a congestion situation in a cell.

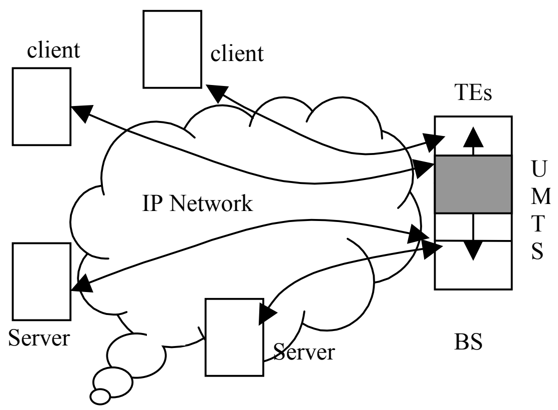

Fig. 1. Standard Client/Server IP applications run through the UMTS protocol stack (shown in shade).

The remainder of the paper is organized as follows: section II summarizes key UMTS services and functionality implemented. Section III describes the testbed architecture. Section IV is concerned with implementation issues such as the time frame, QoS mapping and Radio blocks scheduling. Section V describes the ongoing works that use the testbed.

II. UMTS Protocol Stack

The UMTS Radio Interface architecture is layered into a physical (PHY) layer, a data link layer and a network layer (IP in this work). The data link layer is divided into a Radio Link Control (RLC) sublayer and a Medium Access Control (MAC) sublayer. The Logical Link Control (LLC) sublayer present in many early UMTS proposals is not considered. It is expected that this sublayer will be reduced to a Null-sublayer to minimize protocol overhead and/or is merged with the Packet Data Convergence Protocol (PDCP) Layer, which is doing header compression. PDCP is currently modeled by accounting for its compression on IP-packet length only. Fig. 2 shows protocol termination for a UMTS dedicated channel (DCH). The following is a summary of the main services and functions of Layer 1 and Layer 2 [1] that have been considered in the system:

PHY Layer Services:
- Information transfer services to higher layers through transport channels

Physical Layer Functions:
- Error detection and FEC on transport channels
- Multiplexing and de-multiplexing of transport-channels/coded-transport composite channels
- Synchronization
- Measurements and indication to higher layers

MAC Sublayer Services:
- Data transfer
- Radio resources and MAC parameters reallocation
- Measurements reporting

MAC Sublayer Functions:
- MAC-scheduling and selection of an appropriate transport format
- Multiplexing and de-multiplexing of higher layer PDUs into/from transport frames delivered to/from the PHY layer
- Traffic volume monitoring and reporting to RRM

RLC Sublayer Services:
- Transparent data transfer
- Acknowledged or unacknowledged data transfer
- QoS settings

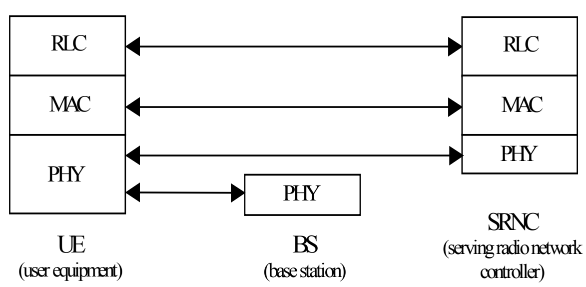

Fig. 2. Protocol termination for UMTS user plane DCH.

RLC Sublayer Functions:

- User data transfer
- Segmentation and reassembly
- Padding
- In-sequence/out-of-sequence delivery of higher layer PDU
- ARQ - backward error correction and flow control

UMTS provides besides other services a reliable RLC mode to be used by non-real time or interactive applications. Based on FEC using various channel coding schemes, the RLC ARQ mechanism recovers received erroneous radio blocks.

As can be seen, most UMTS layer 1 and 2 connectionless services and functions have been considered. Functions for connection establishment on the other hand have been ignored. This assumption is valid for the scope of our project, as we are only interested in the dynamic system behavior of an established link.

III. System Architecture

The testbed is composed of mainly of three modules: **1.** An IP proxy server allowing standard applications to connect. **2.** The emulation of the essential parts of the UMTS protocol stack for layer 2/3. **3.** A module for the PHY-link simulation responsible for error injection based on a scenario driven radio link simulation. Fig. 3 shows the basic interfacing concept. The applications are located in the TE and somewhere in the "network". To distinguish between uplink (UL) and downlink (DL) direction we will in future refer to this as BS entity for the sake of simplicity. These are interfaced via IP to the proxy, where the IP data is brought to our UMTS stack and internally to our PHY-layer simulation.

