2007:088 CIV

EXAMENSARBETE

Scheduling of Multi-Media over 3GPP LTE

Mats Wernersson

Luleå tekniska universitet

Civilingenjörsprogrammet
Medieteknik
Institutionen för Systemteknik
Avdelningen för Medieteknik

2007:088 CIV - ISSN: 1402-1617 - ISRN: LTU-EX--07/088--SE

Scheduling of Multi-Media over 3GPP LTE

Mats Wernersson

March 15, 2007

Abstract

As the 3rd Generation Partnership Project is in the process of defining the Long-Term Evolution (LTE) of 3G, there is a high need for evaluations of different approaches in this future cellular system. One of the most significant changes with LTE, compared to current and earlier cellular telephone systems, is that it is aimed to be an all-IP network where only packet switching is supported. For example, this means that the voice service will no longer be utilizing separate circuit switched channels. Even though this IP-only approach allows for streamlining the system for packet services, which will lead to great improvements in the form of higher bit-rates, lower latencies and a wider array of service offerings, it also poses new challenges that need to be overcome.

This master's thesis investigates the effects that different Quality of Service associated scheduling strategies impose on the performance of mixed services over LTE. A traffic scenario where all users engage in both a Voice over IP (VoIP) conversation and a video session is applied in an extensive network simulator. The Session Initiation Protocol is used to set up the media connections. In some of the performed simulations, separate presence service users were included in the system, with the aim to analyze the effect of the presence noise that they introduce. The simulation results indicate that prioritization of the setup messages is highly recommended in order to ensure their delivery and the completion of multi-media telephony session setups even if the system load is high. Furthermore, this study finds that it is possible to prioritize the VoIP traffic relative to the video traffic and thus heavily increase the VoIP capacity without significantly harming the quality of the video service. This gives operators the opportunity to, in the case of a highly loaded system; guarantee high voice quality even if no video can be delivered due to the high load. Regarding the effect on the mixed service users, caused by the additional presence users, no significant change in the performance could be measured with the applied 200 presence users per cell, no matter if the presence messages were prioritized higher or equal to the media traffic.

Acknowledgements

First, I want to acknowledge and thank my supervisor at Ericsson Research in Luleå, Stefan Wänstedt, whose guidance and assistance throughout this work proved to be crucial to the realization of the thesis. Also, thanks to Krister Svanbro for giving me the opportunity to carry out this study and to my examiner at Luleå University of Technology, Peter Parnes. Furthermore, I wish to express my gratitude to all other employees at Ericsson Research whose discussions and helpful insights greatly have helped me with this thesis.

Last but not least, I would like to thank my friends and family and give special thanks to Anna.

Contents

Abbreviations

Acronym	Explanation
3G	3rd Generation
3GPP	3rd Generation Partnership Project
AMR	Adaptive Multi Rate codec
CDF	Cumulative Distribution Function
CDMA	Code Division Multiple Access
CNAME	Canonical Name
GW	Gate-Way
HARQ	Hybrid Automatic Repeat reQuest
HSPA	High Speed Packet Access
IETF	Internet Engineering Taskforce
IMS	IP Multimedia Subsystem
IP	Internet Protocol
ITU	International Telecommunication Union
LTE	Long-Term Evolution
MMTel	Multi-Media Telephony
MTU	Maximum Transmission Unit
OFDM	Orthogonal Frequency Division Multiplexing
QCI	QoS Class Identifier
QoS	Quality of Service
RB	Resource Block
RFC	Request For Comments
ROHC	Robust Header Compression
RTP	Real-time Transport Protocol
RTCP	RTP Control Protocol
SC-FDMA	Single Carrier Frequency Domain Multiple Access
SDP	Session Description Protocol
SDU	Service Data Unit
SIMPLE	SIP Instant Messaging and Presence Leveraging Extensions
SIP	Session Initiation Protocol
TCP	Transmission Control Protocol
TFP	Traffic Forwarding Policy
TTI	Transmission Time Interval
UDP	User Datagram Protocol
UMTS	Universal Mobile Telecommunications System
UTRA	UMTS Terrestrial Radio Access
UTRAN	UMTS Terrestrial Radio Access Network
VoIP	Voice over IP
WCDMA	Wide-band CDMA

Table 1: Acronyms

Chapter 1

Introduction

In this first section, the background, aim and motivation for this thesis will be introduced. A short presentation on the outline of this report will also be given.

1.1 Overview

Today, with the current 3rd Generation (3G) wideband cellular networks, an increasing array of Internet Protocol (IP) based services are being offered to the end users. Thanks to the broadband capabilities provided by new radio access technologies such as High-Speed Packet Access (HSPA), these IP over wireless services are becoming exceedingly useful. However, since the normal voice service in the 3G networks of today is still delivered over dedicated circuit switched channels there is not an optimal flexibility in the system when the telephony is to be enriched by other complementing media services. Among operators there is a wish to converge the different networks into one single all-IP network [1], partly due to the benefits in flexibility but also as this would reduce network complexity and overall operating costs. The 3rd Generation Partnership Project (3GPP) has therefore standardized a Multimedia Telephony (MMTel) service over the IP Multimedia Subsystem (IMS) [2] [3] and decided that the Long-Term Evolution (LTE) of 3G will be a packet switched system only. As no circuit switched channels will exist there, Voice over IP (VoIP) will be used to deliver the voice frames of a conversation and the media sessions will be setup by the Session Initiation Protocol (SIP). Even though VoIP has many benefits, it also comes at some cost. One disadvantage is the requirement to transfer IP headers all the way to the recipient which results in a higher total bit-rate [4]. Another challenge is how the operators are to guarantee a certain Quality of Service (QoS) in a system where all traffic is propagating over shared channels. The QoS for VoIP needs to be at least very close to the one provided by the public switched telephone network in order to make customers accept the quality and make it commercially viable [5]. In order to achieve guarantees for media qualities and the delivery of important setup messages, the schedulers of the system must be able to differentiate traffic flows based on the service they originate from and prioritize them according to an operator-defined policy.

1.2 Objectives and delimitations

This thesis aims to analyze the effects of QoS associated scheduling strategies on the performance of mixed services over LTE. An evaluation with the goal to determine what scheduling concepts might be favorable if certain quality guarantees are to be delivered will be performed. The schedulers will differentiate traffic

flows based on their origin and prioritize them in accordance with predefined algorithms. The algorithms will be varied in order to rate their relative effect on service performances.

More specifically, three groups of scheduling strategies will be investigated by executing corresponding sets of network simulations. The first group aims to ascertain if SIP messages need to be prioritized in order to ensure that MMTel session setup communication can be carried out even if the system load is high. In the case of prioritization of these setup messages, an analysis will be performed regarding if and to what extent this traffic importance ranking affects the quality of the other services. Furthermore, a study will be performed to see how the traffic from a presence service will influence the performance of other services. Since the presence service over IMS communicates by SIP messages, it is an interesting topic whether the same prioritization scheme can be applied to all SIP traffic or if SIP setup traffic should be differentiated from the presence traffic in the scheduling. Finally, the possibilities to prioritize a certain media flow and thus be able to achieve a higher capacity for this service will be investigated.

Considering delimitations on the scope of this thesis, only the downlink[1] of LTE will be considered, both concerning scheduling algorithms and capacity measurements. Moreover, the number of MMTel services applied in the mixed traffic scenario of the simulations has been limited to VoIP and a video stream, in order to concentrate the focus on the effects on those. The thesis will not delve deep in the area of user satisfaction. A number of simple assumptions based on previous studies will be made regarding how media is experienced by the end users. Only packet loss and packet delay will be used to estimate the quality of the received media streams. Neither will much focus be placed on the presence service, its functionality and the quality experienced by its users. Instead, the area of interest will be how the existence of presence users affects the quality of other services.

1.3 Thesis outline

So far, this report has provided an introduction and a description on what this thesis aims to contribute. In the next chapter, Chapter 2, some basic coverage of the area of interest is given. Some background and basic descriptions of the systems, protocols and functionalities of this study are presented. Following the background section comes a presentation of the network simulator in Chapter 3. Chapter 4 provides more detail on the models applied in this study and Chapter 5 describes the different simulations that were run. Chapter 6 presents the obtained results from the simulations and Chapter 7 finishes with a discussion that includes conclusions and thoughts on future complementing work.

[1] The link from the base transceiver station to the user equipment

Chapter 2

Background

This section will provide an extension to the very general background given in Section 1.1. The LTE system, which is the modeled system in the simulations of this thesis, will be described and discussed in the first part of the chapter. The remaining part will describe some of the most important protocols and concepts later referred to in this report. External references will also be provided in the case that further information on the subjects is desired.

2.1 3GPP Long-Term Evolution

In 2004, 3GPP started a study-item called "Evolved UTRA and UTRAN" [6]. The purpose of this study was to define a Long-Term Evolution (LTE) of 3GPP-based access technology in order to ensure its future competitiveness among other emerging radio technologies. 3GPP LTE is targeted to have lower latencies, have higher user data rates and an overall improved system capacity and coverage compared to systems of today. The system aims to be an extensive evolution of 3G, a precursor to 4G and is planned to be released around the year of 2009.

Even though the LTE project is still ongoing and general in its scope, a number of specific goals have been set up. First of all, the system will support packet-switching only, meaning that the circuit-switched voice connections of current systems will be replaced by VoIP. As a result of this, the complete system architecture can be streamlined for packet services and an evolved QoS concept [7]. Such targets as peak data rates of 100 Mbps for the downlink and 50 Mbps for uplink, round-trip times shorter than 10 ms, increased spectrum flexibility and reduced costs for both end users and operators have been established as well.

One of the main technologies to be used in LTE is a new physical layer with Orthogonal Frequency-Division Multiplexing (OFDM) for the downlink and Single Carrier Frequency Domain Multiple Access (SC-FDMA) for the uplink. An introduction to OFDM can be found in [8]. This design with OFDM and SC-FDMA was for reasons not to be discussed in this brief overview preferred over evolving the current Wideband Code Division Multiple Access (WCDMA). A background, a more detailed description and an evaluation of the proposed 3G LTE radio interface is provided in [9]. Here, another important technique, the Multiple-Input-Multiple-Output technique is also discussed. The basic concept of this technique is to use multiple transmitters and receivers to achieve higher bit rates and improved coverage. In addition to the new physical layer, LTE will have simplified and less complex network architecture with fewer network nodes compared to current networks.

2.2 Session Initiation Protocol

The Session Initiation Protocol (SIP) [10] is a signaling protocol for setting up, modifying and terminating sessions. It also allows for user mobility by using registrations of a user's current location. Through the use of a static identifier for each user, it is possible to reach him or her independently of where the user is currently located.

The protocol is applied at application layer level, is text based and is typically used for internet telephone calls, multimedia distribution, multimedia conferences and other similar IP-based sessions. It is today widely used and is the main signaling protocol of the IMS.

SIP will be used for both setup and presence signaling in the simulation scenarios of this study. The specific SIP message flows applied there are described in Section 4.2. For a more detailed and general description of SIP processes, see [3].

2.3 Presence

Presence service is the functionality for getting input on a user's availability without having to directly contact him/her. The most basic presence states are "online" and "offline", i.e. information on if they can be reached or not, but the status list can be extended by an arbitrary number of possible states, such as "busy", "away", "on the phone" and so on. Some well known applications where presence functionality is used are MSN Messenger, ICQ and Skype.

There is no universal protocol used by all presence applications, but the Internet Engineering Taskforce (IETF) has standardized an extension to SIP [11] in order to make it suitable for presence functionality. The extension is named SIP Instant Messaging and Presence Leveraging Extensions, or SIMPLE. An overview of SIP and presence can be found in, for example, [12].

There exist two commonly used systems regarding how updates on the buddy-list are to be distributed from the server that keeps track of all presence users. The system can either be pull- or push based. In a pull-based system, it is the client that initiates the notification transmissions. By letting the client send a message every time an update is desired, the number of update messages can be limited by being sent only when the users needs them. In a cell-phone scenario, it might for example be unnecessary to have an updated "buddy-list" when the phone is idle with the key-lock activated [13]. However, there is a trade-off between having an updated list and having a reduced number of messages. In a push-based system, it is the server that notifies the client when a user included in the client's subscription has been updated, thus providing better guarantees for an updated list. But, if clients update their status often and/or the buddy list contains many entities, the number of transmitted messages per time period might be higher than with a pull-based system.

In a subset of the simulations of this study, a special type of users employing a presence service on their cell-phones is modeled. The purpose is to investigate how this kind of service utilized in a cellular system affects other traffic.

2.4 Real-time Transport Protocol

The Real-time Transport Protocol (RTP) is a standardized network protocol for audio and video transmission that was developed by the IETF. The initial standard was published in 1996. It was originally designed to be a multicast protocol but has also been extensively used in unicast applications. RTP can carry any type of real-time data and is not dependent on an underlying protocol. It could be applied above

either the Transmission Control Protocol (TCP) or the User Datagram Protocol (UDP), but since RTP is intended for real-time applications and such applications normally are more sensitive to delay than packet-loss, UDP is the usual choice as underlying protocol for RTP.

RTP has become the fundamental protocol in the VoIP industry for transporting media streams and is in this situation normally used in conjunction with SIP for initiating the media sessions and with the RTP Control Protocol (RTCP) for supervision of the media streams. RTCP is a sister protocol to RTP designed to provide out-of-band control information for the RTP flow. It is designed to use a separate UDP port to supply all other members in the media session with feedback on the media quality provided by RTP. Applications may optionally use the information provided by RTCP for such purposes as synchronization of media streams (e.g. audio and video) and quality enhancement through limitations of flow or adjusting codec settings (e.g. low compression instead of high compression).

In this study, MMTel users will be simulated that transmit and receive both a VoIP stream and a video stream. Both of these media streams will be delivered by RTP. In conjunction with the RTP streams, RTCP packets will transmitted according to the specification in the IETF RFC 3550 [14]. This standard defines both RTP and RTCP. RFC 1889 contains the first standardization of RTP but it was made obsolete by the publication of RFC 3550.

2.5 Quality of Service

Quality of Service (QoS) is the concept of trying to provide a particular quality guarantee for a specific type of service. Most commonly, QoS is used in relation to IP based services, since the lack of dedicated channels makes it difficult to certify that a specific service will be granted the bandwidth required to employ it with satisfactory quality. In this study the concept will be used for services in IMS.

In the simplest case, QoS could be achieved by prioritizing all IP packets originating from a service classified as important. Traffic regarded as less important is delayed when load is increased to leave room for the more important traffic, or in the case of extreme load, simply discarded. Only when all the prioritized packets have been granted place in network bandwidth, the leftover space is filled with traffic of less importance. Of course, much more complex prioritization schemes can be applied than this most basic example with absolute prioritization.

Real-time voice and video are two examples of services that require a certain bandwidth to be delivered with decent quality without inconvenient artifacts and delays. To ensure this, QoS mechanisms can be applied in the network so that this traffic is prioritized over other less bandwidth demanding services. In order to distinguish the traffic types and treat them in accordance with a predefined QoS policy, a QoS Class Identifier (QCI) is assigned for each flow and associated with a so called Traffic Forwarding Policy (TFP) [15]. The TFP defines a set of parameter settings regarding traffic forwarding at every node along the path between the end users. By using distinct QCIs for the different services, and pointing this QCI to a certain TFP, the traffic flows can by the TFP settings be given prioritizations and guaranteed a required bandwidth.

Traditionally, and in systems of today, it is the end user (the terminal device) that initiates the QoS, i.e. the client tells the network what kind of service it wants to use and what QCI should be associated with it. This design is based on the assumption that all information about the requested service can only be present in the terminal. There are however, a number of problems with this approach, where perhaps the most severe one is that there is no guarantee that the information the terminal gives about the service about to be used is correct. The terminal could

for example grant all the flows originating from itself the highest prioritization even though the actual services used does not require this. In [15], this complex of problems that exists in today's QoS concept is discussed and an evolved QoS concept that includes a new procedure for network-initiation of QoS is proposed.

