NOMOR RESEARCH GMBH
BRECHERSPITZSTR.8
D-81541 MÜNCHEN
GERMANY

REALTIME NETWORK SIMULATOR (REALNES) PLATFORM

WHITE PAPER

30. OKTOBER 2006
EIKO SEIDEL, JUNAID AFZAL, GÜNTHER LIEBL

REALTIME NETWORK SIMULATOR (REALNES) PLATFORM

1. SUMMARY

Today's mobile communication systems are becoming more and more complex, whereas product cycles and required time-to-market is getting shorter. Cost-efficient and reliable system design is therefore of utmost importance to both operators and vendors. Being able to experience, understand, and assess actual system behaviour already during the design phase before hardware is available or the systems are actually deployed is extremely desirable: Since most of the systems should be user centric, it is essential to understand application performance and user perceptions at the earliest possible state. As a result, there is a growing demand for simulation tools which are capable of emulating the complex nature of today's system in a realistic way.

However, conventional offline system simulators are not very well suited for investigating real-time services and multimedia applications over wireless shared channels. It is well known that certain effects, such as end-user perception, end-to-end QoS, and Quality-of-Experience (QoE) [2] cannot be fully understood with offline simulation models and/or analytical derivations. As a simple example, consider a TCP application running over a wireless link: The experienced data rate at the application layer depends on the round trip time, the packet error rate, and the experienced delay jitter. For this reason, it cannot be assumed that increased peak data rates at the physical layer directly lead to increased throughput also at the higher layers. To investigate the influence of dynamic interactions between layers on the overall end-to-end performance, real-time system simulations are needed instead.

In this paper a dynamic Real-time Network Simulator Platform (RealNeS) is presented, which is capable of simulating any kind of packet-switched network in real-time. Up to now, we have implemented, together with strategic partners, several standard compliant wireless systems (e.g. GERAN, UMTS, HSDPA, HSUPA, MBMS, WiMAX, etc.) in our RealNeS platform. Our approach allows evaluating the performance of all types of IP-based multimedia services over different radio access networks in real-time. With the potential to directly tune various system parameter settings at different layers, our platform represents a valuable tool when trying to maximize end user perception in a network and to minimize radio resource utilization. Furthermore, with the help of our platform, it is possible to develop new protocols and smart scheduling algorithms that boost performance of existing real-time applications, such as interactive video streaming, gaming, VoIP etc. as well as to design new applications for a mobile environment.

In summary, RealNeS allows to investigate almost all aspects of cross-layer optimization and its influence on the Quality-of-Service. This is achieved by various interfaces for tuning radio resource management parameters, such as call admission control, radio resource reservation policies, and scheduling. Finally, the performance of different radio access bearer configurations can be assessed in real time.

2. OVERVIEW OF THE APPROACH

SIMULATOR REALIZATION AND SETUP

Our platform is based on a straightforward modular approach that allows evaluation of IP-based multimedia applications. The entire e2e network simulation is sub-divided into three components: the server, the client, and the network simulator as shown in Figure 1. Each of these entities is implemented on a single PC, and all of these PCs are connected via regular high-bit-rate IP connections, such as Ethernet or WLAN. Hence, the server and the client basically represent the two end-points from an IP perspective.

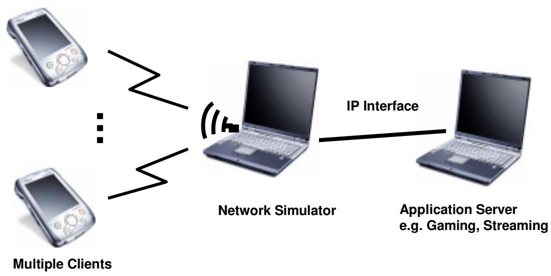

Figure 1. Simulation setup

The network simulator PC simulates the entire wireless network and will either delay or drop the incoming IP packets in both directions according to the momentary characteristics of the radio access system. The network simulator can either include a core network simulation itself, or could be connected to the server via a real IP network.

STANDARD IP INTERFACES

The network simulator provides standard IPv4 and IPv6 interfaces for incoming and outgoing traffic, i.e. it supports almost any type of IP-based traffic using protocols like TCP or UDP. All computers are connected with 100 MBit/s Ethernet or a Wireless LAN (WLAN) connection, which offer data rates several times higher than current cellular networks.