A. TCP/IP Proxy

The proxy provides the IP-interface allowing existing TCP/IP and UDP/IP client applications to connect through predefined SAPs/ports (Service Access Points). Corresponding connections are opened/initiated by the proxy server to predefined application servers using standard SAPs/ports (see Fig. 4.). Incoming data from the client/TE are relayed through the UMTS protocol stack to the server/BS and vice versa. The same predefined Quality of Service (QoS) are assigned to both flows, i.e., to the flow of data from the client/TE to the server/BS and to the flow of data from the server/TE to client/TE. Four QoS classes are defined, namely: Conversational, Streaming, Interactive and Background [4]. These classes and their parameters such as bit rate, jitter bound etc. are passed on to the UMTS protocol stack.

Besides the IP interface for packet switched data services we have designed a UDP-based interface for a circuit switched system emulation. Here we send the real-time-data by UDP from our proprietary adaptive multi rate (AMR) speech codec (or similar) to the proxy. A special port removes the UDP header and injects the plain user-bits to the PHY-layer module. Thus CS-services will not find any kind of protocol overhead, but naturally also no ARQ protection. The results are transmitted via UDP to a proprietary receiver to play the CS-emulated service.

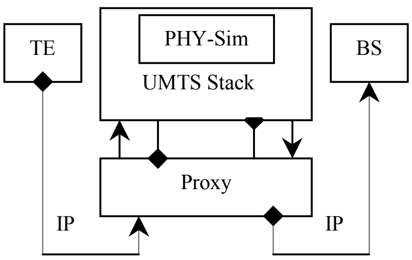

Fig. 3. Application interfacing concept via IP.

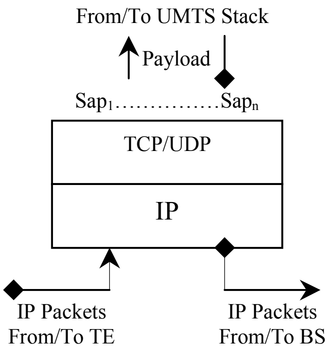

Fig. 4. Proxy architecture.

B. UMTS Protocol Stack

The UMTS protocol stack provides the UMTS data services and functions described in previous section. The interaction of our main modules implementing the essential parts of the protocol stack is given in Fig. 5 for the UL direction. Each of the functions and services of layers 1 and 2 are provided by independent entities, which are described hereafter:

RRM Module:

Signals the available Transport Format Combinations (TFC) to scheduling. Sets power budget for packet and circuit switched services.

DL Scheduler Module:

Under power budget considerations, which are limited by resource allocation from RRM, flows are scheduled with regard to QoS requirements. These are reflected by a bandwidth conserving scheduling discipline.

Due to the flexible Transport Format (TF) an arbitrary number of Transport Blocks (TB) is combined into a transport block set (TBS) and passed on to the PHY module every MAC scheduling interval (typically 10 ms).

UL Scheduler Module:

Currently a simple Round Robbin scheduler is assumed for each TE.

Segmentation Module:

Standard mechanism for segmentation on RLC level; padding is optional.

ARQ Module:

Where ARQ is applied it is put into the RLC-blocks. Received ACK/NACK signals are passed to the module from the data flow in the other direction (UL/DL).

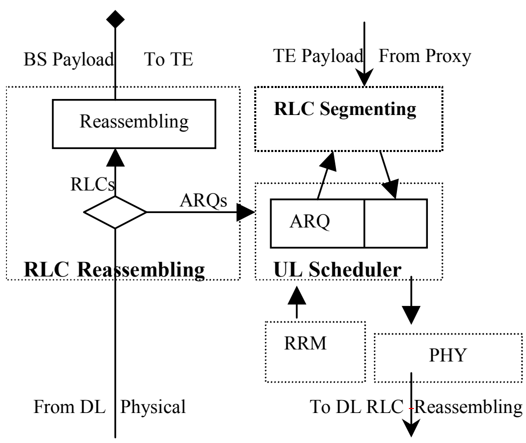

Fig. 5. UMTS (UL) protocol stack modules.

Reassembly Module:

Decodes the TBSs and reassembles the received RLC blocks after eventually stripping padding data. Depending on the flow settings (derived from QoS requirements), ARQ is performed, i.e. ACKs/NACKs are periodically delivered to the scheduler module in the opposite direction. By travelling the same path as regular data, ACKs/NACKs are therefore itself subject to corruption in the physical layer.

C. PHY Layer Module

Fast bit-error injection based on simple burst error models or error gap distributions [14] is applied to each TBS, depending on given background traffic scenarios and assumptions on inner-loop power control algorithms. These have significant impact on the physical link characteristics and constitute a separate field of research. The exact modeling of that link will be addressed in a separate publication. Note that we have to explicitly model handover scenarios, because our simulation is only done for one radio cell.