One of the main tasks of this thesis is to investigate if QoS guarantees can be given by using various scheduling algorithms. In the next chapter, the basic structure of the network simulator of this study will be described and in its last subsection, Section 3.5, the tools available in the scheduler of the simulator to design QoS-based scheduling algorithms will be presented.

Chapter 3

Network simulator

As the conclusions of this thesis are based on simulation results, this chapter will describe the network simulator in which all of these simulations are performed. The chapter opens with some implementation related basics, and follows with more details on the architecture within the simulator. How specific LTE characteristics were modeled will also be covered, as well as how some more radio related models were designed. The chapter finishes with a presentation of the scheduler structure and the functions therein applied in the simulations of this study.

3.1 Simulator overview

The simulator used in this study is a cellular network simulator developed internally at Ericsson. It is an event driven simulator implemented in the object-oriented programming language Java. Being an event driven system, it runs by sequentially executing events placed in an event queue. The events are positioned in the queue based on their defined time of execution. For example, an event set to happen immediately is placed at the front of the queue and thus treated first. New events will be created and placed into the queue as the event currently executed triggers new events.

The network simulator is based on modules, meaning that the interacting parts of the system can be replaced by equivalently equipped modules, in that way enabling for simulations of specific system setups or specific traffic scenarios. One key type of object in the simulator architecture is the user object. Instances of this object type are created throughout a simulation or at the start of it, based on simulation settings. Each user object contains different modules where one of them is the traffic model. The traffic model specifies how the user will act and thus also defines what traffic that will be produced by this user. By implementing new traffic models and employing them on the generated users, new traffic scenarios can be simulated.

One important aspect of the simulator is the logging functionality. Almost all event parameters can be logged for later consideration. The parameters chosen to be logged are measured and logged throughout the complete simulation or during a specified part of the simulation. At the end of the simulation run, the logs are output to files that are used for subsequent post-processing.

Due to the vastness and complexity of the simulator, not all details of it will be described in this report. In the following sections, only the most relevant aspects of it (considering the scope of this thesis) will be explained.

3.2 Simulator architecture

In Figure 3.1 a simplified overview picture of the simulator is shown. The figure is an instantaneous ”snap-shot” of a running simulator, i.e. the set of objects displayed there is an example of the simulator status at one instant of a simulation. The object set may vary greatly throughout a simulation. For example, the presence of two user generators and the pair of differentiated user types that they generate exist only in the simulations that contain presence users. In the majority of the simulations of this study, only the MMTel users are present.

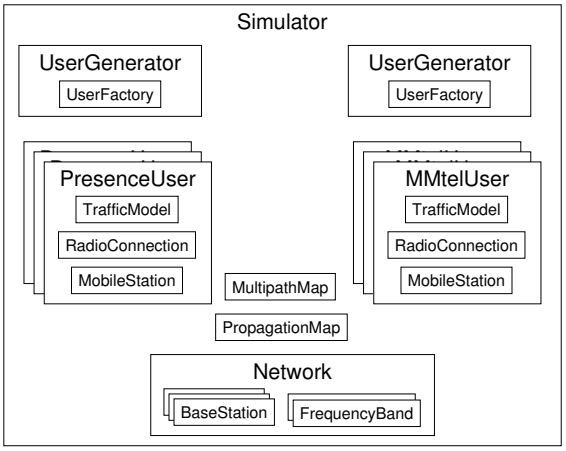


Figure 3.1: Example of a snap-shot overview of the simulator.

Every user instance contains a mobile station, a radio connection and a traffic model. The traffic model of the presence user is the single model described in Section 4.2.2. For the MMTel users, the traffic model is more complicated as the simulated client will act according to a mixed traffic scenario. Here, the model is a set of traffic models, which act side-by-side, as well as a special controller unit, designed to govern how and when the different traffic models shall act. This hierarchical layout with a traffic controller unit and its subordinate traffic models is shown in Figure 3.2. A more detailed view of the architecture inside the three main traffic models can be found in the forthcoming model specific sections (Figures 4.2, 4.7 and 4.8).

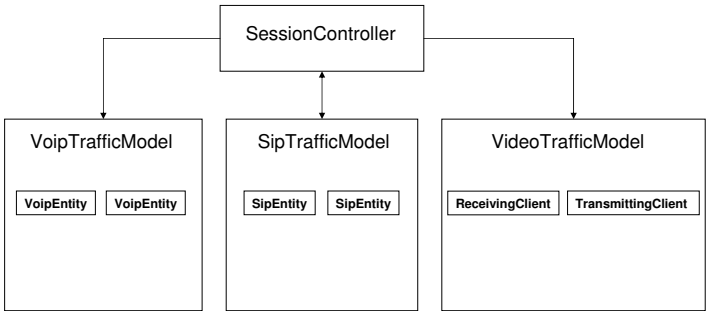


Figure 3.2: The traffic model architecture of an MMTel user.

The radio connection module defines the more radio related aspects of the cell phone of the user instance. The mobile station of every user controls and holds

propagation and mobility parameters. The movement and position of the user is controlled by a sub-object of the mobile station and calculated from the setting of a speed parameter, among other things. The speed of all users in this study was set to 3 km/h. A radio propagation model that defines how signals propagate from the user is also kept in the mobile station.

The characteristics of the complete network that the traffic propagates through is modeled in the network module, seen at the bottom of Figure 3.1. In Figure 3.3, the architecture layout of the modeled network is displayed. In the leftmost position of the figure, the user that this study concentrates on is located. All cell phone and radio connection related characteristics pertain to this object and its communication with the base station. More information on how the connection between the cell phone and the base stations are modeled will be given in Section 3.4. The channel between the base-stations and the core network Gate Way (GW) is simulated in a transport network model, also located in the overall network model. A separate sub-model also simulates the characteristics of the IMS network and the connection to the remote end-user, which is directly connected to the core network. As depicted in the figure, the remote end-user that the cell-phone communicates with is *not* another cell phone with related radio-connections, but rather a terminal located in direct connection with the IMS core network. Since it is the downlink scheduling that is the area of interest for this thesis, it is mainly the characteristics of the traffic sent from the remote terminal to the cell phone that will be investigated.

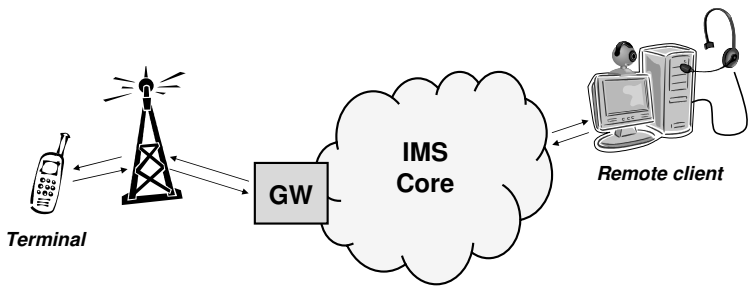


Figure 3.3: The network architecture used in the simulator.

The network simulator also contains functionality to simulate Robust Header Compression (ROHC) [16] and this compression technique was applied on all outgoing IP traffic from all users. ROHC is a standardized method for compression of IP headers and is most vital in wireless networks and for small packets, such as VoIP-packets where the header is a large part of the total size.

3.3 LTE specific settings

With the objective to model the characteristics of the future LTE system, the simulator has been equipped with functionalities required to simulate such aspects as OFDM. In addition to applying this functionality, a number of parameters were set to values that are most probable to be valid for an LTE system. Table 3.1 presents some of the most important of these settings.

The 5 MHz bandwidth setting was selected to make capacity comparisons with HSPA easier (see e.g. [17], [18]). In reality, LTE will hold the flexibility to be used with different bandwidth settings, all according to the desires of the operator. Bandwidths from 1.25 MHz up to 20 MHz can be realized.

Only downlink parameters have been considered and discussed in this section

Parameter	Setting (unit)
Frequency band bandwidth	5 (MHz)
Maximum downlink transmission power	20 (W)
Transmission Time Interval (TTI)	1 (ms)
Number of sub-bands	25
Number of symbols per sub-band	144

Table 3.1: LTE specific settings in the simulator

since it is the downlink capacity that will be measured and analyzed in the simulations.

3.4 Radio model

Geographically, the simulated radio network consists of three base stations with three hexagonal sectors or cells laid out around each, resulting in a total of 9 cells. Each cell has radius of 500 m. In order to avoid border effects and to obtain the same interference load in all cells, wrap-around mapping is applied. Figure 3.4 shows the geographical cell layout used in the simulations.

The model used for such radio phenomena as fading and dispersion is the 3GPP Typical Urban model.

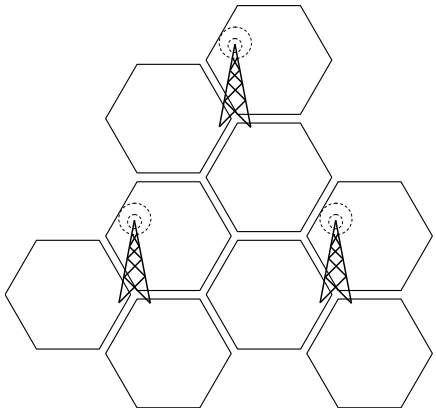


Figure 3.4: The geographical cell layout of the radio model.

Hybrid Automatic Repeat reQuest (HARQ) [19] was modeled to handle communication errors between the cell phone and the base station. The retransmission interval was set to 6 ms and the maximum number of re-transmissions was set to 10.

3.5 Scheduler

This section will only describe the downlink scheduler of the simulator, since it is on the downlink that all measurements of this study are performed.

The network simulator of this study implements a scheduler with an additive weight system where weights are allotted to send-queues on account of their characteristics and of the Service Data Units (SDUs) currently located therein. Scheduling decisions are periodically performed with respect to these weights. The queue with the highest weight is first granted a so called Resource Block (RB) in the current

TTI. Weights are then recalculated for all queues and the SDU now holding the highest weight is scheduled for another RB in the same TTI. This is repeated until all RBs within the TTI are filled. The RB concept and the scheduling over both time and frequency domain, which is specific for OFDM, is visualized in Figure 3.5. The size of an RB is defined both by the length of a TTI, the total available bandwidth and the number of sub bands it is divided into. All of these values are presented in Table 3.1.

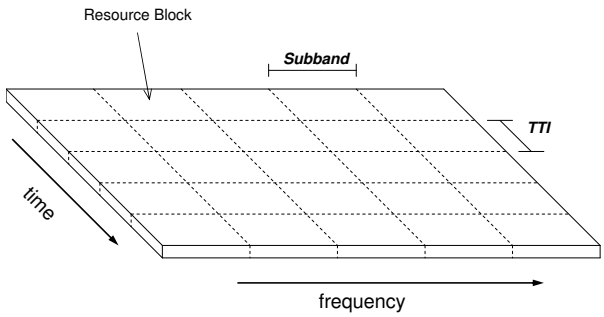


Figure 3.5: Scheduling is performed in both time and frequency domain.

Each type of traffic produced in the scenarios of this study is linked to a QoS-dependent priority class, which holds a configuration that specifies how weights for the SDUs of this traffic flow is to be calculated. For every receiving terminal, there is one send queue each for every priority class, containing packets of traffic types connected to this class that are about to be transmitted. More than one traffic type may be associated to a priority class. At the beginning of every scheduling period, weights are calculated for each queue through the application of a weight calculation formula defined in the priority class that the send queue is a part of.

The design of the simulator allows for the use of a number of factors to affect the scheduling weights, but in the simulations of this thesis, only four weight factors are employed. Below they are presented under a separate heading each.

Fixed weight bonus

The first and most basic weight factor is a static bonus value that can be assigned to all queues of a priority class. Absolute prioritization of a specific traffic type can be achieved by awarding its class a bonus weight of such magnitude that it cannot be exceeded by other priority classes.

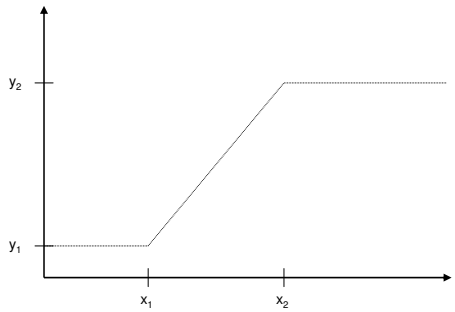


Figure 3.6: The form of the linear equations used in weight calculations.

Scheduling delay weight

An additional weight bonus can be determined with respect to the time that has passed since a queue was previously scheduled. A linear equation of the type depicted in Figure 3.6 is applied to calculate the size of the bonus. The x-axis represents the time since the previous scheduling and the y-axis the resulting weight bonus. For every QoS-dependent priority class, the values of *x1, x2, y1* and *y2* can be set independently.

Packet age weight

The third weight bonus is dependent on the age of the SDUs within the send queue. The factor is much similar to the delay weight factor, but here it is the age of the oldest pending SDU in the queue that relates to the x-axis values of Figure 3.6. As with the scheduling delay weight formula, the upper and lower limits for x- and y-values can be set.

Re-scheduling weight

The last weight factor, the re-scheduling weight, can be used to save control overhead by scheduling fewer users per TTI. This is achieved by granting a static weight bonus to the queues that have already been scheduled within the current TTI. If the weight bonus is set high enough (higher than the maximum achievable sum of all other bonuses), the queue scheduled first in a TTI will be allowed to use all sub-bands for transmission if it needs to. By setting it slightly lower, a less greedy approach that still saves overhead by favoring already scheduled users, can be acquired.

The next chapter describes the traffic models by which the different traffic flows are generated.

Chapter 4

Traffic models

This chapter offers first a presentation of the two traffic scenarios that are employed by the simulated users of this study. Each traffic scenario relates to one type of user, and the user types are presented in conjunction with the scenarios. In the subsequent sections of the chapter, all of the traffic models utilized in the traffic scenarios are described in one section each.

4.1 Traffic scenario

In this section, the scenarios that the virtual users act by and the IP-traffic that they create are described. As there are two different types of users, there exist also two traffic scenarios, which are presented in a subsection each. First, the scheme of events that all standard MMTel users follow and the mixed traffic flow that originates from them are described. The second type of users, the presence users, appears only in a subset of all simulations and produces only one type of traffic. More details on their behavior is given in Section 4.1.2.

4.1.1 MMTel users

All users of the main type act by the same scheme of events, i.e. the multi-media communication sessions that they engage in follows the same predefined pattern. This is done to make capacity and performance analysis more easily accomplishable. The IP-traffic generated in this session is mixed and consists of data originating from different types of services. The following are the main traffic types produced by each MMtel user in the simulations:

- SIP traffic for MMTel session setup.
- VoIP traffic for voice conversation.
- A real-time video stream as a compliment to the audio conversation.

In addition to the above traffic types, an RTCP flow is generated for each of the media streams. The modeled RTCP stream imitates the traffic generated by the sister control protocol to RTP, by which both of the media streams are delivered.

Figure 4.1 depicts the life time of a virtual MMTel user along a time-line. The figure shows when and in what combinations the different traffic flows are active.

The following "story-line" may be used to describe the events in the life of a modeled user: A client, walking in an urban environment, receives a call on his/her LTE cell phone. The phone is answered 3 s after the first ring-signal has sounded and a voice conversation is then initialized with the calling part. The caller is

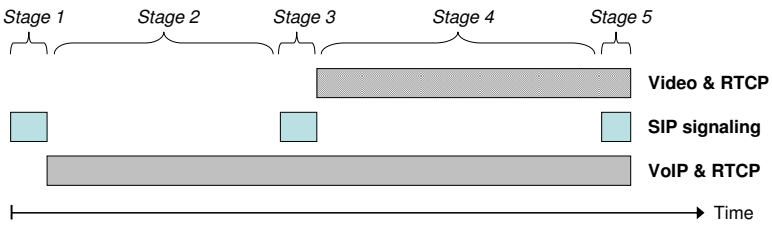


Figure 4.1: The life time events of a user in the simulation.

located at home, using a VoIP service on a personal computer. After having spoken for 15 s, the caller wants to share a visual experience through the use of the web camera and he/she thus sets up a video stream (simplex video) to the other user. When the user has received the real-time video-stream for 10 s, the end-parts say goodbye and the remote part terminates and thereby closes the MMTel session.