The network simulator is able to support more than one networked client (terminal) that receives IP traffic from one or more application servers. Mapping of IP traffic onto different radio users and flows is achieved with a combination of filtering based on IP addresses and transport layer ports. Each client might use multiple applications in parallel, whereby the effects of different parameter settings, such as user topology, link quality, traffic characteristics, flow priorities, etc. on the mobile system and the transport protocols are monitored in the user interface.

In principle, any IP based application can be tested over the simulator. Examples are, for instance:

- Video streaming and conferencing
- Voice over IP/over IMS (VoIP/VoIMS)
- Push to talk over Cellular (PoC)
- Gaming
- WWW, FTP, Email etc.

Application developers already make frequent use of this platform to test the user perception and to optimize the application behaviour for a particular network setting. Of particular interest here are QoS-critical applications, since our simulator provides all means to investigate the feasibility of such services.

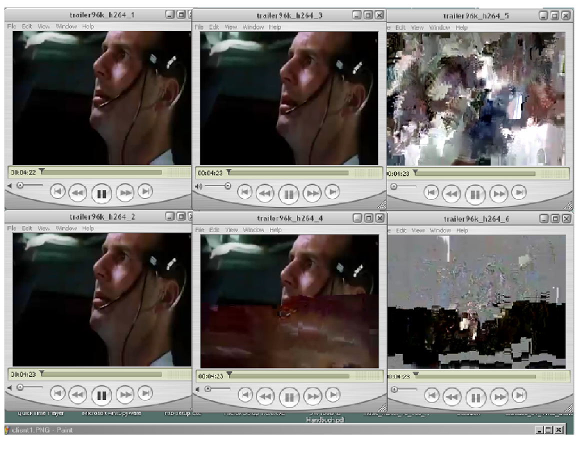

Figure 2: MBMS video streaming of different quality

In the following we will briefly discuss some details behind the realization of our simulation environment for wireless networks. In principle, any communication protocol or network could be implemented. Figure 3 illustrates the example of an HSPA network, whereby different colours are used to differentiate between various layers of abstraction.

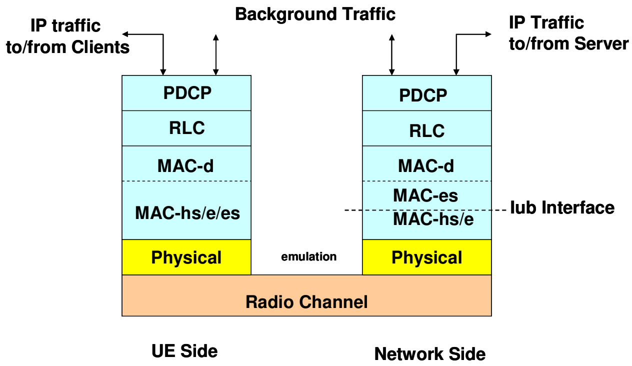

Figure 3: Functional split in protocol stack for HSPA

SPLIT MODELLING APPROACH

The system complexity, the large number of algorithms and protocols involved from end-to-end, the numerous possible parameters and configurations, in combination with mobility, multi-user aspects, different traffic characteristics, etc. require different levels of abstraction. For this reason, we have resorted to a split modelling architecture for the network simulator, which consists of the following components:

- Protocol Simulator
- Link Level Simulator
- System Level Simulator

PROTOCOL SIMULATOR

The protocol simulator is used for modelling the various functions inside the radio protocol stack, such as segmentation, encapsulation, multiplexing, prioritization, scheduling, retransmission, reordering, reassembly etc.. For UMTS HSPA the protocol is, for instance, implemented according to the specifications for PDCP [3], RLC [4] and MAC [5] as depicted in Figure 3.

In our model, header compression/decompression is applied at the PDCP layer to remove redundant information in the cumulative Internet header before transmission. In the radio link control (RLC) entity, a large number of complex tasks is performed: Segmentation/reassembly according to the requirements of a specific bearer service, as well as optional backward error correction of corrupt segments by applying ARQ techniques. Moreover, since most IP-traffic is in general assumed to be transported over so-called shared channels at the air interface, throughput and delay are not only determined by the above-mentioned protocol functionality and the loss rate of the chosen bearer service, but also depend largely on the actual scheduling strategy at the medium access (MAC) layer. Hence, our approach contains the complete protocol stack below the network layer at the air interface.

