Real-Time Demonstrator for QoS testing in 3G/4G mobile networks with IP multimedia applications

Xavier Revés, Anna Umbert, Ramon Ferrús, Antoni Gelonch and Ferran Casadevall

Polytechnic University of Catalonia, Department of Signal Theory and Communications, Barcelona, Spain.
e-mail: {xavier.reves, annau, ferrus, antoni, ferranc}@tsc.upc.es

Abstract **: Outgoing mobile communications research gives a relevant role to the availability of software/hardware platforms able to cope with the increasing complexity of the 3G and beyond addressed scenarios arisen from the need to, among others, integrate different technologies and evaluate end-to-end performance of the proposed solutions, architectures, applications or services. This paper presents an experimental analysis carried out over a real time UMTS emulation platform designed to evaluate end-to-end QoS performance of multimedia IP-based applications. The developed demonstrator is shown to be a valid tool to assess quantitative but also qualitative results with the flexibility and affordability provided by the fact of being built over a cluster of Personal Computers.**

1. Introduction

Next generation networks are designed to provide a wide range of multimedia services with Quality of Service (QoS) requirements. Within UMTS radio access, Radio Resource Management (RRM) [1] strategies become crucial in providing QoS while IP-based approaches are considered for the core network. To be able to test in the laboratory realistic scenarios addressing such issues, the availability of a testbed, understood as a flexible hardware/software platform, becomes practically mandatory.

This paper presents an experimental analysis carried out over a real time UMTS emulation platform designed to evaluate end-to-end QoS performance of multimedia IP-based applications on a UMTS network with advanced RRM algorithms. The main objective of the analysis is to show the capabilities of the developed platform in assessing quantitative but also qualitative results and the possibility to correlate both of them.

The demonstrator has been developed under the framework of IST ARROWS [2] and GLAMOUR [3] projects. The purpose of the UMTS emulator is to reproduce in real-time the behaviour of a user under test with more accuracy than what would be obtained from system simulators (more suited for global systems performance analysis), but with less implementation complexity than in a real system. Even though the initial UMTS-oriented design and implementation of the demonstrator, its modular structure allows a seamless replacement, addition or suppression of individual parts to improve the behaviour or to evolve towards other system architectures within the framework of 4G systems.

The singularity of the implemented demonstrator relies on the combination of achieved performance, easy of use and cost. Indeed, the platform is built over a networked cluster of off-the-shelf personal computers (PCs). The 10ms UTRAN frame duration mandates to perform hundreds of millions of computations per second. This fact requires a confident timing control and the use of transparent and deterministic communication mechanisms among processes within the emulator.

In the paper, Section 2 describes the main elements of an UMTS network that have been included in the platform. Next, in Section 3, some aspects related to the hardware and software solution are given. Section 4 addresses which are the capabilities and results that could be obtained from the real–time testbed by means of the analysis of a specific UMTS scenario. Finally Section 5 concludes the paper.

2. UMTS Emulation Approach

The aim of the UMTS Testbed is to emulate as close as possible a real 3G UMTS system composed by [4]: the User Equipment (UE), the UMTS Terrestrial Radio Access Network (UTRAN), and the UMTS Core Network (CN). Figure 1 shows the basic mapping of these entities into a network of PC's while Figure 2 provides the main functions included in each UMTS network entity.

The UMTS emulation approach is based on the following key points (for a detailed discussion refer to [5], [2]):

•**Execution of real-time applications**. The user under test is evaluated by means of typical multimedia applications used in IP networks. Different applications (mbone tools for video and audio, Mpeg4ip for streaming, Mozilla for interactive and background) have been selected according to the four service classes identified within UMTS. For each application different PDP contexts with the correspondent QoS profile are allowed. Each QoS profile considers parameters such as peak and guaranteed bit rate, error ratio constraints and transfer delay. Applications are executed over UE and IP server machines (see Figure 2).

•**Quality of service framework**. The adopted solution is based on IntServ/RSVP (Integrated Services/Reservation Protocol) over an UMTS access network that, in turn, can interoperate with DiffServ (Differentiated Services) core networks [6]. Within this approach, an end-to-end IP Bearer service is assumed and, below this level, in the UMTS network, QoS of service is provided by the Radio Access Bearer

(RAB) Service and the CN Bearer Service. A specific library has been developed within the project to allow non-RSVP capable applications to use RSVP and thus QoS.