Each scenario stage, as they are shown in Figure 4.1, relates to a set of traffic types that is sent and/or received during that period. The conversation stages (Stage 2 and 4) are assigned static duration values (15 s and 10 s) while the setup stages (Stage 1, 3 and 5) are dynamic and dependent on the time it takes for the SIP message exchange to complete. The RTCP traffic flows are active when their related media streams are active.

4.1.2 Presence users

The presence users only produce one type of traffic, namely SIP traffic used for presence information exchange. The traffic consists of server registrations, subscription messages and status updates both to and from the presence server.

Presence users are by definition either online or offline and it is only when they are online that they are active and send SIP messages. In the offline stage the only possible activity is the act of going from offline to online, which is done by registering to the server. If the user on the other hand is in the online stage, the following four events may occur:

- Publication of personal status update.
- Notification on a status update of a ”buddy”.
- Re-registration to server.
- De-registration (going offline).

A personal status update means that the client changes his/her availability status, e.g. going from ”online” to ”busy”. When this happens, the user must publish the new status to the presence server. When another client in the ”buddy-list” of the presence user changes status, a notification on this is received from the server. If no messages have been exchanged between the client and the server within an hour, the client must re-register to the server in order to avoid being dropped and not being sent any more notifications. Finally, the user may go offline, which is done by de-registering himself/herself to the server.

4.2 SIP model

The protocol used for signaling purposes in the simulations is the widely used Session Initiation Protocol (SIP) [10]. A brief background on SIP was given in Section 2.2.

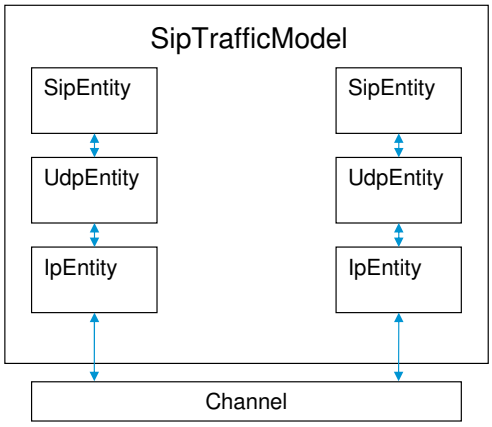


Figure 4.2: The architecture of the SIP traffic model.

The SIP model is applied in two different variants in this study. The first one is for media session setup and the second is for presence information exchange. The presence model is described in more detail in Section 4.2.2 and the session setup model is presented in Section 4.2.1.

In Figure 4.2 the general architecture of the SIP model is shown. The *SipEntities* represent the application layer and is thereby the source and receiver of SIP messages. As seen in the graphical representation, they are placed above a UDP layer. The fact that SIP is modeled to operate over UDP, which is an unreliable transport protocol, means that there is no guarantee for a message to reach the destination client. To ensure that the setups are completed even if some packets are lost, a retransmission scheme was implemented at the layer above UDP, i.e., in the *SipEntity* module. Not all messages are crucial and require retransmissions. The ones selected to be retransmitted if no answer to them is received are printed in bold in the flow charts of the subsections below (Figures 4.3, 4.4 and 4.6). The retransmissions were scheduled as follows: The first retransmission occurs 500 ms after the original message was sent if no reply was received. The second 1 s after that, the third after 2 additional seconds and the fourth after 4 more seconds. After that, each time-out delay is 4 s. These values were taken from [20].

Figure 4.2 can be related to the network architecture of Figure 3.3 in the way that one of the *SipEntities* is located inside the cell phone terminal while the other is placed in the personal computer at the remote end. The channel, as it is depicted in Figure 4.2, is all of the intermediate network models of Figure 3.3.

With the objective to measure the performance and load on the system with SIP in use, the size of the messages is an important issue. In a real situation, the size of the messages is variable and dependent on their information content. The content may include client and session specific parameters, normally presented using the Session Description Protocol (SDP) [21]. Since this information is of no importance in the model, and because the variations may be due to factors not simulated, the information field was left out and each message type was assigned a static size. In order to get realistic sizes on the SIP-messages numbers from previous studies were

used. In Table 4.1 all SIP messages are displayed along with their implemented sizes.

Message Type	Size (bytes)
183 SESSION IN PROGRESS	1270
INVITE	1113
PRACK	1014
200 OK	890
180 RINGING	888
BYE	878
ACK	427
PUBLISH	800
NOTIFY	700
SUBSCRIBE	600

Table 4.1: SIP message sizes in the model.

4.2.1 Session setup

In the process of setting up or terminating a media session between two entities, messages are exchanged following a predefined order. When a specific message is sent to a peer at a certain stage of a setup or termination, a reply in the form of another specific message type is expected. To complete a setup, a number of such transmissions back and forth must be performed. By these multiple so called handshakes, all parameter settings for a session are agreed upon between both concerned parts. In this section, the message flows that occur between the peers of all setups and terminations will be described.

By strictly following the proposed traffic model the number of message flows necessary to model can be limited to only two. These are the SIP INVITE method (the same flow can be used for SIP reINVITE when starting the video-stream, only with a few changes in delay parameters) and the SIP BYE method.

Invite

The Invite method is applied when a media session is to be set up between two end parts. It can either be the first initiative to communication, like in a ”call-up”, or it can be used between two clients already engaged in communication to re-negotiate the session setup or to initialize a new media stream. In the latter case, it is referred to as a re-invite.

In Figure 4.3 a schematic representation of the message flow when establishing a session can be seen. What is represented there is a non-interrupted communication where all the messages reach the receiver as planned. In the case of a lost packet, the messages with names printed in bold would be retransmitted in the fashion described in the previous section, Section 4.2.

The boxes seen in the session setup flow charts represent delays that would arise in a corresponding real-world situation. The values that these delays are set to in the simulations can be found in Table 4.2. In the case of a re-invitation (as when the video stream is initiated), the radio bearer delay is removed[1] and the pick-up delay is replaced by a default delay. In order to separate reception and reply transmissions, a very small process delay is also applied in every transmission procedure.

[1] This is due to the fact that in this stage, the radio bearer is already established.

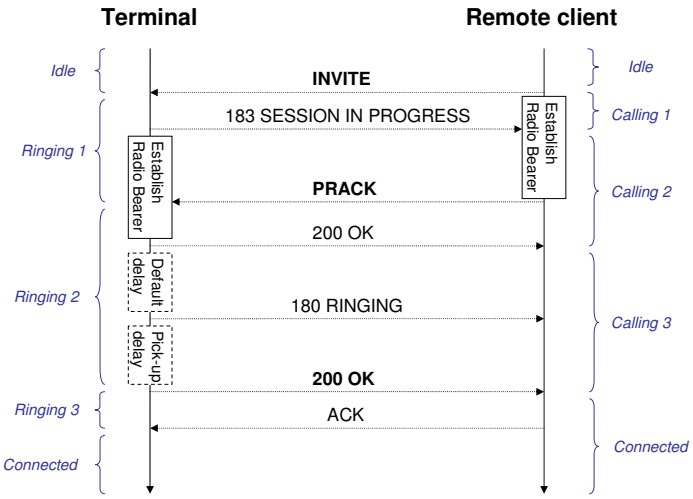


Figure 4.3: Schematic view on the modeled SIP INVITE method.

Variable name	Value (unit)	Description
$radioBearerEstablishTime$	600 (ms)	The time it takes to establish a radio bearer.
$pickUpDelay$	3 (s)	The time between a ring-signal is produced at the receiving end and the time the phone is picked up.
$processDelay$	1 (ms)	A small process delay used to separate receptions and answer transmissions.
$defaultDelay$	0.1 (s)	A default delay applied in SIP processes. Summarizes process delays that arise in the terminals.

Table 4.2: Delay variables applied in the SIP model.

Bye

In Figure 4.4 the message flow of the SIP BYE method is depicted. This method is invoked when a media session is to be terminated. As seen, this flow is much less complicated than the SIP INVITE method.

Figure 4.4: Flow of messages when the SIP BYE method is invoked.

In order to determine where in a SIP session an entity in the model is at a particular moment and how to react when receiving a message type, the model was constructed as a state machine, entering different states at the various stages of a media session setup or termination. The states entered at the stages of a setup are shown in the flow diagrams of Figures 4.3 and 4.4 in the leftmost position for the *Terminal* and in the rightmost position for the *Remote client*. Furthermore, in Figure 4.5 the state machine is depicted in a state chart. Here all the possible

states in the model are displayed inside the ellipses and, on the arrows connecting them, the messages that need to be received or the methods needed to be invoked in order to go from one state to another is shown.

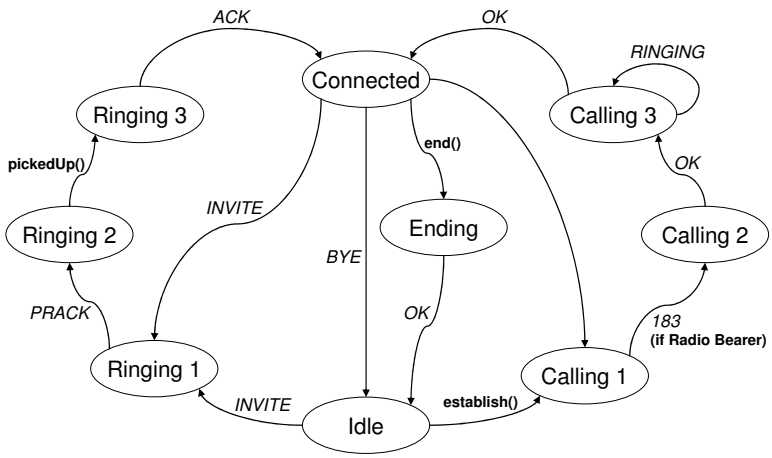


Figure 4.5: State chart for the SIP model.

4.2.2 Presence model

The presence model was based on SIP. This means that SIP messages are used to send notifications on clients' availability status, such as "online", "offline", "busy", "out of office" and so on. IETF has defined a standard extension package for SIP in order to make it suitable for presence signaling. A specification of the standard can be found in [11]. This implementation was designed to follow this standard.

As mentioned in Section 2.3 there exist two different approaches regarding how updates are to be distributed from the presence server. Since push-based systems are the most common today, the model implementation was designed to follow that approach. Whether or not a pull-based system might be more suited for cellular networks is not covered in this thesis, but discussions on this can be found in [13].

The presence model was constructed to simulate traffic between one presence client and one presence server. The presence client and server are in relation to each other placed in the network architecture as the terminal and the remote client are in Figure 3.3. The client uses the presence server on his/her cell-phone and connects to the server placed inside the IMS network core.The events that invoke the message flows are the following: Registration, Publishing, Notification, Re-Registration and De-Registration. All of the events are scheduled to occur with exponentially random intervals and, in the simulations of this thesis, with the mean values presented in Table 4.3. In the subsections below, all of the events are described in more detail.

Registration

When a client is about to enter a presence session, a registration to the presence server must first be performed. It is first when the registration has completed that the client can start receiving updates on other clients' availability. The registration is performed by the sequence of message exchanges that can be seen in Figure 4.6. As in previous figures depicting message flows, the messages with names printed in bold are transmitted repeatedly after a time-out if no answer is received. The

Interval	Length	Description
Publish interval	20 (min)	The mean value of intervals between publish events from the client, i.e., how often does a user update its status in general.
Notification interval	80 (s)	The mean time between notifications from server. The value is based on the assumption of a buddy-list length of 15 clients with the same status update interval as this user (20/15 (min)= 4/3 (min) = 80 (s)).
Online duration	6 (h)	The mean time that a user remains online.
Offline duration	6 (h)	The mean time that a user remains offline. None of the other intervals are valid while off-line, since no such events can occur.
Server time-out	60 (min)	If the server has not received any messages from the client within this interval, the subscription is dropped.
Re-registration interval	55 (min)	If no messages have been sent to server within this interval, a re-registration is executed. Set to occur before registration time outs.

Table 4.3: Activity parameter settings for the presence users.

states that the entities enter throughout the sequence can be seen in the left-most and right-most position of the figure.

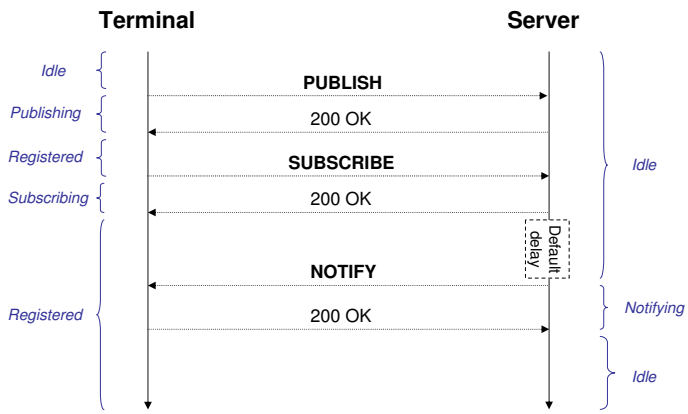


Figure 4.6: Schematic flow of messages when setting up presence over SIP.

In the first part of the registration, the client sends a *PUBLISH* message to the server where the status of the client is expressed. The server status is updated with the provided information. Since the purpose of the model is only to produce a realistic traffic flow between a client and a server, the messages do not need to contain the status information here. Instead, when the client's publication reaches the server in the model, the server only updates a boolean flag in its state, indicating that the client has registered to the server. The server responds with a *200 OK* message to acknowledge that the message has been received.

In the second part of the registration, the client tells the server what information it wants to receive notifications on, i.e., which other clients it has on its buddy-list, by sending a *SUBSCRIBE* message containing this information. The list is stored at the server, and when any of the entities in the list announces a change of its status, the subscribing client will receive a notification. As with publish, the subscribe message in the model does not actually contain any information regarding a subscription list, but rather just has a static size in order to model a default case. However, the server registers that the client is now subscribing by setting a

subscription flag to true. Again, the server responds with a *200 OK* message when the subscription has been registered and approved.

The last part of the registration process is for the server to notify the now subscribing client on the status of all the clients in the buddy-list. This is executed by sending a *NOTIFY* message containing status of all clients subscribed to. As with previous messages, no real information is sent in the implemented model since it bears no importance for the simulation. The client acknowledges the notification with a *200 OK*.

The default delay, applied between the *200 OK* response to the subscription and the *NOTIFY* from the server represents data processing time at the server. It is also necessary to keep them from being sent at the same instant of time. The default delay and the process delay values are general for the SIP models and are given in Table 4.2.

Publishing

Whenever the client updates his/her availability status, e.g. going from "online" to "busy", the new status must be published to the server in order for the server to propagate the correct status to other subscribing clients. The publish is performed by sending a *PUBLISH* message to the server, which is acknowledged by a *200 OK*. The updates of an active user were scheduled to occur with exponentially random intervals and with the mean value of 20 minutes.

Notification

Since the presence model is push-based, the server notifies the client whenever an update is received from a client in the buddy-list. The notification is performed with a *NOTIFY* message from the server, which the client acknowledges with a *200 OK* after reception. Since the model only includes one client and one server, no actual updates are received from other users than the one in the model. In order to get a realistic number of notifications, notifications are scheduled to occur at the server with respect to the same mean value used for publishing events at the user and a static number representing the number of users in the buddy-list. The list length was set to 20. By dividing the mean length of a publish interval with the number of users in the buddy-list, a mean value for notification intervals is found. This simulates that every user in the list updates with the same mean interval as the client in the model.