LINK LEVEL SIMULATOR

A real-time implementation of the physical layer of each radio link in the system is completely infeasible even with the most powerful of today's PCs. For this reason, all computationally demanding operations, such as modulation, coding, channel estimation, detection etc. are simulated separately for each practically relevant channel condition that could occur in the system simulations. The offline link-level simulations produce block error rate (BLER) statistics for a single user and a varying set of parameters, which are SNIR, modulation and coding scheme, HARQ retransmission count, etc.

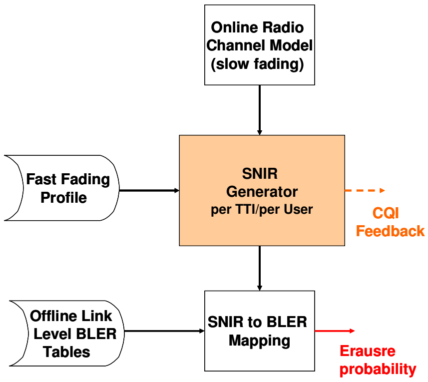

Figure 4: Offline-online processing of physical layer and radio channel

Thus, the physical layer can be simply modelled as a so-called packet erasure channel in the real-time system-level simulation, in which look-up tables are used for packet loss calculation. All BLER lookup tables (BLER vs. SNIR) have been generated according to [6]. For a given SNIR, the corresponding BLER serves as the probability of erasure for that MAC frame as shown in Figure 4 (see [7] for more details on interface). A random generator then produces frame errors with a certain probability that corresponds to the actual BLER. A frame error means that the current MAC frame is not successfully decoded (in reality the PHY will detect packet errors by means of a CRC check sum added to each transport block). In the simulation, the physical layer error indicator is forwarded to the MAC and subsequently to the RLC layer. Hence, a received RLC-PDU can be assigned one of two different states at the receiver: either they are correct, or they are lost.

SYSTEM LEVEL SIMULATOR

The system level simulator provides simulation of a whole mobile environment with a specific cellular layout, base station deployments, and transmitter configurations. We assume hexagonal cells of variable size. Base stations are located at the edges of the hexagons and thus supply signals to three cells by making use of 120 degree sector antennas. Standard-compliant antenna beam patterns are applied to the transmit signal. As shown in Figure 5, a variable number of mobile users may move across the coverage area according to a certain mobility model. This model involves random movement of the users according to the speed that matches the fast fading channel model (3 km/h for pedestrian, 50 km/h for vehicular). The radio link is subject to multipath propagation and is thus characterized by a certain power delay profile and Doppler power spectral density according to the user velocity. Four different channel models are commonly used in 3rd generation networks: Pedestrian A/B and Vehicular A/B (see [8]). The shadowing component models the influence of buildings and other obstacles on the long-term signal propagation. It contains, for example, statistical models with a given (log-normal) standard deviation.

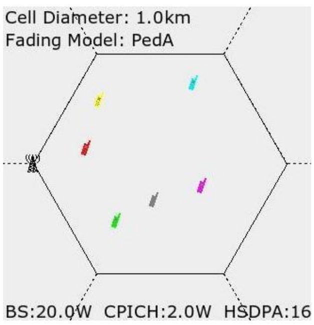

Figure 5: Exmplary cell view

For each uplink and downlink physical channel the signal-to-noise-and-interference ratio (SNIR) is computed considering path loss, shadowing, fast fading, as well as intra- and inter-cell interference due to other users in the system and self interference caused by multipath propagation. Since the SNIR of each user typically varies over time, it must be dynamically updated as shown in Figure 4. For downlink transmission, the update is done once per TTI, whereas for uplink transmission, it is updated once per slot. In the latter case, the effective SNIR must then be determined for each TTI based on the equivalent convex metric (ECM) [9] before accessing the BLER curves.

The total allowed power of the base station (per sector) is configurable. Nevertheless, not all of this power is available for the user data channels in the downlink due to the resource consumption by other physical channels (e.g. BCCH, PCH, Pilot Channel, etc). On the uplink, the limiting parameter is the maximum transmit power of a mobile terminal in most systems.

An extensive set of performance measurements or system state information can be extracted from the network simulator and must be visualized in real-time during a simulation run. Hence, we have designed an interactive graphical user interface (GUI) to facilitate this task.