•**Emulation of the lower layers of the Uu interface**: Packet Data Convergence Protocol (PDCP), Radio Link Control (RLC) and Medium Access Control (MAC). Lower layers design has been done in accordance to 3GPP specifications. Physical layer emulation has been addressed by means of histograms obtained from off-line simulations [7]. Different histograms have been obtained for the different environments (outdoor 120 km/h, outdoor 50 km/h or pedestrian 3 km/h), for each RAB configuration and associated TF (Transport Format) and for different values of E_b/N_0 (bit energy over noise energy).

•**RRM functions**. RRM operations implemented in the testbed include essential functions like admission control, congestion control, outer and inner loop power control, handover management, radio resource allocation and transmission parameters management [8]. Although only the user under test is running real applications, RRM algorithms are applied indistinctly over all the traffic generated by the rest of users emulated in the demonstrator. Traffic generation of these users is done according to some published models summarised in [8]. Furthermore, the mapping of a given service class over the transport channels [9] could be configured in a flexible way in the testbed.

•**Propagation Scenarios and mobility models**. The definition of the scenarios takes into account the radio environment, the mobile distribution and the mobility pattern used to update mobile positions. In the demonstrator a given scenario is loaded by means of a radio propagation matrix that contains, for each position of the matrix, the propagation losses to each one of the BS sites included in the scenario. Scenarios of up 5kmx5km dimension with 10x10m resolution and cell site separation of 500 metres are currently handled in the testbed.

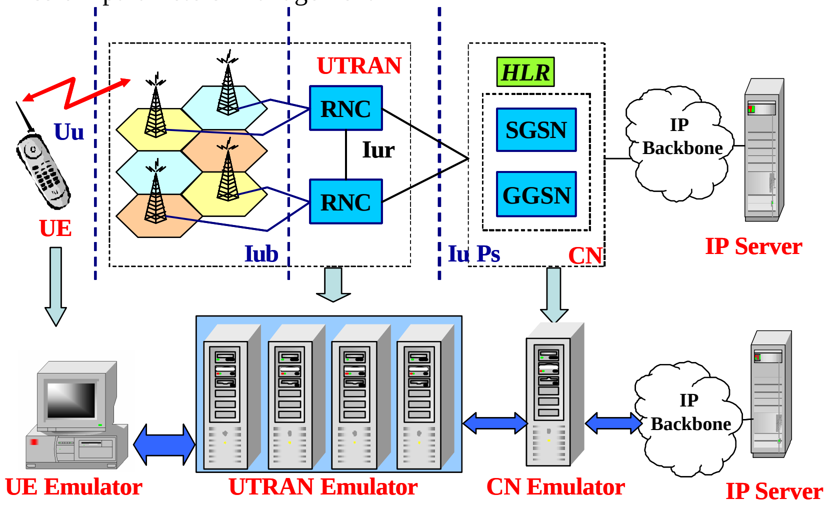

Figure 1. Mapping of the generic UMTS architecture into the demonstrator platform.