Re-Registrations

Every subscription at the server has an expiration time, and if the client has not re-registered to this subscription at that time, the subscription is dropped. This is to avoid that the server keeps sending updates to users that are not active anymore, i.e. to clients that have left the presence session without announcing their departure. In order to avoid getting the subscription dropped, the client must re-register within the time-out interval. This is under the condition that no other communication with the server has occurred during this period of time. The re-registration is done in the same way as the initial registration described above and depicted in Figure 4.6. The subscription expiration time was set to 60 minutes and re-registration was set to be executed every 55 minutes in order to occur before the time-out at the server.

De-Registrations

When a client goes offline and exits the presence service in a correct fashion, a de-registration is performed with the objective to inform the server that this client

is no longer active and shall receive no more notifications. A de-registration follows the exact same procedure as a registration (see Figure 4.6). In a corresponding real-world situation, the first publish indicates that the new user status is ”offline” and the subscription message ends the subscription that the client holds.

4.3 VoIP model

The model for the voice conversations models both how two parts in a conversation interact with each other as well as more VoIP specific characteristics. The implementation simulates the speech codec AMR12.2 [22].

4.3.1 Model architecture

The architecture of the VoIP model is depicted in Figure 4.7.

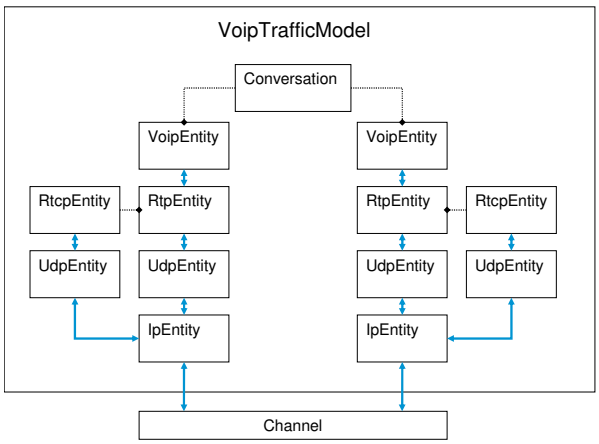


Figure 4.7: The architecture of the VoIP traffic model.

The traffic model contains two entities that send and receive VoIP frames, one associated with the cell phone terminal of Figure 3.3 and the other with the personal computer terminal of the same figure. In order to control how and when the users talk to each other, a conversation controller (referred to as *Conversation* in Figure 4.7) was applied above the entities.

The VoIP frames are sent by RTP, which in turn is applied above UDP. In order to overview and control the RTP flow an RTCP module was bundled with each of the RTP entities. The RTCP entities send RTCP reports over a separate UDP port but with the same IP.

4.3.2 Conversation control

Initially, when a new user with a VoIP traffic model is created, the upcoming voice activities of the user is configured based on the settings of a group of parameters. In Table 4.4 the parameter settings used for the entities in the simulations are presented.

Start conversation

When the conversation is started the intermediate period between talk spurts of the clients is calculated and set based on the setting of $VoiceActivity$ and

Parameter name	Short	Value (unit)	Description
TalkSpurtDuration	t_{spurt}	2 (s)	The mean length of a talk spurt
VoiceActivity	a	0.5	The mean rate of the voice activity for each of the clients. A value of 0.5 means that one user is always talking.
EncodingDelay	d_{enc}	15 (ms)	The time it takes to perform encoding at the sending part.
DecodingDelay	d_{dec}	5 (ms)	The time it takes to perform decoding at the receiving part.
FramePeriod	p	20 (ms)	The inter-frame time, i.e. the time between voice frame transmissions.
FrameSize	s	264 (bits)	The size of the voice frames. Octet-aligned value. Set with respect to the RTP payload format.
MaxDelay	d_{max}	150 (ms)	The maximum VoIP frame delay (mouth-to-ear). Later packets are regarded as lost.

Table 4.4: Parameter settings in the VoIP model

TalkSpurtDuration. The intermediate period is the rate of the conversation that will be silent (according to the *VoiceActivity* setting) multiplied with the mean talk spurt duration. If the voice activity is less or equal to 0.5, as it is in the scenario of interest, the following formula will be used to calculate the intermediate period (denoted t_{inter} in formulas henceforth):

$$t_{inter} = t_{spurt}\frac{1-2a}{2a} \tag{4.1}$$

The calculated value of t_{inter} will then be applied as the mean value to an exponential random generator for intermediate period lengths. In the case of interest when $a = 0.5$, the mean time between talk spurts will by application in Equation 4.1 be 0, which means that one client will always be talking.

An exponential random generator for talk spurt lengths will be assigned the mean value of *TalkSpurtDuration*. When the voice activity parameters have been calculated and set, one of the entities is randomly chosen to start talking. The traffic model assigns a state transition to the state-holding module *Conversation* that updates the state for the client that starts to talk. As the state is transformed from silence to one of the clients talking, the conversation module also tells the entity in question to start sending voice frames.

At the time that the conversation is started, an event that starts an intermediate period between talk spurts is scheduled. The instant that this event occurs is generated from the random generator for talk spurts described above.

Start intermediate period

When an intermediate period is entered, the state in *Conversation* is set to mutual silence and both clients are set to stop the transmission of voice frames. Based on a value given from the exponential random generator for intermediate period durations, an event that starts a talk spurt is scheduled to occur. As the length of the intermediate period in the simulations of this study is always 0, a talk spurt will immediately be started when an intermediate period is entered.

Start talk spurt

When a new talk spurt is about to be started, a check on which client spoke most recently is performed. If client A sent a talk spurt most recently, then client B will start sending voice frames. If the opposite is true, client A will be the one

to send a talk spurt. The length of the talk spurt is again given from the random generator with the mean value of $TalkSpurtDuration$. At the end of the talk spurt an intermediate period is again started.

This switch between talk periods and intermediate periods is repeated until the VoIP conversation is ended.

4.3.3 Frame transmission and reception

Each VoIP entity has a frame timer scheduled to periodically time out with the interval given by the setting of $FramePeriod$. At each time out a check is performed whether or not this entity is talking. If it is talking, a frame of the size $FrameSize$ is sent to the peer client. The value of $FrameSize$ is set with respect to the RTP payload format for AMR defined in RFC 3267 [23]. A sequence number is assigned to the packet and the time of transmission is set in the frame so that the delay can be computed at the receiver.

When a voice frame is received at the VoIP entity, the delay of the packet is first calculated by application of Equation 4.2 where d_{trans} denotes the transport delay[2] and d_{tot} the total delay experienced by the end user.

$$d_{tot} = p + d_{enc} + d_{trans} + d_{dec} \tag{4.2}$$

If $d_{tot} > d_{max}$ the frame will be considered as lost and logged as such. Otherwise it is logged as received. In either case, the delay of the packet will be logged for later analysis.

4.4 Video streaming model

The video model simulates real-time video streaming from a transmitting client to a receiving client. In the scenario of this thesis, it is the computer user of Figure 3.3 that transmits the video stream, e.g. from his/her web camera, and the cellphone that is the receiver of the real-time stream. The video codec mimicked by this video model is the International Telecommunication Union (ITU) standard H.263 [24].

4.4.1 Model architecture

As seen in Figure 4.7; displaying the general architecture of the video streaming model, there is one video transmitter and one receiving client in the traffic model. The video is sent from the transmitter to the terminal over RTP with an accompanying RTCP stream for surveillance of the RTP. RTP and RTCP are sent over separate UDP ports. Noted should be that video traffic is only sent in one direction while RTCP reports are sent in both directions.

4.4.2 Video transmitting client

When the video streaming is started in the traffic model, a frame timer is started at the transmitting client. The timer is scheduled to expire periodically with the interval $1/r_{frame}$, where r_{frame} is the frame rate of the video stream. The value of r_{bit} can be found in Table 4.5 along with a selection of other important parameter settings in the streaming traffic model.

At each frame timer expiration a video frame is sent. The size of the frame is set with respect to a number of factors, such as the streaming bit rate, current state

[2]The time between the transmission and the reception of the packet at the transport layer.

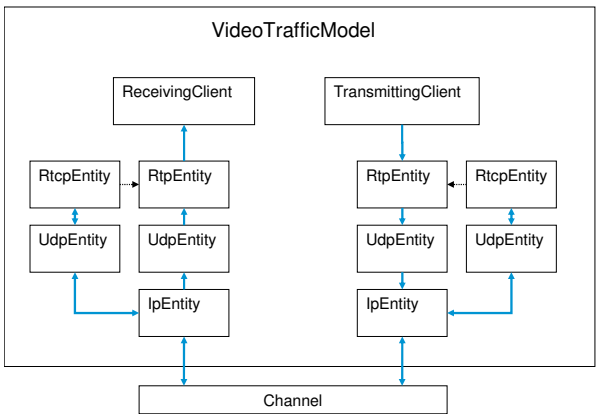


Figure 4.8: The architecture of the video streaming model.

Parameter name	Short	Value (unit)	Description
StreamingBitRate	r_{bit}	220 (kbps)	The rate at which the video stream is sent (in kilo bits per second).
FrameRate	r_{frame}	12.504 (fps)	The rate at which video frames are transmitted from the streaming side (in frames per second).
EncodingDelay	d_{enc}	40 (ms)	The time it takes to perform encoding at the sending part.
DecodingDelay	d_{dec}	40 (ms)	The time it takes to perform decoding at the receiving part.
MaxDelay	d_{max}	250 (ms)	The maximum video frame delay (end-to-end). Later packets are regarded as lost.
PlayOutFrequency	r_{play}	60 (Hz)	The frequency at which frame updates are checked for. Set to be the frame timing of the display device.

Table 4.5: Parameter settings in the video model

(changed throughout the streaming session) and a randomization factor. The end result of the employed algorithm is a stream of video frames with varying size but with an overall bit rate (if calculated over a sufficiently long period of time) equal to that given by the setting of $StreamingBitRate$. Every video frame is given a unique sequence number generated from a sequence number counter that is incremented by one each time a frame is sent.

If a single video frame is larger than the Maximum Transmission Unit (MTU[3]) of the underlying UDP layer, the frame is fragmented into frames with the same size (or smaller, if it is the last in the fragmentation series) as the MTU. The MTU of UDP was in the simulator set to 1600 bytes. All fragmented frames have the same original sequence number, allowing them to be identified and concatenated at the receiving end.

4.4.3 Video receiving client

The receiving client in itself does not send any packets; instead it is only the receiver of the video stream originating from the transmitter. When a frame is received at this client, it is first concatenated back into the original fragmented

[3]MTU is the largest frame size that can be transmitted over the network.

frame. That is, if it is not already a complete frame. In the concatenation process, the fragmented frame is checked against a buffer for fragmented frames to see if all the other parts of the original frame are present there. If so, the original frame is restored by combining all its parts and then placed in a play-out buffer. If not all parts of a frame have been received, the fragment is placed in the fragmented frames buffer to await the reception of the remaining parts.

At the reception of the first frame, a play out timer is started. It is set to periodically expire with the interval of $1/r_{play}$. The variable r_{play} is the play out frequency, and the setting of it can be seen in Table 4.5.

At the play-out timer's expiration, the play out buffer is checked. If there are any frames available there, the one with the lowest sequence number is played out and removed from the buffer. When the frame is played out, the delay of the frame is calculated by applying Equation 4.3, where d_{trans} is the measured transport delay (the time between the video frame is split and sent until it has been concatenated at the receiving client).

$$d_{tot} = d_{enc} + d_{trans} + d_{dec} \tag{4.3}$$

If $d_{tot} > d_{max}$ the packet is delayed too much and is regarded as lost. All calculated delay values are also logged in a histogram for later analysis.

4.5 RTCP model

The RTCP model was constructed in accordance with the IETF RFC on RTP [14].

For each RTP stream in the traffic model (the VoIP stream and the video stream), an accompanying RTCP stream is generated. A module connected with the RTP entity supervises the entity for incoming and outgoing RTP packets, and transmits control packets for the stream to the corresponding module on the other side. RTCP packets start to flow when an RTP stream is identified and the transmission of control packets is discontinued if the media stream stops. The RTP entity is only aware that it is supervised and is not involved in what the controlling module sends and receives. Each time an RTP packet is sent or received, a notification message is sent to the observer, i.e. the RTCP entity, which thereby decides on what action to be taken.

4.5.1 RTCP packet types

The model was designed to send five different types of packets: Sender Reports (SR), Receiver Reports (RR), Source Description Items (SDES), Bye packets (BYE) and Compound packets. The Compounds are sets of packets, containing one or more of the above types. In Table 4.6 the different packet types can be seen along with a specification on how their sizes were set (n = number of identified senders in the media session). In addition to these four packets, support was implemented for a sixth type, in accordance with the IETF specification, the Application specific packet (APP). However, since no application specific functions were required in the scenario, this packet type was never used. Also, due to traffic model specifications, no Bye packets were required to be sent and therefore left out of the simulation as well.

Sender reports

SRs are sent from clients that are classified as senders. The RTCP client is classified as such if it has sent an RTP packet since the 2nd previous report was

Message Type	Header Size (bytes)	Content Size (bytes)
Sender Report (SR)	8	$20 + 24 * (n - 1)$
Receiver Report (RR)	8	$24 * n$
Source Description Item (SDES)	4	$8 * n$
Bye Packet (BYE)	4	4
Application Specific (APP)	12	-

Table 4.6: RTCP message sizes in the model

sent. A sender report is defined to contain reception and transmission statistics from all senders that it knows of (in this one-to-one scenario, there can only be one or two senders). In order to hold the complexity low, and because these statistics are logged in other parts of the model, no such information was included in the sender reports. However, the packet sizes were designed as if containing this information and with the number of known senders in consideration.

Receiver reports

RRs are very much similar to SRs, but are sent from clients that are not defined as senders. That is, clients that have only received and not sent RTP packets for a specified interval. An RR is defined to contain the same information as an SR except that five words (20 bytes) of sender information are omitted.

Source description items

Source Description Items hold information on the sources that the RTP stream is transmitted from. A required part of the SDES is a Canonical Name (CNAME) for each identified source. Unlike the description tag for every source in the SR and RR, the CNAME will not change due to conflicts or program restarts and it is therefore required to keep track of each participant. In order to make sure that a CNAME also is unique, they should be derived algorithmically and not entered manually. Typically, a CNAME has a format like "user@host". CNAMEs for sources can also be used for the purpose of synchronizing different media streams to each other, such as lip-syncing a video with an audio stream. None of the information included in a SDES was necessary to transmit in the simulation since the participants could be identified by other means, but the size of the item was set as if the described information was included.

Bye packets

Bye packets are sent at the end of a media session to signal that the participant is about to end his/her participation. When a bye packet is received, the originator of that packet is removed from all member lists. A bye packet should always be sent last; otherwise the client will reappear in the member list as soon as a later sent packet is identified. If a bye packet is not sent at the end of a session, or it is lost, a time out will occur at the other parts when no RTCP packet has been received for a period of time, and the participants will then be removed from member lists in that case also, only at later time. As pointed out above, no bye packets are sent in the scenario of this work. Following the specifications of the traffic model, the two communicating clients close their conversation and applications at the same time (can be seen as a "goodbye exchange" on the voice channel followed by simultaneous "hang-ups") and the media session termination is completed by using SIP. No additional BYE packets in the RTCP stream are necessary.

Compound packets

Compound packets are multiple RTCP packets of the types described above; stacked to form one compound RTCP packet. All RTCP packets that are sent in this model are compounds, always containing at least one report packet (SR or RR) and an SDES packet. This is in accordance with the RFC [14].

4.5.2 Transmissions

In order to determine when to send RTCP packets, a state must be kept at each client. The most vital variables that this state holds are displayed in Table 4.7 along with descriptions.