The GUI usually consists of:

1. *Toolbars* containing drop down menus that allow switching of the most important parameters dynamically, e.g. scheduler algorithm, fast fading model, data rate, etc.
2. *Parameter panels* containing selected parameters of the simulated users, system parts, or protocol functionalities.
3. A *cell view* which displays the user position and movement in the cell as depicted in Figure 5. Each user is identified by a unique colour. Dragging the user to a specific position in the cell is also possible.
4. *Taps* to switch between different system views, e.g. uplink and downlink view.

In Figure 6 and Figure 7 two examples of different parameter panels are given.

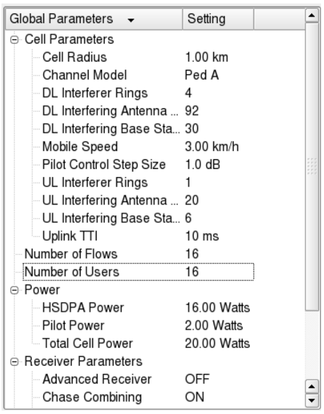

Figure 6: Example global parameter panel

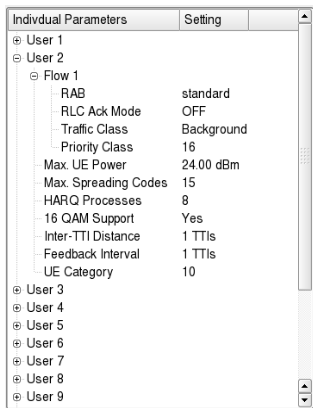

Figure 7: Example user parameter panel

Figure 8 and Figure 9 present two realizations of a graphical user interface. Figure 8 illustrates the GUI of a WiMAX network simulator, with different windows to report the current resource allocation, the channel state, and the achievable user and system throughput and delay.

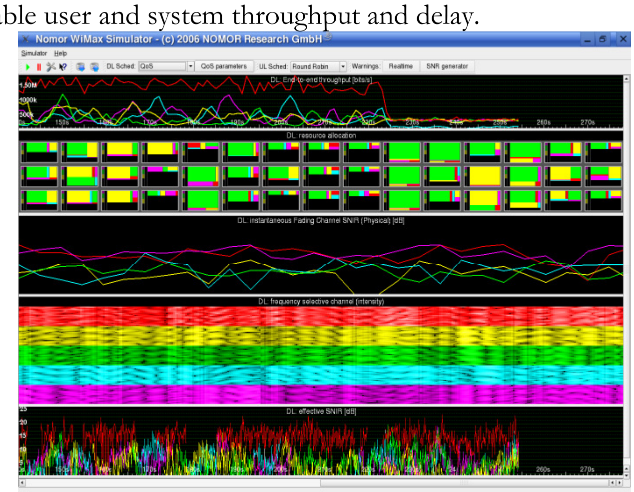

Figure 8: GUI for a WiMAX Network Simulator

Each user is identified by a unique colour both in the cell view and the different plots. Readability is enhanced by interactive selection of plots. Alternatively, selected statistics can also be written into trace files for further analysis. At each layer, standard performance metrics, like throughput, delay, and BLER, can be analyzed for each user or each flow. Other protocol-specific statistics are also possible, e.g. number of retransmissions, queuing delay, interference values, rise over thermal (RoT), power measurements, statistics on the used modulation/coding format, etc. However, the availability of these statistics largely depends on the chosen scenario and the system configuration in general.

The GUI in Figure 9 visualizes only the instantaneous QoS (throughput, delay, error rate), which is important when testing applications. Furthermore, it also reduces the processing requirements for the graphical display, which could have a negative impact on the real-time processing.

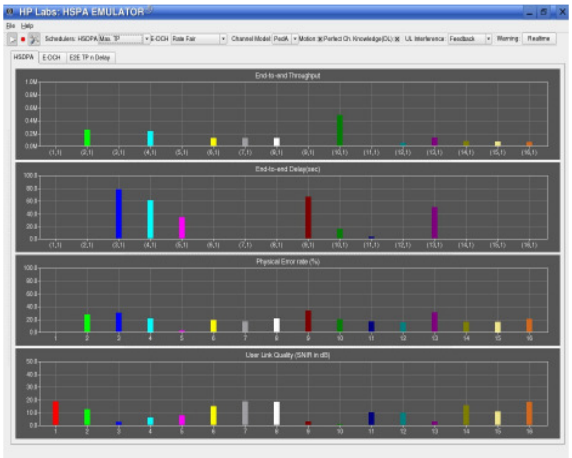

Figure 9: GUI for an E2E application test suite

5. APPLICATION PROGRAMMING INTERFACE'S

Our simulation platform provides certain well-defined APIs that give potential users the possibility to plug in and test customized algorithms. Basically, this method allows separation of the system design into modular tasks and to perform stepwise testing of different components within the overall system environment. The simulator can dynamically link the customized algorithms as '*plugins*', which replace the respective functionality in the original version.. Examples for specified APIs are:

- Uplink scheduler
- Downlink scheduler
- Header compression algorithm per flow
- Link to system level interface

These plugins run within a sandbox, i.e. rely on the main program's public interface and have a well-defined boundary to their possible set of actions. This approach leads to more flexible code design, which reduces the development costs.