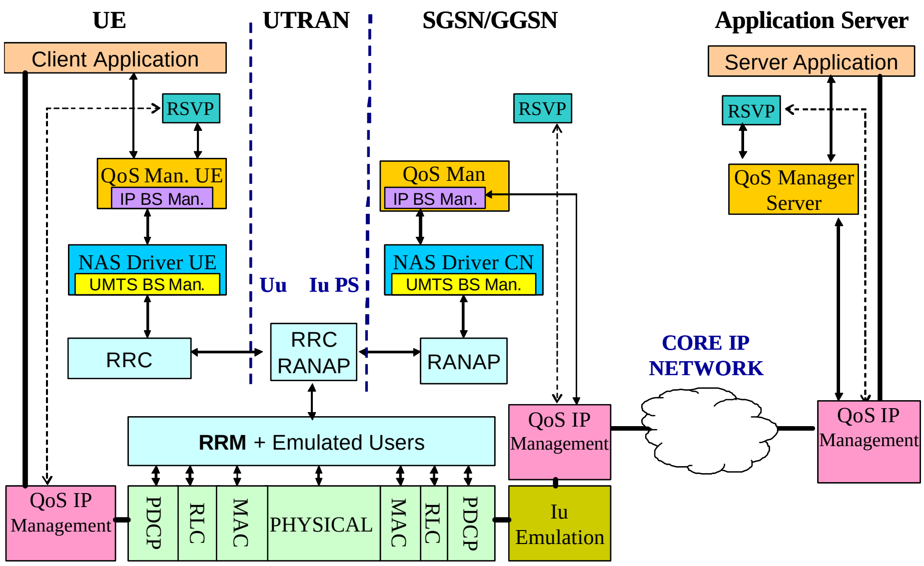

Figure 2. Main UMTS functionalities included in the testbed.

3. Software/Hardware Platform

One of the key aspects of this demonstrator is its real-time operation. In particular, the UTRAN part implementation is the most restrictive because of the 10ms frame duration. Furthermore, the emulation of a high number of users and the execution of complex RRM strategies requires a very high computational power. These computational requirements are out of the scope of today’s off-the-shelf PCs. Then, a cluster of PCs has been constructed to distribute the algorithm throughout different processors. To do so, a tool named Communications Manager (CM) has been designed and implemented to make this distribution completely transparent to the algorithms. This tool allows the construction of clusters within the demonstrator with several PCs offering a scalable platform able to cope with different algorithm structures. CM works as a kind of abstraction layer that can be extended over a set of networked PCs and that separates the algorithms from the underlying hardware.

The current demonstrator is composed of a total of 8 off-the-shelf Personal Computers (PC) connected through two independent Fast Ethernet networks. The operating system used is a Linux Red Hat distribution. There are four main entities within the demonstrator: the user equipment (UE) that uses one PC, the UMTS Terrestrial Radio Access Network (UTRAN) that requires four PCs, the UMTS core network part (CN) that uses one PC, and the Application Server implemented in another PC. The last PC is the demonstrator console that manages and configures all its functional blocks. The management includes monitoring the processes that perform the different tasks (e.g. signalling messages exchanged), viewing real-time statistical results and parameter evolution, starting and stopping the demonstrator, etc. By the other hand, configuration includes parameter initialisation, real-time parameter modification, scenario selection, etc. The console is a graphical tool that has been developed using JAVA and that is in agreement with the rest of the demonstrator philosophy in the sense that is not tied to a concrete architecture but is completely adaptable to any system architecture. A more detailed description of the hardware aspects of the platform can be found in [10].

4. Analysis of an specific UMTS scenario

Studies addressed to show the goodness of RRM strategies under different scenarios could be carried out following two main approaches. The first approach is based on the quantitative evaluation of system dynamics and statistics using a monitoring tool developed for this purpose. The second one consists of evaluating the user perceived quality by means of a panel of human observers. In this way, the real-time operation of the testbed makes possible to obtain valuable results for quite complex scenarios in a very short-time when compared to other event-based simulators tools.

Next, as an example of the aforementioned testbed capabilities, we address the analysis of a specific case study. It is worthy to remark here that our intention is limited to give a better understanding of the testbed capabilities instead of discussing about RRM strategies. Further details concerning RRM for the selected scenario can be found in [11]. The objective of this experimental demonstration is to show how the selection of different short-term Transport Format selection algorithms and congestion control strategies in uplink impacts on system load and interference patterns but also on user perception at application level.

4.1. Effects of short-term algorithms over interactive services.

To avoid any confusion, we mean here by short-term algorithms the strategies which are used to decide at frame level the most suitable Transport Format (TF). This decision is taken by each active user in a distributed approach. However, the TF is chosen among a limited set known as Transport Format Combination Set (TFCS) that is assigned by the system at the beginning of a session (PDP Context establishment) after admission control checking. Table 1 provides which are the TFCS that have been used in the analysis to configure the Radio Access Bearers (RABs) for conversational and interactive traffic classes.