Variable name	Description
tp	The last time an RTCP packet was transmitted from this entity
$pmembers$	The estimated number of media session members the last time the next transmission time was calculated
$members$	The most current estimate for the number of session members
$senders$	The most current estimate for the number of session senders
$rtcpBw$	The target RTCP bandwidth, i.e., the total bandwidth that will be used for all RTCP packets by all members in this media session. A specified fraction (optimally around 5% of the RTP bandwidth) of the session bandwidth provided to an application at startup. In bytes per second
$weSent$	A boolean flag that is true if any RTP packets have been sent since the 2nd previous RTCP packet was transmitted
$avgRtcpSize$	Average size of the compound RTCP packets that have been sent and received by this participant, in bytes
$initial$	A boolean flag set to true if the application has not yet sent any RTCP packets

Table 4.7: Variables kept in RTCP state

In addition to these variables, a transmission timer and a list of all media session members and a list of all senders is kept in every state as well. The current time, which also is needed for transmission computation can be obtained through a call to a simulator function and will be denoted as tc below.

Every time an RTP packet is sent or received, a notification is sent to the RTCP module. If it is the first notification, the client has just entered the session and an initialization procedure is performed. In this procedure, the state variables are set to initial values and in accordance with media session specifications. If the session is entered as a sender, $senders = 1$, $members = 1$, $weSent = true$ and the entity adds itself to the sender list. Otherwise $senders = 0$, $members = 1$, $weSent = false$. In either case, the entity also adds itself to the member list. After this, a transmission interval is calculated and the transmission timer is set to expire after the calculated interval.

Transmission interval calculation

The transmission interval between reports is calculated following a procedure designed with the purpose of scaling the bandwidth use with the number of participants and senders in a media session. The algorithm is adopted from [14].

- First, if the number of senders is less than 25% of all members in the system, the interval will depend on whether this entity is a sender itself, determined by the value of $weSent$. If a sender, the constant C will be set to: $avgRtcpSize/(rtcpBw * 0.25)$ (i.e., the average packet size divided by 25%

of the bandwidth) and the constant n will be set to the number of senders. If the participant on the other hand is not a sender, the constant C will be set $avgRtcpSize/(rtcpBw * 0.75)$ (i.e., the average packet size divided by 75% of the bandwidth) and the constant n is set to the number of receivers $(members - senders)$.

- Otherwise, if the number of senders is greater than 75% of all members, no distinction will be made between senders and receivers. The constant C is set to: $avgRtcpSize/rtcpBw$, and n is set to the number of members in the session.

- The next step is to set the constant $Tmin$. If the participant has not yet sent any RTCP packets (the variable *initial* is true), $Tmin$ is set to 2.5, otherwise to 5.

- The deterministically calculated interval referred to as Td is set to: $max(Tmin, n * C)$.

- A random number between 0.5 and 1.5 is generated and multiplied with Td in order to avoid synchronization of transmissions between participants. The calculated value is assigned to T, which finally is divided by $e - 3/2 = 1.21828$ to compensate for the fact that the timer reconsideration algorithm converges to a value of the RTCP bandwidth below the intended average. T is the calculated transmission interval.

Transmission timer expiration

When the transmission timer expires, a current transmission interval T is calculated using the method described above. This new interval is compared with the previously applied expiration time in the procedure described below. The procedure makes sure that in the case that state variables have changed since the transmission interval was first calculated; the interval will never be too short considering updated variables:

- If $tp + T$ is less or equal to the current time, an RTCP report is sent, tp is set to the current time and the timer is set to expire again after the interval time T.

- If on the other hand $tp + T$ is greater than the current time, no RTCP packet is transmitted but the timer is instead set to expire again at $tp + T$.

- At every expiration, the variable $pmembers$ is set to $members$.

- A check for time-outs of other participants in the media session is made at every timer expiration by applying the following scheme: A deterministic interval Td for a receiver (with *weSent* set to false) is computed by using the algorithm described in the section above (no randomization applied). Every member in the member list that has not sent an RTP or RTCP packet for five transmission intervals ($tc - 5 * Td$, where tc is the current time), is removed from the list. A similar check is performed on the sender list. Any sender that has not sent an RTP packet within the last two RTCP report intervals (since time $tc - 2T$, where tc is the current time) is removed from the sender list.

Transmitting RTCP reports

- If *weSent* is true, that is, this entity has recently sent RTP packets, a compound packet consisting of an SR and an SDES is transmitted.

- Otherwise, if this entity is not a sender ($weSent$ is false), a compound packet containing an RR and an SDES is transmitted.
- When a report has been sent, $initial$ is set to false and the value of $avgRtcpSize$ is updated by the following calculation where $packetSize$ is the size of the packet just transmitted: $avgRtcpSize = (1/16) * packetSize + (15/16) * avgRtcpSize$.

Chapter 5

Simulations

The purpose of this chapter is to define the settings and assumptions for all of the performed simulations. Before specific simulation settings are given in the second and third and sections of the chapter, it is specified how user satisfaction was calculated in the simulations. Section 5.2 outlines the settings that are common to all simulations while Section 5.3 presents the unique aspects of the simulation setups in one subsection for each simulation.

5.1 User satisfaction

The concept of user satisfaction is very important when interpreting system simulations. There are many parameters to consider, and even when just looking at a single service, it may be very difficult to determine a user's satisfaction when only objective results are available. A number of studies have been performed with the target to associate objective parameters with subjective evaluations. In the area of audio quality quite extensive models have been obtained that relate a packet loss rate with perceived quality [22]. With respect to these models, it has in the simulations of this study been assumed that an end user is satisfied with the voice conversation if no more than 1% of all voice frames are lost. For terminal-to-terminal communications with two wireless links, the normal requirement is less than 2% frame loss, that is, 1% loss per wireless link. Since the communication architecture of this study contains only one wireless link, the requirement was set to 1%. Concerning video quality, there are not as reliable models available. This is partly due to the higher number of parameters that affect the perceiver, which in turn results in much more complex models. In this study, a simple method with high resemblance to the one for audio has been settled upon. If more than 5% of all video frames are lost in a video session, the user is said to be unsatisfied with the visual experience.

Even more complicated is the issue of the overall experience when receiving both audio and video. Studies have indicated that they interact and affect how the other media is regarded [25]. However, no unified model exists on how they affect each other and the overall experience. Seemingly, the content of the video, as well as other factors affect how they interact and what model should be applied. Because of this, the following simple assumption has been made for result analysis of this study: A user is satisfied with the overall media experience if and only if he/she is satisfied with both the audio and the video separately.

5.2 General simulation settings

Each simulation iteration runs for 120 s. Since each MMTel user follows the same set of events, simultaneous creation of all users would result in a synchronization of all users' actions, i.e. all users would transmit the same type of traffic at the same time. Since it is desirable to have the users evenly spread out into the different stages of the scenario at a given moment in time, an approach was chosen where users are generated randomly with a mean value for user generation per second. The mean number of users per cell in a simulation will thereby be given by Equation 5.1 where t_{life} is the mean lifetime of a user, n_{cells} is the number of simulated cells and r_{gen} is the mean user generation rate, in users per second.

$$n_{users} = t_{life} \frac{r_{gen}}{n_{cells}} \tag{5.1}$$

Since the user life time is variable, depending on setup and termination delays, it is hard to predict the exact number of users per simulation. However, a good estimation can be done by assuming setup delay values. The life time without considering setup delays is 25 s [1] and assuming 5 s for setup and termination delays a mean life time of 30 s is obtained.

In order to ensure stabilization of the user count in the simulation, logging of simulation statistics was not started until after 35 s into the simulation. To reach the number of users given by Equation 5.1, at least one whole life time must elapse from the beginning of the simulation. Only users with their whole lifespan within this logging period were considered in satisfaction calculations.

5.3 Simulation setups

The following section describes the specific settings of the simulations performed in this study. Since it is the effect of QoS dependent scheduling strategies that is the object of study in this thesis it is mainly scheduling settings that are varied in the different simulations and therefore described in the subsections below.

Apart from the first reference simulation, the simulations can be categorized into three sets based on what they are aimed to investigate. Simulation 1a and 1b forms the first pair that studies the effect of setup signaling prioritization. The second category focuses on how additional presence traffic influences system capacity and if the effects can be altered by applying different prioritization schemes. This group consists of Simulation 2a and 2b. The last group examines how media flows can be prioritized in relation to each other and is made up of Simulations 3a, 3b and 3c.

5.3.1 Reference simulation

The reference simulation was performed with the user generation rate set to 3 users/s. By the application of Equation 5.1 and an estimated mean life time of 30 s, this would result in approximately 10 users per cell. No differentiation between the different traffic types was made in the scheduling, meaning that no traffic was prioritized over another. A simple round-robin scheduling was applied where the entity with the longest waiting time is scheduled first.

The round-robin scheduling that acts without traffic type distinction is achieved by associating all traffic flows with only one priority class that utilizes only the delay weight factor. By applying the linear equation of the delay weight factor from time 0 up to 2 s, the queue that was scheduled least recently will be given the highest weight and thus scheduled next.

[1] VoIP session length: 15 s, Video session length: 10 s

The re-scheduling weight of the only priority class was set to a higher value than the maximum achievable delay weight, resulting in a greedy algorithm that lets the queue that is scheduled first in a TTI transmit on all available frequency bands if needed.

10 iterations were performed with varying randomization seeds. Statistics were collected throughout all simulations and the results were in the post-processing averaged for all 10 iterations.

5.3.2 Simulation 1a

Following the reference simulations, a capacity study was performed with the same scheduling algorithm as in the reference simulation. Here, simulations were performed with increasing user load and the decrease of capacity factors with increasing load was analyzed. The user generation rate values for every iteration in the suite can be seen in Table 5.1 where n is the iteration number and r_n is the users generated in mean per second for iteration n.

n	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
$\mathbf{r_n}$	.25	.5	1	2	3	4	5	6	7	8	9	10	11	13	16	20

Table 5.1: User generation settings for each iteration

The complete iteration series was run four times, allowing for average calculations with four results for each user generation setting.

5.3.3 Simulation 1b

After the capacity simulations without different prioritizations for traffic types, a simulation was performed where SIP messages were always prioritized higher than all other traffic types. If there exist a SIP packet in queue to be sent, that packet is always transmitted before any other type of packet. The scheduling scheme within the same priority group was set to round-robin. This results in a scheduling algorithm that can be visualized as in Figure 5.1.

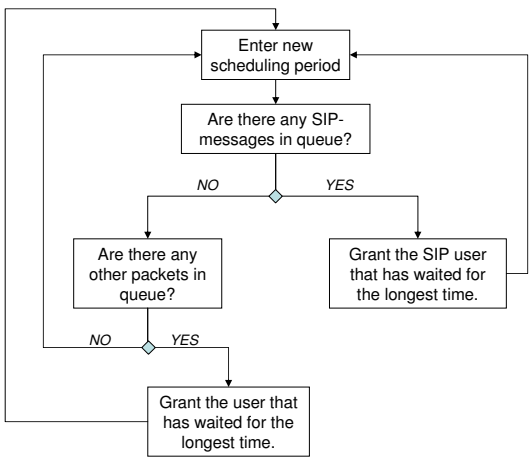


Figure 5.1: The scheduling mechanism for prioritized SIP.

The described scheduling scheme was achieved by splitting the SIP traffic and all other traffic into two different priority classes and giving a large static bonus

weight to the SIP associated priority class. The static extra weight was set to be of such magnitude that it could never be exceeded by the weight factors of the other priority class. In addition to this static weight, both priority classes apply the delay weight factor settings as in the previous simulation, resulting in class internal round-robin scheduling. Also as before, the re-scheduling weight was set so that the algorithm was greedy.

Simulation iterations with the same user generation settings (see Table 5.1) as in the above capacity simulations were run. To get better average results, four simulation runs were performed for the complete iteration series, as in the previous simulation. Statistics were logged and analyzed as before.

5.3.4 Simulation 2a

The scheduling settings of this simulation are exactly the same as in Simulation 1b. What differs the simulation is that here, additional presence users, as described in Section 4.1.2, are existent in the simulated system. The SIP messages that are transmitted to and from these presence users are treated in the same fashion as all other SIP messages. That is, setup messages and presence related messages are both given the absolute prioritization described in subsection 5.3.3 and depicted in Figure 5.1.

The number of presence users in the system is, in contrast to the number of MMTel users, constant throughout the simulation and was set to be 1800 for the complete system, or if divided per cell, 200 presence users per cell. However, even though the number of presence users in the network is static, the number of active presence users is not. In mean, only half of them will be online and actively transmitting and receiving SIP messages, while the other half is offline. The online and offline periods are exponentially distributed and with equal mean lengths. These values, as well as other parameter settings defining the behavior of the presence users, are presented in Table 4.3 in Section 4.1.2.

5.3.5 Simulation 2b

Analogous to Simulation 2a, this simulation includes presence users that generate a ”presence noise”. The number of presence users is, just as in the previous simulation, static and set to 200 users per cell. All other parameters concerning the clients that exploit the presence service are also equal to those in Simulation 2a.

What differs between this and the previous simulation is that here, SIP messages originating from setup procedures are differentiated from presence related SIP messages. The setup messages are still given absolute prioritization, as in Simulation 1b, but the presence packets are assigned to the priority class that the media flows belong to, meaning that they will conform to the basic round-robin scheduling without any additional priority.

5.3.6 Simulation 3a

With the objective to investigate the possibility to favor voice conversations and thus increase the audio capacity, a capacity simulation was performed with a scheduling algorithm that prioritizes VoIP traffic relatively to video traffic. As a base, the same round-robin scheduling where SIP traffic is absolutely prioritized as used in Simulations 1b, 2a and 2b (see Figure 5.1) was used. In Simulation 3a however, the media streams were split into to different priority classes, resulting in three priority classes that can be treated with diverging algorithms. Worth mentioning is that the RTCP reports fall into the priority class that the media stream it controls belongs to. The class that relates to the video traffic utilizes the exact

same algorithm that in previous simulations was employed for all media streams, while the VoIP related priority class awards special weight bonuses to packets based on their age. As explained in Section 3.5, the scheduler of the simulator contains functionality for assignment of packet age dependent bonus weights. By applying this weight factor to the priority class associated with VoIP traffic and setting the time-limits (see Figure 3.6) tight together and close to the maximum tolerated delay for VoIP frames (150 ms), a scheme is obtained where audio packets are deemed as most important to transmit if their delay approaches the acceptable limit.

Traffic type	**Static bonus**	**Delay bonus**				**Age bonus**				**Re-scheduling weight**
		x_1	x_2	y_1	y_2	x_1	x_2	y_1	y_2	
SIP	3000	0	2	1	900	-	-	-	-	2000
VoIP	0	0	2	1	900	0.070	0.110	1	1000	2000
Video	0	0	2	1	900	-	-	-	-	2000

Table 5.2: Scheduling settings aimed to prioritize SIP and VoIP traffic.

In Table 5.2 the scheduler settings that result in the scheduling scheme described above are shown. Since the y_2-value of the age weight factor for VoIP traffic is higher than the largest possible delay bonus, urgent VoIP packets are always scheduled before all of the video frames and the VoIP frames with only delay weight. However, as long as the audio frames are not delayed more than 70 ms, they are treated equally to video frames.

In Simulation 3a, the iteration series was extended in accordance with Table 5.3 to reach higher user generation rates. As for all previous capacity simulations, the series was run four times and the results averaged for all of them.

n	17	18	19	20	21	22
$\mathbf{r_n}$	25	30	35	40	45	50

Table 5.3: Extension of user generation settings in Simulation 3a

5.3.7 Simulation 3b

As a contrast to the previous simulation where VoIP traffic was prioritized higher than the video streams, a simulation was set up where the video traffic is relatively prioritized in a fashion very much like the VoIP traffic was in the simulation described in Section 5.3.6. Table 5.4 presents the scheduler settings that aim to give a relative preference in the scheduler to urgent video frames in comparison with voice frames.

Traffic type	**Static bonus**	**Delay bonus**				**Age bonus**				**Re-scheduling weight**
		x_1	x_2	y_1	y_2	x_1	x_2	y_1	y_2	
SIP	3000	0	2	1	900	-	-	-	-	2000
VoIP	0	0	2	1	900	-	-	-	-	2000
Video	0	0	2	1	900	0.120	0.160	1	1000	2000

Table 5.4: Scheduling settings aimed to prioritize SIP and video traffic.