IV. System Implementation Issues

We have implemented the testbed on a standard UNIX-workstation. Modules are running asynchronously as POSIX threads. Multi-threaded implementation ensures modularity and concurrency between protocol functions. The system offers the following flow parameter set for an arbitrary number of TCP/UDP ports: **1.** QoS, one of the four traffic classes: Conversational, Streaming, Interactive, Background. **2.** Bandwidth requirements **3.** Reliability requirements, which map to protocol related attributes such as ARQ Type (Idle, No ARQ, Selective ARQ), ARQ-Window-Size; and the number of retransmissions. These also guide the resource allocation for UMTS code branches, which are out of the scope of this investigation. **4.** Delay requirements, which will be used in future to drive improved QoS scheduler disciplines. Table 1 shows the various protocol related system parameters and sample values.

V. Ongoing work

The testbed has been used successfully to demonstrate various concurrent Internet TCP and UDP applications such as file transfer, web browsing and a proprietary adaptive and scalable H263++ video codec. The central goal for the testbed is to demonstrate user-perceived QoS, which is difficult to put into word or graphs.

TABLE 1: Sample System parameters

Parameter	**Typical value**
Traffic Class	Interactive
IP maximum segment size	1460 byte
IP header Compression	Off
RLC header	8 byte
RLC-ARQ mechanism	Selective
RLC-ARQ window size	16
Number of Retransmissions	3
TB size	60 byte
MAC header	3 byte
PHY Interval time	10ms
Cell/load scenario	SCN1

VI. Conclusions

We have designed a real-time testbed for performance assessment of IP-based packet switched and emulated circuit switched multimedia applications over UMTS. Our proposed architecture allows concurrent access for an arbitrary number of application-flows, while we provide a QoS understanding on a per-flow basis even for non-QoS aware applications. This specifically allows investigating user-perceived QoS of today's multimedia applications over a simulated wireless link.

Starting from existing investigations on the performance of TCP over a wireless link we broaden the field of investigations to other higher layer control loops such as those found in adaptive RTP-based video applications, and combine them with simulated third generation wireless specific RRM functionality and wireless lower layer protocols.

References

[1] Technical Specification Group Services and System Aspects, *QoS Concept*, Technical Report 3G TR 23.907 version 1.1.0., 1999.

[2] B. Jabbari, E. H. Dian, and W. Fuhrmann, “Performance Aanalysis of a multilink packet access for next generation wireless cellular systems,” PIMRC 1998.

[3] T. Stefansson and M. Allinks, “Real-time testbed for assessing a CDMA-based system,” *IEEE Personal Communications*, October 1995.

[4] O. Lataoui, T. Rachidi, L.G. Samuel, S.Gruhl, and R. Yang, “A Qos management architecture in packet switched mobile systems,” submitted to INTEROP'00, IEEE, 2000.

[5] C. Robool, P. Beming, J. Lundsjo, and M. Johansson, “A proposal for an RLC/MAC protocol for wideband CDMA capable of handling real-time and non-real-time services,” VTC’98, pp. 107-111.

[6] M. Sundelin, W. Granzow, and H. Olofson, “A test system for evaluation of the CDMA technology,” VTC’98, pp. 983-987.

[7] 3GPP, *Radio Interface Architecture*, 3GPP TS 25.301, version 3.1.0, July 1999.

[8] E. Dahlman, P. Beming, J. Knutsson, F. Ovestjo, M. Persson, and C. Roobol, “WCDMA- the radio interface for future mobile multimedia communications*,” IEEE Trans. on Vehicular Technology*, Vol. 47, NO. 4, 1998.

[9] M. Meyer, “TCP performance over GPRS,” Wireless Communications and Networking Conference, pp. 1248-1252, 1999.

[10] G. Brasche and B. Walke, “Concepts, services, and protocols of the new GSM phase 2+ General Packet Radio Service,” *IEEE Communications Magazine*, pp. 94-104, August 1997.

[11] M. Allman, et. al., “Ongoing TCP research related to satellites,” Internet-Draft: http://www.ietf.org/internet-drafts/draft-ietf-tcpsat-res-issues-11.txt, 28 Sep 1999.

[12] S. Dawkins, et. al., “End-to-end performance implications of links with errors,” Internet-Draft: http://www.ietf.org/internet-drafts/draft ietf-pilc-error-02.txt, 26 Oct. 1999.

[13] S. Dawkins, et. al., “End-to-end performance implications of slow links,” Internet-Draft: http://www.ietf.org/internet-drafts/draft-ietf-pilc-slow-02.txt, 21 Oct 1999

[14] L. N. Kanal and A. R. K. Sastry, “Models for channels with memory and their application to error control,” *Proc. IEEE*, vol. 66, no. 7, pp. 724-744, July 1978.