6. TRAFFIC GENERATORS

The behaviour in any wireless network largely depends on the cell load. Therefore, our simulation platform supports two different types of traffic sources: On the one hand, multiple live traffic flows can be injected into the simulator both in the uplink and downlink direction to test external IP-based applications. On the other hand, a wide selection of virtual IP traffic generators are available to model cross-traffic from background users. This traffic is not visible on the network interfaces of the simulator computer; but is internally routed through the whole protocol stack to achieve a sufficient loading of the cell. The following basic traffic generators can be selected for the background traffic.

- Constant bit-rate (CBR)
  Generates constant bit-rate traffic for a desired packet size.
- Exponential (EXPO)
  Generates bursty traffic with exponentially distributed periods of silence.
- World Wide Web (WWW)
  Generates internet browsing traffic with pareto distributed packet size, whereby burst length and inter-burst pauses follow an exponential distribution.
- Fill Traffic (FILL)
  Generates as much traffic as currently supported by the radio bearer (simple model of a TCP source).
- Multimedia Messaging Service (MMS CBR)
  Generates uplink MMS bursty traffic with constant bit-rate over specified time interval.
- Multimedia Messaging Service (MMS FILL)
  Generates uplink MMS traffic with FILL principle.
- Voice over IP (VoIP)
  Generates traffic with typical encoded conversational voice characteristics.

Any mixture of traffic can be configured for the different users in the system. Additionally, each user can have multiple flows in parallel.

7. REFERENCES

[1] White Paper: The technology of High Speed Packet Access, Nomor Research http://www.nomor.de/home/technology/white-papers

[2] 3GPP TS 23.107 (V6.3.0) "QoS Concept and Architecture (Release 6)"

[3] 3GPP TS 25.323 (V6.3.0) "Packet data convergence protocol"

[4] 3GPP TS 25.322 (V6.5.0) "RLC protocol specification"

[5] 3GPP TS 25.321 (V6.7.0) "MAC protocol specification (Release 6)"

[6] B. Zerlin, M. Ivrlac, W. Utschick, J. A. Nossek, A. Klein, I. Viering, "Joint optimization of radio parameters in HSDPA", in Proc. IEEE Vehicular Technology Conference (VTC) Spring 2005, Stockholm, Sweden, May 2005.

[7] A. Seeger, M. Sikora, and A.Klein, "Variable Orthogonality Factor: a Simple Interface between Link and System Level Simulation for High Speed Downlink Packet Access ", in Proc. IEEE Vehicular Technology Conference (VTC) Fall 2003, Orlando, Fl, USA, October 2003.

[8] Rec.ITU-R M.1225 Guidelines for Evaluation of Radio Transmission Technologies for IMT-2000.

[9] Tdoc R1-040227 "Comparison between Short Term and ECM Approach for H-ARQ", Motorola

ABOUT NOMOR

Nomor Research GmbH is a leading company in the area of real time system emulation and specialized in the implementation of future radio access networks. In sustained research projects with strategic partner companies we develop and implement leading edge technologies such as HSPA, MBMS, WiMAX, and Long Term Evolution. Our end-to-end emulation platform running in real-time allows

- to design, test and verify algorithms across all layers
- to demonstrate system performance and feasibility
- to show the actual experienced user perception of applications,

years before systems are actually deployed or available in hardware. Nomor is active in various standardisation bodies to enhance today's mobile communication systems with a main focus on future radio access schemes, protocols, and applications.

Nomor Research is an independent company. Our customers gain from our experience in research, standardisation, and implementation, as well as from our standard conform simulation environment. We invite you to find out more about our mission, goals, and competence and how we can help YOU.

Phone: +49 89 978980-00
Fax: +49 89 978980-10
Email: info@nomor.de
Url: www.nomor.de