If Table 5.2 and 5.4 are compared, one can see that the relation between VoIP and video traffic is very much reversed in this latter simulation, only with altered values assigned to y_1 and y_2. Here, the values are chosen to lie close to the maximum accepted transport delay for video (the encoding and decoding delays which are added upon reception are also considered), and are thus higher than for the VoIP traffic in the previous case.

5.3.8 Simulation 3c

In Simulation 3c, the scheduling strategies of Simulation 3a and 3b are combined in an effort to reach high media quality for both audio and video. As with the previous media prioritization schemes, the setup traffic is given absolute priority and the other flows are prioritized relative to each other through the employment of delay weight and age weight bonuses. The parameter settings of the scheduler is shown in Table 5.5.

Traffic type	**Static bonus**	**Delay bonus**				**Age bonus**				**Re-scheduling weight**
		x_1	x_2	y_1	y_2	x_1	x_2	y_1	y_2	
SIP	3000	0	2	1	900	-	-	-	-	2000
VoIP	0	0	2	1	900	0.070	0.110	1	1000	2000
Video	0	0	2	1	900	0.120	0.160	1	977.5	2000

Table 5.5: Scheduling settings in Simulation 3c.

The weight calculation algorithm for VoIP traffic is identical to the one applied in Simulation 3a and that for video traffic resembles the one in Simulation 3b, only with the exception for the slightly lower setting of y_2.

In Chapter 6, the results from all simulations are presented.

Chapter 6

Results

In the sections of this chapter, simulation results will be presented and compared in groups. With the exception of Section 6.1, each of the following sections aims to investigate the effects of a certain type of scheduling and thus present results from simulations that implement such scheduling algorithms. Section 6.2 presents the measured effects of setup signaling prioritization, Section 6.3 reports on the influence of presence traffic and its prioritization on the mixed traffic users while Section 6.4 presents the results from the scheduling algorithms designed to increase media capacities.

6.1 Reference simulation

As the statistics from the 10 iterations of the reference simulation were averaged and analyzed, it was found that the average number of users per cell matched well with the approximation described in Section 5.3.1. The approximation assumed a mean life time of 30 s, which resulted in the assumption of 10 users per cell. The simulation results, displayed in Table 6.1, indicate a slightly shorter mean life time than in the approximation, but the measured number of users per cell does however match the assumed value of 10 users/cell. All the values in the table with simulation results are means.

Parameter	**Value (unit)**
Users per cell	10.0
Total satisfaction	99.8 (%)
VoIP satisfaction	100.0 (%)
Video satisfaction	99.8 (%)
VoIP setup time	3.87 (s)
Video setup time	0.312 (s)
Termination time	0.045 (s)
Life time	29.2 (s)

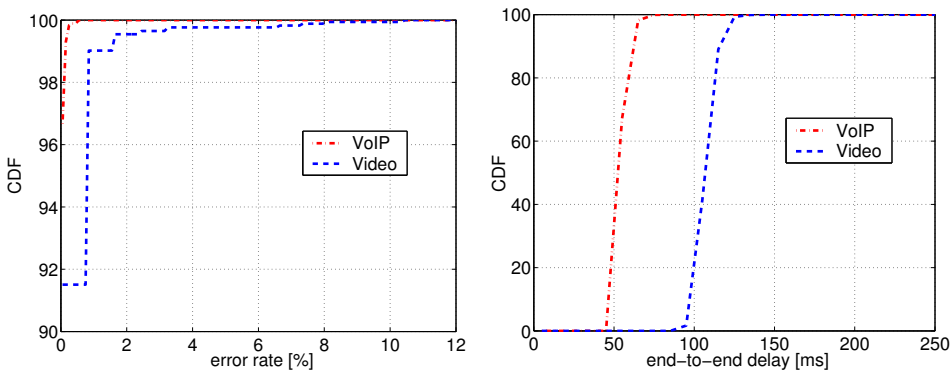
Scenario stage	**Rate of users**
VoIP setup	13.16 (%)
VoIP conversation	51.40 (%)
Video session setup	1.06 (%)
VoIP and video	34.22 (%)
Terminating session	0.15 (%)

Table 6.1: Results from reference simulation

Apart from satisfaction statistics and setup times, the rate of users at a specific stage in the user scenario, depicted in Figure 4.1, was also measured. This measurement was performed in order to see if the ratios were to be changed in later simulations. The result from this measurement can be found in the right part of Table 6.1.

Also measured were the end-to-end delays for VoIP and video frames and the frame error rate for each user and traffic type. The distribution of measured delays

for media packets is shown in Figure 6.1(b) and in Figure 6.1(a) the Cumulative Distribution Function (CDF) for the users' frame error rate is displayed.



(a) Error rates (b) End-to-end delays

Figure 6.1: Error rates and end-to-end delay in the reference simulation.

In Figure 6.1(a) it should be noted that the y-axis range starts at 90 and that the video graph enters at a y-value slightly above 91 and the VoIP graph at a value just below 97. This means that more than 90 % of all video transmissions have no errors at all, and more than 96 % of all VoIP transmissions were completely error-less. This results in very low mean error rates for both VoIP and video. The mean VoIP error rate was 0.006 % and the mean error rate for video was 0.09 %.

The mean delay for voice frames was 61 ms and the corresponding value for video frames was 112 ms. As seen in Figure 6.1(b) the video delays had a slightly wider distribution. By breaking out the term for transport delay[1] in Equation 4.2 for VoIP and Equation 4.3 for video, the mean transport delays (d_{trans}) are found to be 32 ms for video frames and only 11 ms for voice frames. When comparing these two delays, the fact that the video frames are split into sub frames and that this value pertains for the concatenated frame, should be taken into account.

As displayed in the delay graph, all packets arrive within the accepted delay interval (150 ms for VoIP and 250 ms for video). This means that all errors shown in the error graph are due to packet losses rather than packet delay.

6.2 Simulation 1a and 1b

Figure 6.2 shows two graphs that display how the overall satisfaction changes with increasing generation rate in Simulation 1a and Simulation 1b. As seen, the two graphs are very similar in their form, indicating that no deterioration of the media quality is introduced even if SIP signaling is prioritized over VoIP and video. In fact, if capacity is defined as the maximum user generation rate at which 95% or more of all users are satisfied with both audio and video, the capacity is slightly higher if SIP signaling is prioritized. In Table 6.2, all capacity results from Simulation 1a and Simulation 1b are presented separately along with the measured number of active users per cell that they correspond to. The audio and video capacities are calculated applying the same criteria as for the overall capacity, but here only for the voice and the video part of the conversation, respectively.

An interesting result from Simulation 1a without prioritization of SIP signaling was that the user generation intensity did not linearly match the measured users

[1]The transport delay excludes encoding and decoding delays.

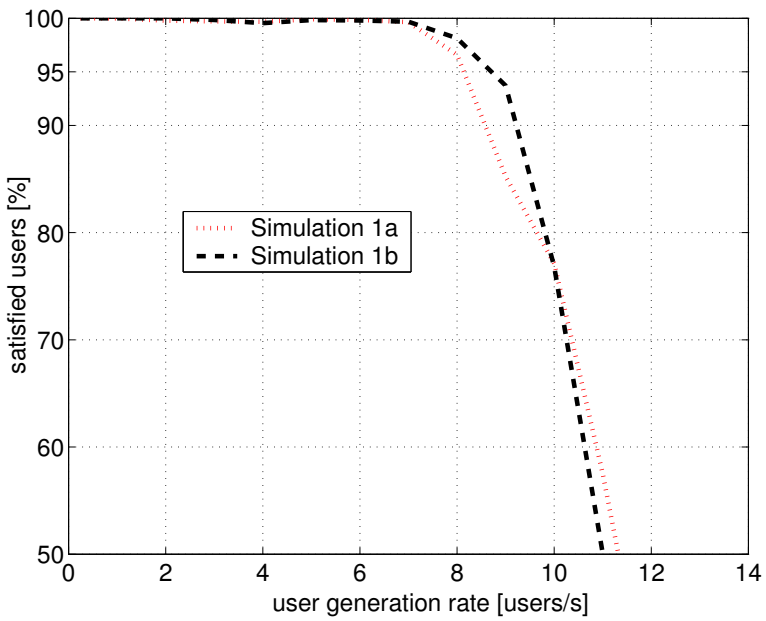


Figure 6.2: The overall user satisfaction rates in Simulation 1a and 1b.

Simulation 1a - *No prio*		
	Generation rate	**Active users**
Overall capacity	8.1 (users/s)	26.8 (users/cell)
VoIP capacity	8.2 (users/s)	27.0 (users/cell)
Video capacity	8.3 (users/s)	27.2 (users/cell)

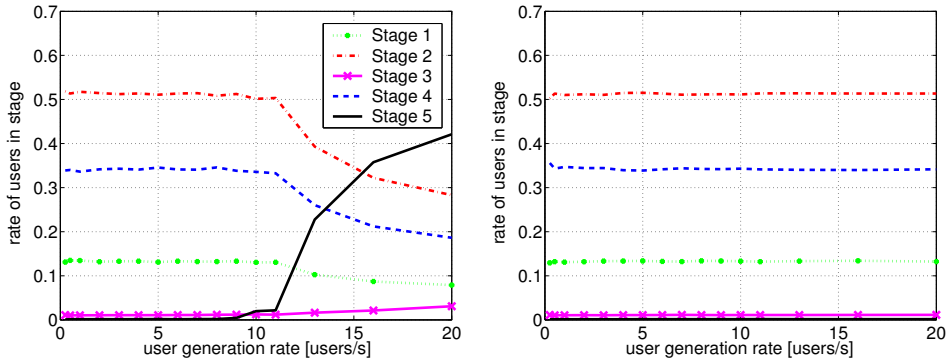
Simulation 1b - *SIP prio*		
	Generation rate	**Active users**
Overall capacity	8.7 (users/s)	27.9 (users/cell)
VoIP capacity	8.9 (users/s)	28.5 (users/cell)
Video capacity	9.0 (users/s)	28.8 (users/cell)

Table 6.2: Capacity results from Simulation 1a and 1b

per cell at a given time. In accordance with Equation 5.1, the relation between the setting of user generation intensity and the number of users per cell should be linear if all of the other variables remain constant. By studying the measured average mean life time of the users, it is found that this variable has not, however, remained constant, but has instead risen from 29 s to above 51 s as the generation rate reaches the highest parameter setting of 20 users/s. This can be seen in Figure 6.4(b). The result of the extended life time is a higher number of active users per cell in average. At the maximum generation rate of this simulation, there is as much as 117 active users/cell, which can be seen in Figure 6.4(a). The prolonged life times are due to extended setup times, in turn caused by longer SIP transmission delays.

In Figure 6.3(a), which pertains to Simulation 1a, it is obvious that the relative number of users at different stages of the MMTel session (see Figure 4.1) is heavily deformed as the load becomes high. More and more users end up in the termination process, which initially is very short but increases dramatically (from 41 ms to 20 s in mean) and becomes a major part of the MMTel session as the generation rate becomes high.

Even though all of the setup delays increase significantly with higher user gener-



(a) Simulation 1a - *No prio*

(b) Simulation 1b - *SIP prio*

Figure 6.3: Rate of users in scenario stages.

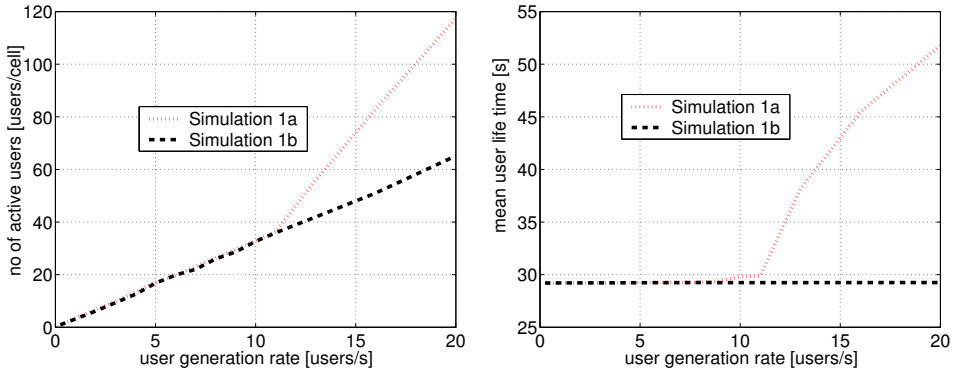
ation rate in the simulation without prioritizations, the most drastic change is seen in the termination delay. The increase of the video setup delay is much lower (from 300 ms to 1.5 s in average) and the length of the VoIP setup is changed the least (from 3.9 s to 4.1 s in mean). This difference is probably due to the fact that for the VoIP setup only SIP traffic is transmitted from the single user, while during the video setup voice packets are simultaneously transmitted and in the termination process, video frames are being sent in conjunction with SIP termination messages (the parts stop talking, thus ending transmission of voice frames, but the video streaming continues until the MMTel session is ended). If other traffic with high bit rate is transmitted from the same user during a SIP setup, the messages will be much more severely delayed as the round-robin scheduler grants the users one transmission at a turn, independently of which type of packets the client wants to transmit. In the termination, the SIP messages have to "compete" with the heavy bit-rate video transmission and is therefore heavily delayed.

When the results from Simulation 1b are investigated, it is evident that the drastic escalation of setup and termination lengths that was found in Simulation 1a is here not an issue. Neither can the non-linear relation between user generation intensity and the number of active users per cell be seen in the results of Simulation 1b. By prioritizing SIP, the drastic degradation of the SIP signaling has been avoided with almost static setup lengths even when the load is increased. When the length of the setups and termination maintain virtually the same length throughout the simulation, so does the overall user life time and a linear relation between user generation intensity and users per cell is obtained. This is visualized in the graphs of Figure 6.4(a). This figure shows how the number of active users changes with increasing generation rate for both of the simulations.

The setup times in Simulation 1b do in fact increase slightly as the load is increased, but the change is very small compared to the previous scenario where no prioritization was applied. When increasing the user generation intensity from 0.25 to 20 users/s the mean termination delay increases with approximately 8 ms, the setup delay for VoIP with 13 ms and the length of the video setup increased with 21 ms. This is to be compared with the drastically higher values for Simulation 1a.

6.3 Simulation 2a and 2b

With the objective to map the behavior patterns and activities of the presence users of Simulation 2a and 2b, measurements on the number of messages they send



(a) Number of active users

(b) User life time

Figure 6.4: Users per cell and their life time with increased user generation rate.

as well the time that pass between their different activities were performed. In Table 6.3, numerical results from these measurements are presented. All values presented there are means and were calculated over the entire iteration series. The publish interval and the notification interval refer to the time between publish events and notification events, respectively. Since these events may only occur when the user is online, the intervals where the opening event occurs before an offline period and the subsequent event after the period were not considered in the measurements.

	Simulation 2a	**Simulation 2b**
Message rate per user	37.5 (mess/h)	37.5 (mess/h)
Rate of users online	50.15 %	49.90 %
Rate of users offline	49.85 %	50.10 %
Online duration	21 095 (s)	21 540 (s)
Offline duration	21 084 (s)	22 658 (s)
Publish interval	1184 (s)	1161 (s)
Notification interval	80 (s)	80 (s)

Table 6.3: Mean measurements for the presence users

The measurements concerning the presence users' activities were, as one might expect due to the identical settings in this area, very similar in the two simulations with presence users. The measured results in Table 6.3 may be compared to the presence user settings in Table 4.3.

In Figure 6.5, three graphs displaying how the overall satisfaction rates changes as the generation rate increases in the Simulations 2a, 2b and 1a, are shown. The satisfaction rate graph of Simulation 1a was included as a reference since it holds the same scheduling settings as Simulation 2a and 2b (absolute prioritization of SIP messages), only without the existence of presence users and messages.