Service		WWW (UL)	VIDEOPHONE (UL)
TrCH type		DCH	DCH
TB sizes, bit		336 (320 payload, 16 MAC/RLC header)	640
TFS	TF0, bits	No transmission	No transmission
	TF1, bits	1×336 (16 Kb/s, SF=64)	2×640 (64 Kb/s, SF=16)
	TF2, bits	2×336 (32 Kb/s, SF=32)	-
	TF3, bits	3×336 (48 Kb/s, SF=16)	-
	TF4, bits	4×336 (64 Kb/s, SF=16)	-
RLC mode		ACK	Transparent
TTI, ms		20	20

Table 1 Transport formats for the considered RABs in the uplink.

As shown in Table 1, WEB traffic is offered through a Dedicated Channel (DCH) where 5 TFs are allowed: TF0-TF4. The selected transport channel to carry out interactive services is provided in Table 1. Under this configuration, TF0 indicates no transmission in a Time Transmission Interval (TTI), TF1 provides 16 kbits/s with SF=128 and a maximum instantaneous bit rate of 64 kbits/s with Spreading Factor (SF) equal to 16 is achieved using TF4.

The user under test will execute Mozilla and download/upload information from/to the server. A PDP context is established with a guaranteed bit rate of 24 kbits/s in both links. The interactive traffic model for the rest of users considers the generation of activity periods, where the user downloads information, followed by inactivity intervals accounting for the time spent after a new page is requested. Traffic models parameters have been adjusted to provide a mean bit rate close to 24 kbits/s. Table 2 provide some of the relevant parameters used to set up the demonstration.

Scenario size	5 km x 5 km
BS parameters	
Cell radius	500 m
Cell type	Tri-sectorial
Maximum transmitted power	43 dBm
Thermal noise	-106 dBm
CPICH_Power	33 dBm
Shadowing deviation	10 dB
Shadowing decorrelation length	20 m
UE parameters	
Maximum transmitted power	21 dBm
Minimum transmitted power	-44 dBm
Thermal noise	-100 dBm
Mobile speed	50 km/h
Handover parameters	
Active Set maximum size	2
AS_Rep_Hyst (replacement hysteresis)	1 dB
Measurement period T_{HO}	0.5s
Time to Trigger	1 T_{HO}
Admission Control	**All users are admitted**

Table 2. Generic parameters of the demonstrated scenario

Two short-term strategies for the selection of the TFs are tested [11]:

- **Rate-oriented strategy or Service Credit (SCr) algorithm**: A certain mean bit rate is guaranteed. The SCr of a connection accounts for the difference between the obtained and the expected bit rate for this connection. So, connections with accumulated credits will result in lower SF and thus higher transmission rate.
- **Maximum Rate (MR) algorithm**: the user is always allowed to select the TF that provides the highest transmission bit rate according to the amount of bits to be transmitted. This approach will tend to reduce the delay for the UE that makes use of it but it will also tend to increase the interference over the rest of UEs (in any case, if the TFCS has been suitably selected by the RNC, this wouldn't pose a problem).

A first case considers a scenario where all the users are executing interactive services. To force the system to be in high load conditions a total of 4000 interactive users are randomly distributed over an urban service area of 5kmx5km. Figure 3, compound of captured windows from the monitoring tool, depicts the probability distribution functions of the TFs usage and the load factor for MR and SCr strategies. For the SCr case, it can be observed that, when transmitting, most of the time TF1 and TF2 are used. This is because the UE buffer queues several packets and, so, it tends to transmit the information at 24kb/s (that falls in the middle between TF1=16kb/s and TF2=32kb/s). In turn, in the periods when the UE buffer is empty (TF0 is used) the UE is gaining service credits and, when a new packet arrives, the instantaneous transmission rate is increased (i.e. TF3 and TF4 are used) to keep the average bit rate around the target value. On the other hand, when MR strategy is applied, UE-MAC chooses the TF according to the buffer occupancy and tries to transmit the information as fast as possible. Consequently, most of the transmitting time TF4 is being used but there are also many time periods where the radio interface is unused (i.e. TF0 is selected) because the buffer is empty. Even though behaviour of both strategies in TF usage is quite different, the evolution of most system level parameters, such as the load factor, remains quite similar for both strategies. Load factor distributions (given in Figure 3 together with the load factor temporal evolution) are calculated at runtime within an average period of 1s. Thus, despite that each user may apply very different patterns on the SF usage, system performance results lead to the conclusion that the specific algorithm applied at UE-MAC level does not provide very different interference patterns in the network. This fact can be justified because the global amount of information to be transmitted is the same in both analysed cases and there is no statistical difference between following either a time multiplexing (MR strategy) or a code multiplexing (SCr strategy) approach. Table 3 compares several parameters at system level instead of at individual user level. It can be realised that quite similar values are obtained in all cases.