As a complement to the satisfaction graphs, Table 6.4 presents the capacity results from both of the simulations with presence users. Capacity measurements for both of the media streams separately and in combination as well as the measured number of users per cell that they correspond to are shown there.

As can be seen, both in the satisfaction graphs and the numerical capacity results, the difference in the results from the simulations is very small, both if comparing Simulation 2a and 2b and also if they are set against the capacity and satisfaction results of Simulation 1b. If the error rates for the media streams are investigated, a high resemblance between the simulation results is found here as well. Figure 6.6 shows these error rates and how they change with increasing load.

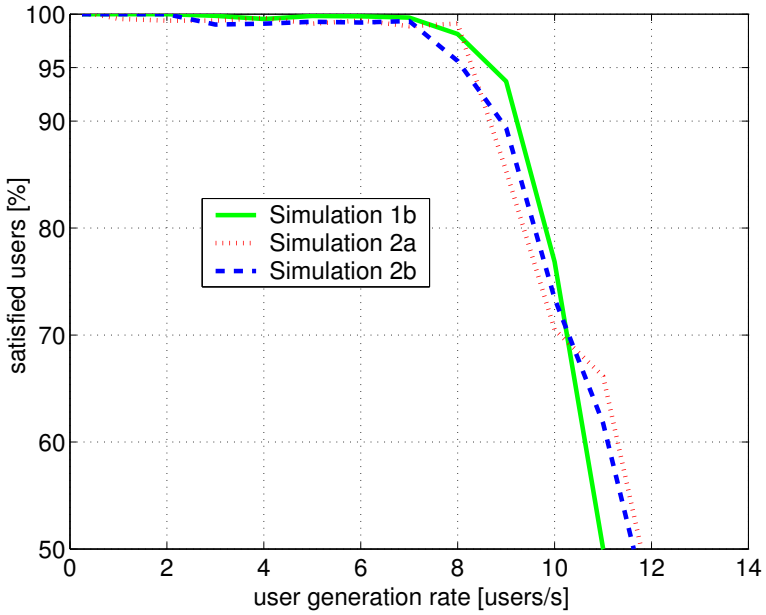
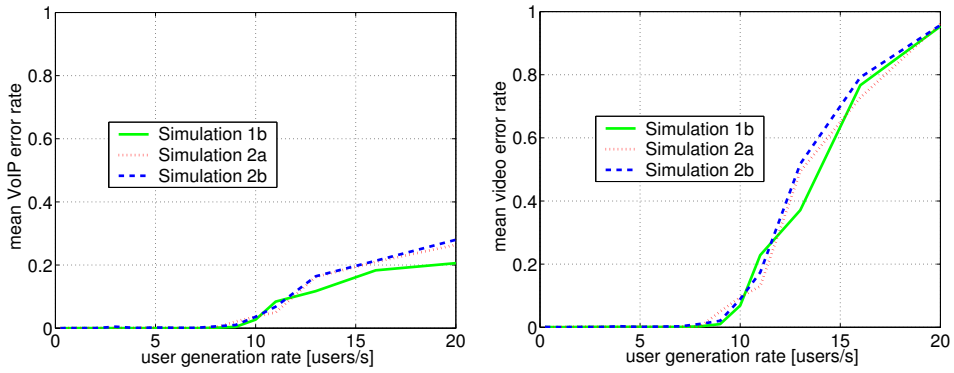


Figure 6.5: The overall user satisfaction rates in Simulations 2a and 2b with Simulation 1b included as a reference.

Simulation 2a - *High prio presence*		
	Generation rate	**Active users**
Overall capacity	8.3 (users/s)	27.6(users/cell)
VoIP capacity	8.3 (users/s)	27.7 (users/cell)
Video capacity	8.4 (users/s)	27.9 (users/cell)

Simulation 2b - *Low prio presence*		
	Generation rate	**Active users**
Overall capacity	8.1 (users/s)	26.6 (users/cell)
VoIP capacity	8.2 (users/s)	26.9 (users/cell)
Video capacity	8.4 (users/s)	27.7 (users/cell)

Table 6.4: Capacity results from Simulation 2a and 2b



(a) VoIP error rates

(b) Video error rates

Figure 6.6: Mean error rates for Simulations 2a, 2b and 1b.

When the capacity and the error rates are summarized for Simulation 2a and 2b and compared to each other and Simulation 1b, it is found that a slight overall decrease of media quality is imposed as presence users are introduced into the system. It is, however, hard to see any difference in media quality due to the prioritization of presence messages. The results seem to suggest that the capacity decreases if presence users are introduced into the system (if Simulation 1a values are compared to those of Simulation 2a and 2b), but the difference is very small. If instead the effects on setup signaling are analyzed, the opposite relation between Simulation 2a and 2b is found. The results of Table 6.5 indicate that the setup processes are less impaired if the presence messages are not as highly prioritized as the setup messages. The difference is, however, as before, minor.

	Simulation 1b	**Simulation 2a**	**Simulation 2b**
Mean VoIP setup delay	3.872 (s)	3.888 (s)	3.877 (s)
Mean video setup delay	317 (ms)	348 (ms)	344 (ms)
Mean termination delay	45.4 (ms)	57.8 (ms)	55.0 (ms)
Mean user life time	29.23 (s)	29.28 (s)	29.27 (s)

Table 6.5: Setup measurements in Simulations 2a (with high prio presence) and 2b (with low prio presence) with Simulation 1b (no presence) as a reference

The measured values presented in Table 6.5 further illustrates that the setup processes are prolonged if presence users exist in the system (if Simulation 1b results are compared to those of Simulation 2a and 2b).

6.4 Simulations 3a, 3b and 3c

In Table 6.6, the measured capacity results from Simulations 3a, 3b and 3c are presented. The values for Simulation 3a indicate that the prioritization applied in this simulation achieved a high audio quality even at high user loads. The capacity for VoIP in this simulation is far higher than any other capacity result in any of the other simulations of this study. The video and overall capacities are in contrast not very far below the results from Simulation 1b.

The capacity measurements from Simulation 3b are low if compared to measurements on other simulations of this study. Not even the video capacity is very high, despite the fact that the video frames were prioritized in this simulation.

In Simulation 3c, the overall capacity is the highest in the study. As seen in the capacity table, both of the media streams hold relatively high capacities separately and the video capacity in Simulation 3c is the highest measured in all of the simulations. The VoIP capacity is the second highest, only lower than that of Simulation 3a.

Figure 6.7 visualizes the substantially higher VoIP capacity in comparison to the video capacity of Simulation 3a. In this figure, it is shown that the satisfaction rate for audio remains at 100 % far beyond the point where the satisfaction rate for video drops below 95 %. As the overall capacity is defined as the highest user generation rate where more than 95 % of all users are happy with both the audio and the video, the overall satisfaction will completely be determined by the video satisfaction rate. Thus, the overall capacity does not increase as VoIP is prioritized (compare overall capacity results for Simulation 1b and 3a). Important to notice, however, is the fact that it does not significantly decrease either, meaning that the video quality is not heavily impaired by the down-prioritization. This is further supported by the graphs of Figure 6.10 and 6.11, which show the frame delays and error rates of all the simulations with media stream prioritizations as well as corresponding graphs for Simulation 1b as a reference. If the graphs for Simulation 1b are compared to

Simulation 3a - *VoIP prio*		
	Generation rate	**Active users**
Overall capacity	8.4 (users/s)	27.3 (users/cell)
VoIP capacity	45.9 (users/s)	151.0 (users/cell)
Video capacity	8.4 (users/s)	27.3 (users/cell)

Simulation 3b - *Video prio*		
	Generation rate	**Active users**
Overall capacity	6.9 (users/s)	22.9 (users/cell)
VoIP capacity	7.0 (users/s)	23.6 (users/cell)
Video capacity	8.1 (users/s)	26.0 (users/cell)

Simulation 3c - *Media delay based prio*		
	Generation rate	**Active users**
Overall capacity	9.4 (users/s)	30.2 (users/cell)
VoIP capacity	11.2 (users/s)	35.7 (users/cell)
Video capacity	9.4 (users/s)	30.3 (users/cell)

Table 6.6: Capacity results from Simulations 3a, 3b and 3c

those for Simulation 3a, it can be seen that the frame delays and error rates of the video stream has only increased slightly in the latter simulation. Noticeable in Figure 6.10(a) is that none of the scheduling schemes succeeds in achieving significantly lower error rates for the video stream at any user generation rate. Not even Simulation 3b, in which the scheduling algorithm aims to prioritize video frames based on their age, achieves any lower error rates. Figure 6.8 shows how the failure to produce low video error rates in Simulation 3b results in low satisfaction rates for the video stream. The low satisfaction rate results in turn in the low capacity results discussed above and displayed in Table 6.6.

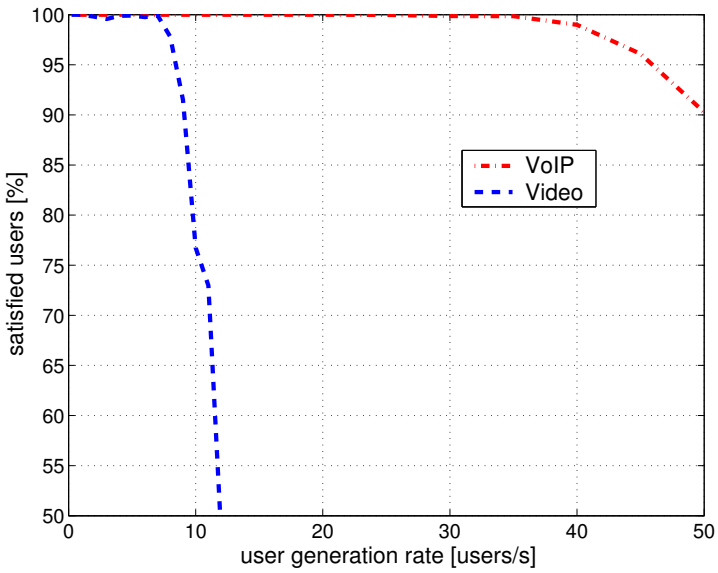


Figure 6.7: Satisfaction rates for the media streams in Simulation 3a.

Even though the error rates for the video were not increased in Simulation 3b (if compared to Simulation 1b without video frame prioritization) it is obvious when studying the graphs of Figure 6.10(b) that the mean delay for video frames has been significantly decreased for user generation rates above 9 users/s. The mean delay values represented in the graphs of Figure 6.10 were measured from the delays

of those frames received within the simulation. This means that if a frame is not received at all, it will not contribute to the mean delay measurement but will affect the mean error rate factor. Figure 6.12 presents the rate of errors that are caused by frames that are received, but beyond the acceptable delay limit. As seen, most of the graphs reach a maximum y-value at the middle of the x-range. Their decrease in the right part of the graphs is probably due to the fact that beyond a certain load more and more packets will not reach the receiver at all within the life time of the user and will therefore not be considered as a delay due error. Most interesting in Figure 6.12(b) is the graph for Simulation 3b. This graph holds no steep slope even at user generation rates where the increase of the error rate (see Figure 6.11) is just as rapid as for the other simulations. This means that in Simulation 3b, most errors in the video stream are due to completely missing frames as opposed to frames that are received, but too late. It seems as when the system strain becomes high; the scheduling scheme employed in Simulation 3b succeeds in delivering some frames with short delays, but on the other hand fails to deliver many packets.

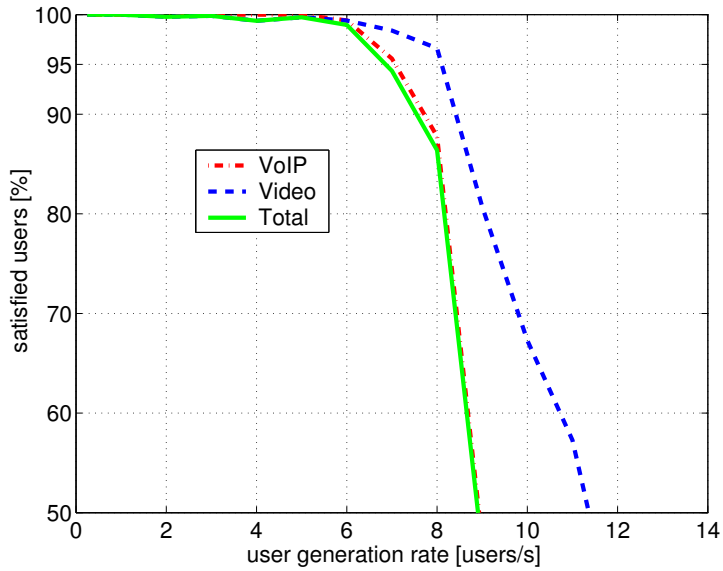


Figure 6.8: Satisfaction rates in Simulation 3b.

Another effect caused by the prioritization of the video in Simulation 3b is that the SIP messages experience increasing delays at user generation rates above 10 users/s. The extended result of this is increased setup lengths. However, they are not nearly as severe as in the case of Simulation 1a, where no prioritization of the SIP messages was applied. Figure 6.13 depicts the growth of setup and termination times as the user generation rates increase in Simulations 3a, 3b and 3b. As is evident in this figure, also the scheduling algorithm of Simulation 3c shows signs of causing increasing SIP message delays at higher user generation rates, even though the increase is even smaller than in Simulation 3b. Possibly, this setup degradation can be linked to the video frame age dependent prioritization that exists in both Simulation 3b and Simulation 3c.

In Simulation 3c, the highest overall capacity for all simulations was reached. Figure 6.9 shows how the rate of users satisfied with each of the media streams as well as both of them drops as the user generation rate is increased. As seen, it is the video satisfaction rate that limits the overall satisfaction since these two graphs highly coincides.

In the last, concluding chapter a discussion on the results will be given.

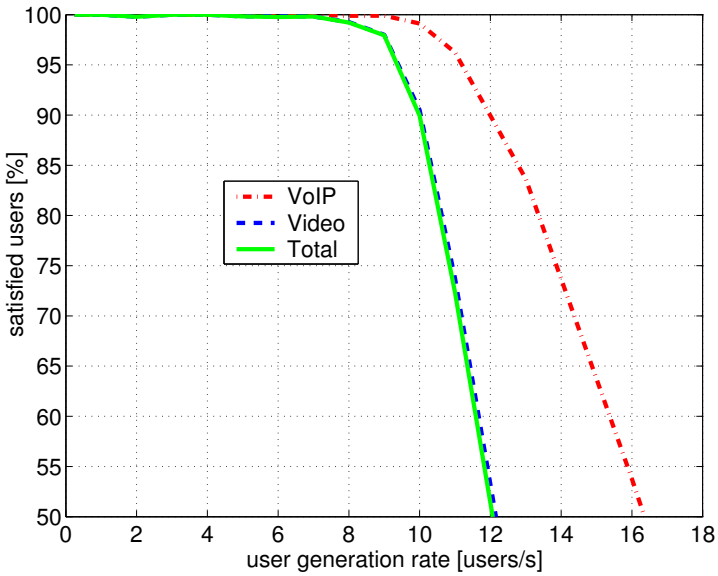
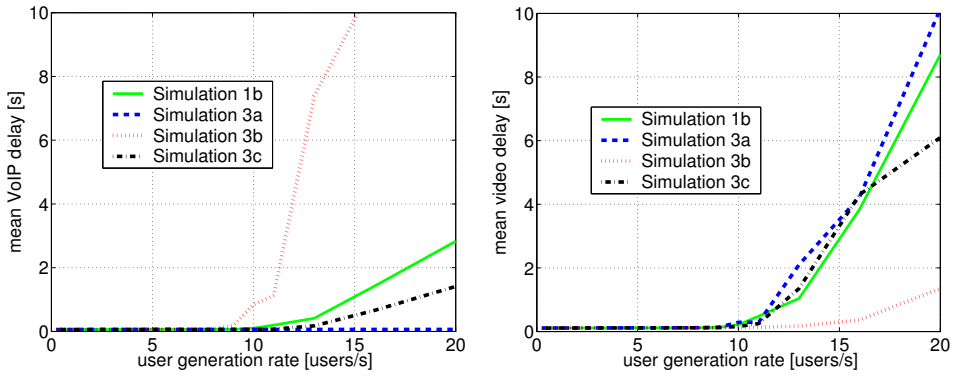
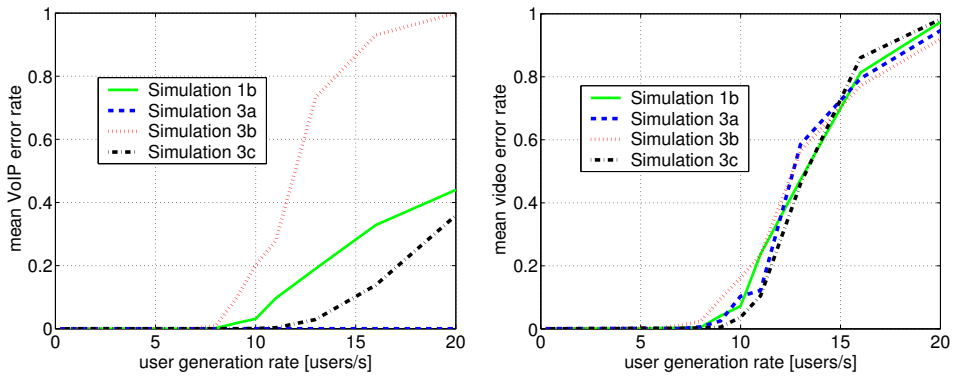


Figure 6.9: Satisfaction rates in Simulation 3c.