However, when comparing both mechanisms (SCr and MR) in terms of averaged packet delay and mean rate per page (kbits/s) observed by individual users, results clearly reveal that MR strategy outperforms SCr. This effect is observed in the perception of the user under test when uploading a file to the server machine. When MR is selected, the use of the service seems to be more comfortable since for this kind of application user perception mainly relies on delay. The following table (Table 4) compares both mechanisms (SCr and MR) in terms of packet delay and rate per page (kbits/s) averaged for all the users and the mean time spent by the reference user to upload a 100kB file.

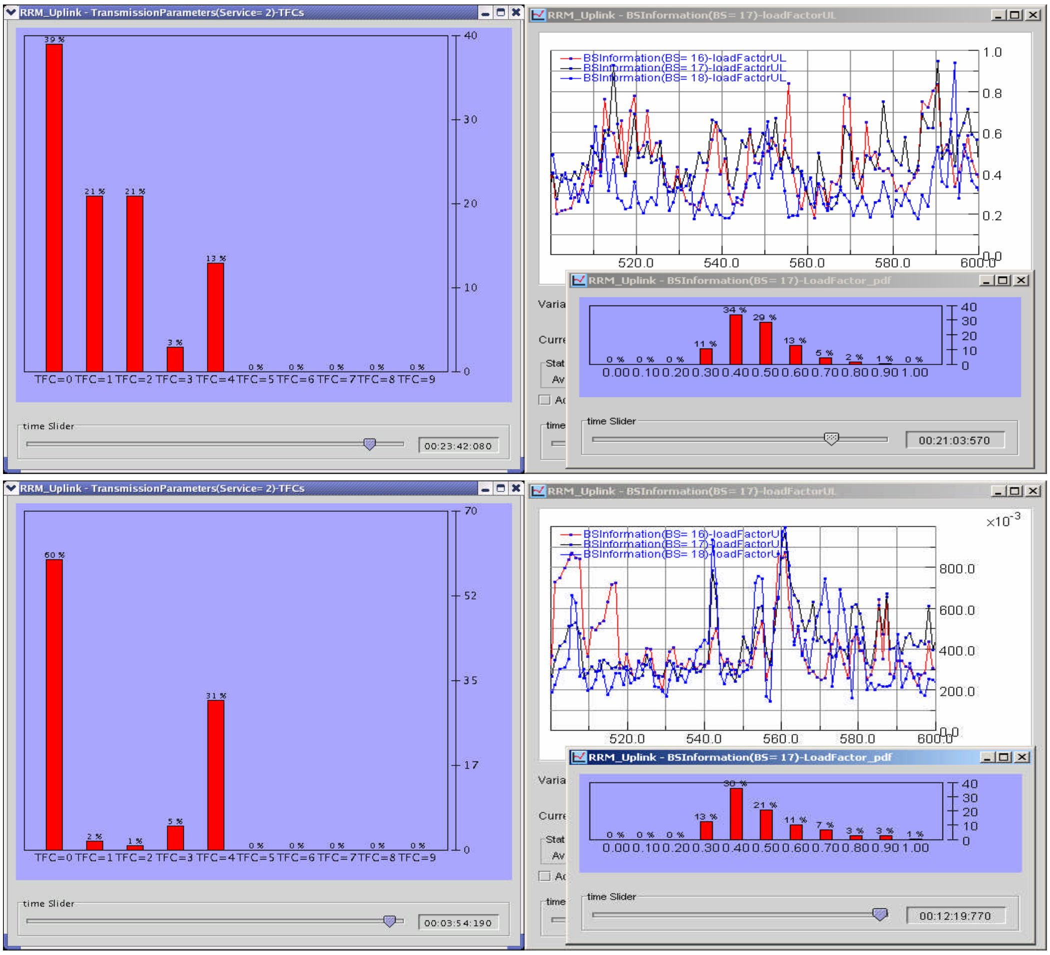

Figure 3. TF distribution (leftmost side) and load factor temporal evolution and distribution (rightmost side) for SCr (top) and MR (bottom) strategies.

Short-term Strategy	Global Averaged BLER (%)	Averaged Load Factor(η)	Prob (η>0.75)	Global Rate (Kb/s)
SCr24	1.47%	0.50	9.84%	11045.8
MR	1.38%	0.49	7.09%	11235.8

Table 3. System performance for different UE-MAC strategies.

Strategy.	Average packet delay (s)	Packet delay jitter (s)	Rate per page (Kb/s)	Rate per page deviation (Kb/s)	**REFERENCE USER** (Mean time for uploading a 100kB file)
SCr24	0.490	0.89	19.9	4.8	51.5 s
MR	0.105	0.172	25.77	12.9	20 s

Table 4. Individual user performance for different UE-MAC strategies.

4.2. Effects of congestion control on conversational services

To extend given results to a more sensitive QoS application, a new trial is carried out based on the previous settings but now the user under test executes a videoconference call (vic/rat tools). The QoS profile is set up with 64 kbits/s guaranteed bit rate for both audio and video. Conversational traffic is mapped over a DCH channel and RLC is configured in transparent mode (no retransmissions). When running this application with the same number of users as stated above the quality is seriously degraded due to the appearance of periods where the load factor is too high. This effect is clearly shown in Figure 4 (leftmost picture). In case of using a congestion control strategy [11] that prioritises conversational service class over interactive class, we can appreciate that the quality of the videoconference call improves significantly while maintaining the same load as shown in Figure 4 (rightmost picture).

However, the improvement achieved in conversational services when using congestion control strategies produces an awful effect on interactive traffic because the observed delay is considerably increased. Results in Table 5 show this effect. Note that results in such a case are quite different from the ones provided in Table 4 where no congestion control mechanism was used. So, the usage of congestion strategies lead to an important increase in transfer delay for interactive users since their TF is controlled in a way that the total amount of interference is kept under control.

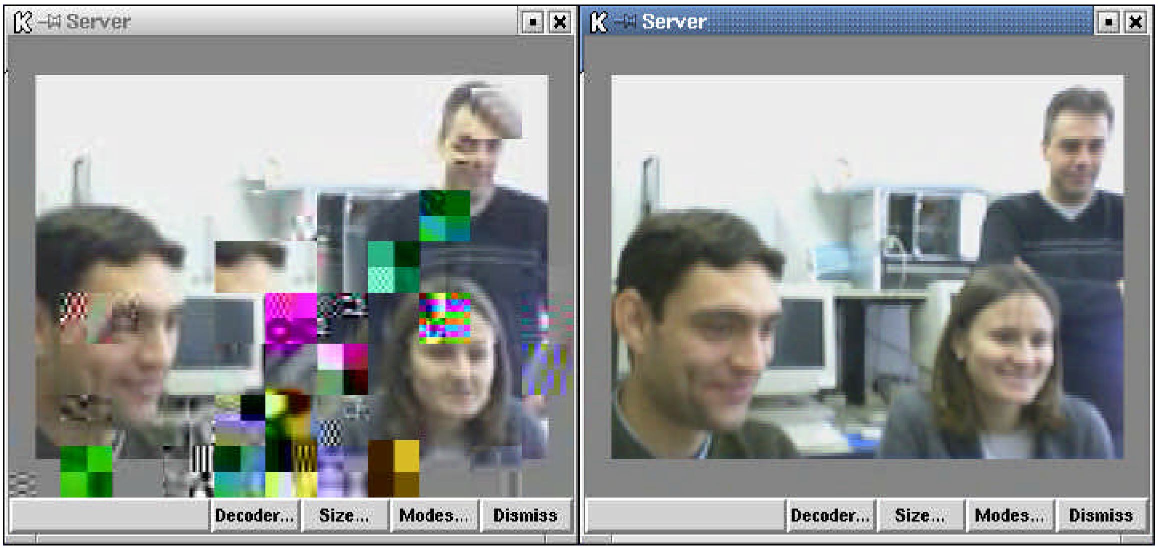

Figure 4. Videoconference application. Left side corresponds to the case when no congestion control is applied.

Strategy.	Average packet delay (s)	Packet delay jitter (s)	Rate per page (Kb/s)	Rate per page deviation (Kb/s)	**REFERENCE USER** (Mean time for uploading a 100kB file)
SCr24	2.55	9.84	18439.5	6238.5	185 s
MR	1.54	7.57	23893.7	13207.0	138 s

Table 5. Delay and rate for different UE-MAC strategies when congestion control is used.

5. Conclusions

The UMTS testbed implemented within ARROWS and GLAMOUR projects includes a complete UMTS stack for a user under test that allows end-to-end IP-based QoS-aware applications to be tested under complex scenarios. The key point of a platform like this is the possibility to obtain quantitative results that can be correlated with the perceived quality when running a real application. Addressing qualitative results is very important to check the goodness of applications not specifically designed to run over mobile networks. This fact could be very relevant during the deployment phase of new services.

The test-bed has been designed to cope with different scenario configurations concerning number of users, type of services, PDP context mapping into RABs, mobility models, propagation conditions, etc. This, together with the emulation of the radio physical channel, provides to the test-bed a realistic behaviour where RRM algorithms must show its effectiveness in the assignation and control of resources. It is worth mentioning here that new different layouts, new radio resource management algorithms or new standard applications can be incorporated in an easy way into the testbed.

Even though the high computational requirements and timing restrictions of the emulated system, a network of standard PCs with a Linux Operating System has been shown to be feasible to cope with real-time UMTS emulation challenges. Some minor modifications to Linux kernel permit all the required control over processes and internal data flow without lost of flexibility. This approach represents an affordable solution to carry out many research activities on this field.

Further work actually being carried with the platform is concerned with the execution of MOS

(Mean Opinion Score) trials for the multimedia applications implemented in the platform.

Acknowledgments

This work has been partially funded by the European Commission under the ARROWS project within the IST framework (IST 2000-25133) and by the Spanish Science & Technology Ministry under grant TIC 2001-2222.

REFERENCES

[1]. Harri Holma and Antti Toskala, “WCDMA for UMTS”, John Wiley & Sons, 2000.

[2]. ARROWS (IST-2001-25133) Project WEB site, http://www.arrows-ist.upc.es/

[3]. GLAMOUR (TIC2001-2222)Project WEB site, http://www.glamour.upc.es/

[4]. 3GPP Technical Specification 23.101, “General UMTS Architecture”.

[5]. Ramon Ferrús et alt., "Real-time emulation of RRM strategies for UMTS Bearer Services”, Vehicular Technology Conference 2002 Fall (VTC2002-Fall). Vancouver, British Columbia, Canada, 24–28 September 2002.

[6]. 3GPP Technical Specification 23.107 v4.0.0, “QoS Concept and Architecture (R4)”.

[7]. J. Olmos, S. Ruiz, “UTRA-FDD Link Level Simulator for the ARROWS Project”, IST’01 Conference Proceedings, pp. 782-787.

[8]. “System specification. Radio Resource Management Algorithms: Identification and requirements” Public Deliverable D04, ARROWS Project.

[9]. 3GPP Technical Specification 25.211 v5.1.0, “Physical channels and mapping of transport channels onto physical channels (FDD)”.

[10]. Xavier Revés et alt., "Implementation of a Real-time UMTS Testbed in a PC Network”, 3rd ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing, June 26-27, 2002, Madrid, Spain

[11]. O. Sallent, J. Pérez-Romero, R. Agustí, F. Casadevall, “Provisioning Multimedia Wireless Networks for Better QoS: RRM Strategies for 3G W-CDMA”, IEEE Communications Magazine, February 2003, pp. 100-106.