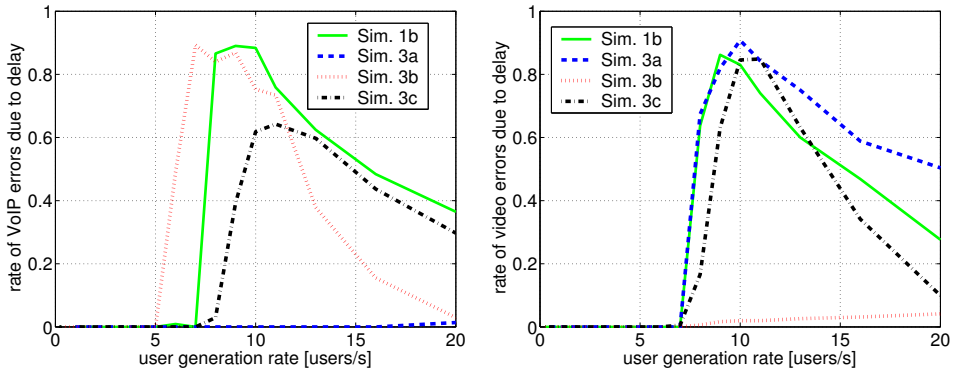
(a) VoIP (b) Video

Figure 6.10: Media delays in Simulations 3a, 3b, 3c and 1b.



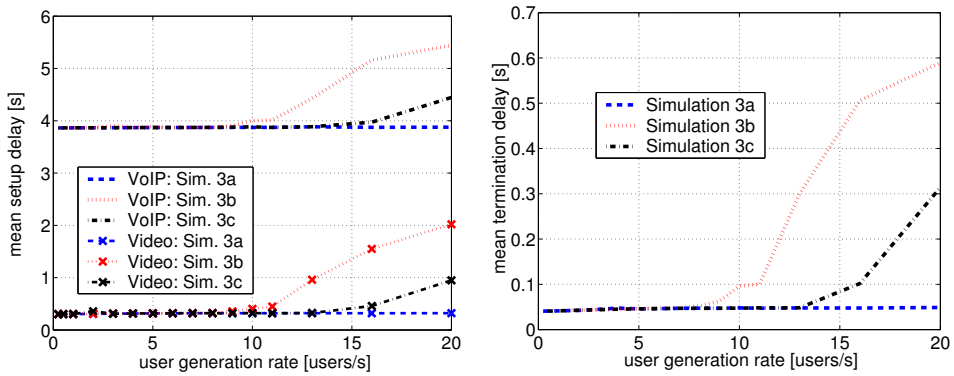
(a) VoIP (b) Video

Figure 6.11: Error rates in Simulations 3a, 3b, 3c and 1b.



(a) VoIP (b) Video

Figure 6.12: Delay due error rates in Simulations 3a, 3b, 3c and 1b.



(a) Setup delays (b) Termination delay

Figure 6.13: Setup and termination delays in Simulation 3a, 3b and 3c.

Chapter 7

Discussion

In the first part of the discussions of this chapter, conclusions are drawn, based on the results of this study. The last and concluding section of this report discusses what studies might best complement the findings presented in this thesis report.

7.1 Conclusions

By comparing results from the simulations with and without SIP prioritization, it can be concluded that the SIP signaling performance in our multi-media scenario is heavily impaired when the load on the system is high if all traffic is scheduled equally with no prioritizations. The result of this is very long setup times and the possible effect that MMTel session setups cannot be completed at all. Even though the heavy degradation of SIP signaling quality occurs at a stage where most clients are to be considered unhappy with the audio and video quality, this effect might be considered severe since SIP signaling is the most vital traffic, being the basic foundation by which other services are established. In the simulations where SIP traffic is prioritized it is shown that if the setup messages are absolutely prioritized this complex of problems is avoided. With this scheduling scheme, media session setups and terminations remain functional far beyond the point where other services yield unsatisfactory quality. Judging by the capacity results, the prioritization of SIP traffic does not seem to harm the media traffic significantly. On the contrary, the capacity measured in the simulation with prioritized setup traffic shows to be slightly higher, possibly because SIP messages need not be retransmitted as frequently, which results in lower amount of total traffic. The higher capacity is, however, very small and may only be regarded as a possibility.

When presence traffic is introduced into the system, a small decrease of media quality and a slight lengthening of the setup and termination delays can be seen. Even though it is hard to find any clear and significant distinction in the measured effects of the different prioritization algorithms for the presence traffic, some indications are visible. Regarding setup processes, it seems to be favorable to not prioritize the presence messages as highly as the setup messages. On the other hand, the distinction in prioritization between setup messages and presence messages seems to generate slightly lower capacity, which is due to higher losses in the media streams at medium system loads. It can be assumed that if the number of presence users had been higher, thereby increasing the traffic load introduced by the presence service, the indications discussed above might have been more significant. However, since the differences in the results of the simulations with presence users are as minor as they are, no certain conclusions regarding the prioritization of presence messages and its effect on other traffic types can be drawn.

The simulation results of this study further suggest that it is possible to prioritize VoIP frames based on their age and thus achieve a highly increased capacity for the voice service. This scheduling scheme does on the expense of other traffic types prioritize VoIP frames, but only when they urgently need to be transmitted, i.e., when their delay approaches the acceptable limit. This approach achieves very favorable results for such a low-bit rate service as VoIP and seems not to introduce any large amount of extra errors to the (more bit-rate demanding) video service. However, when the same approach was applied on the video traffic, no higher video capacity could be achieved. This is probably due to the much higher-bit rate of the video. Thus it is reasonable to conclude that the concept of prioritizing a special type of packets as they approach their acceptable delay limit is only successful if they constitute a relatively small part of the total traffic load. The conclusion that it is possible to prioritize VoIP traffic in the above described fashion provides the operators with the opportunity to guarantee their clients a high VoIP quality even in situations where video transmissions are not possible due to high system load. An operator policy could thus be defined where both voice and video (and other medias) are delivered with high quality as long as the system load is not too high, but where only voice conversations are allowed for new MMTel sessions if the user load exceed a certain level. Admission Control could be used to not allow new video sessions if the load is above the level where reasonable video quality can be assured, thus further ensuring that users at least can set up and engage in VoIP conversations of satisfactory quality.

When the frame age dependent priority algorithm was applied on both of the media streams, the overall capacity was slightly increased, indicating that the combination benefited the quality of both of the media streams in a system with relatively high load. However, as before, the quality of the video streams was the capacity limiting factor, further suggesting that if the video traffic is such a large part of the traffic it is hard to maintain high video quality for a high number of simultaneous users.

To summarize the conclusions, the results from the simulations in this study suggest that prioritization of SIP setup messages is necessary in order to assure delivery of the important setup messages in an all-IP network. In addition to this, it was shown that with the prerequisite that VoIP traffic is a relatively small part of all data traffic in the system, audio quality can be guaranteed for a vastly higher user load if the voice frames are given a priority such that they are always scheduled before other media frames if they approach their acceptable delay limit. Furthermore, the results state that it is possible to design a scheduling algorithm such that the combined satisfaction of both voice and video can be increased. However, the video stream proved in the scenario of this study to be the limiting media due to its significantly greater demand of bit-rate. Regarding presence traffic and the prioritization of it, no certain conclusions can be drawn except that the traffic load induced by the 200 presence users per cell that was applied in this study did not notably harm the other traffic flows, no matter if it was highly prioritized or not.

7.2 Future work

When studying mixed traffic scenarios in an all-IP system, there are a large number of parameters and scenario assumptions that can be varied. In this study, the scope was limited to a very specific traffic scenario. A natural extension to this work would therefore be to complement it with similar scheduling studies but on different traffic scenarios. Slight, but interesting, modifications could include the lowering of the bit-rate for the video and changing the length of the users' life time and the stages that it is comprised of. With a lower video bit-rate, perhaps

the video would not be an as dominating capacity limit. It is also possible that changes in the video encoding and transport settings could result in a higher video capacity. More radical changes in the traffic scenario could consist of the inclusion of other traffic types, originating for example from such services as web-browsing and text-messaging, or the application of less static traffic scenarios for each user. Simulations could also be run where not all clients act by the same scheme of events, even though this would introduce more complexity in the result analysis.

Another intuitive addition to the work presented in this report would be to analyze an extended list of scheduling algorithms. Both small traffic type dependent variations in the already implemented algorithms as well as the application of new ones would be of great interest. Variations could for example investigate the effect of other forms than *absolute* prioritization of SIP signaling. Age dependent priority bonuses, much like that applied on the media streams in some of the simulations, could for example be assigned to the setup messages. Moreover, it would be interesting to implement functionality in the scheduler for discarding of packets already too far delayed. Possibly, this would increase the system capacity.

The section of this study that dealt with users of the presence service, and how presence messages should be prioritized, did not yield any conclusive results. Due to resource limitations, the number of presence clients in the simulations had to be limited to a level where no significant effects on other traffic could be noticed. It would therefore be of high relevance to perform more simulations with a higher number of presence users per cell.

As this thesis was only concerned with the downlink scheduling of LTE, there is still a need for a corresponding study of the uplink.

Furthermore, some of the models of this study could be extended and modified to model some more complex aspects of a real system. One important example is the media quality and user satisfaction models that are very simplified in this study. These models could be extended to take into account such aspects as jitter, synchronization and inter-media influences. The RTCP model could also be extended with the additional functionality to perform actual synchronization between the media streams.

Finally, the results of this study could be further verified by running more iterations of the presented simulations.

Bibliography

[1] U. Olsson, "Toward the All-IP Vision," *Ericsson review*, vol. 1, 2005. [Online]. Available: http://www.ericsson.com/ericsson/corpinfo/publications/review/2005_01/files/2005011.pdf

[2] 3GPP, "IP Multimedia IM Subsystem; Stage 2," 3rd Generation Partnership Project, Tech. Rep. TS 23.228, 2006. [Online]. Available: http://www.3gpp.org

[3] S. Chakraborty, J. Peisa, T. Frankkila, and P. Synnegren, *IMS Multimedia Telephony over Cellular Systems*. John Wiley & Sons, Ltd, 2007. [Online]. Available: http://eu.wiley.com/WileyCDA/WileyTitle/productCd-0470058552.html

[4] M. Ericson, L. Voigt, and S. Wänstedt, "Providing reliable and efficient VoIP over WCDMA," *Ericsson review*, vol. 2, 2005. [Online]. Available: http://www.ericsson.com/ericsson/corpinfo/publications/review/2005_02/06.shtml

[5] R. Cuny and A. Lakaniemi, "VoIP in 3G Networks: An End-to-End Quality of Service Analysis," Nokia Networks and Nokia Research Center, Tech. Rep., 2003. [Online]. Available: http://whitepapers.techrepublic.com.com/whitepaper.aspx?docid=101835

[6] 3GPP, "Requirements for Evolved UTRA (E-UTRA) and Evolved UTRAN (E-UTRAN)," 3rd Generation Partnership Project, Tech. Rep. TR 25.913, 2004. [Online]. Available: http://www.3gpp.org

[7] 3GPP, "System Architecture Evolution (SAE): Report on Technical options and conclusions," 3rd Generation Partnership Project (Rel 7), Tech. Rep. TR 23.882, 2005. [Online]. Available: http://www.3gpp.org/ftp/Specs/html-info/23882.htm

[8] O. Edfors *et al.*, "An introduction to orthogonal frequency-division multiplexing," Luleå University of Technology, Tech. Rep., 1996. [Online]. Available: http://www.sm.luth.se/csee/sp/research/report/esb96rc.pdf

[9] E. Dahlman *et al.*, "The 3G Long-Term Evolution - Radio Interface Concepts and Performance Evaluation," *IEEE*, 2006. [Online]. Available: http://www.ericsson.com/technology/research_papers/wireless_access/doc/the_3g_long_term_evolution_radio_interface.pdf

[10] Rosenberg *et al.*, "RFC3261 Session Initiation Protocol (SIP)," *IETF*, 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3261.txt

[11] J. Rosenberg, "RFC3856 A Presence Event Package for the Session Initiation Protocol SIP," *IETF*, 2004. [Online]. Available: http://www.ietf.org/rfc/rfc3856.txt

[12] H.-P. Rajaniemi and K. Yanev, "SIP and Presence," University of Helsinki, Tech. Rep., 2005. [Online]. Available: http://www.cs.helsinki.fi/u/yanev/simplep.pdf

[13] D. Henriksson, "Analysis and Optimizations of Presence Generated Traffic for Cellular Networks," Master's thesis, Luleå University of Technology (LTU), Luleå, Sweden, January 2005.

[14] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, "RFC3550 RTP: A Transport Protocol for Real-Time Applications," *IETF*, 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3550.txt

[15] R. Ludwig *et al.*, "An Evolved 3GPP QoS Concept," *IEEE*, 2006. [Online]. Available: http://www.ericsson.com/technology/research_papers/wireless_access/papers/evolved_3gpp_qos.shtml

[16] C. Bormann *et al.*, "RFC3095 RObust Header Compression (ROHC): Framework and four profiles: RTP, UDP, ESP, and uncompressed," *IETF*, 2001. [Online]. Available: http://www.ietf.org/rfc/rfc3095.txt

[17] M. Ericson and S. Wänstedt, "Mixed Traffic HSDPA scheduling - Impact on VoIP Capacity," *VTC2007 - Spring*, 2007.

[18] S. Wänstedt, M. Ericson, K. Sandlund, M. Nordberg, and T. Frankkila, "Realization and Performance Evaluation of IMS Multimedia Telephony for HSPA," *PIMRC*, 2006.

[19] E. Soljanin, "Hybrid ARQ in Wireless Networks," 2003. [Online]. Available: http://cm.bell-labs.com/cm/ms/who/emina/talks/ppmcs1.pdf

[20] T. Eyers and H. Schulzrinne, "Predicting internet telephony call setup delay," 2000. [Online]. Available: citeseer.ist.psu.edu/eyers00predicting.html

[21] M. Handley and V. Jacobson, "RFC2327 Session Description Protocol (SDP)," *IETF*, 1998. [Online]. Available: http://www.ietf.org/rfc/rfc2327.txt

[22] 3GPP, "TSG-SA Codec Working Group: Mandatory speech codec; AMR speech codec; Interface to Iu and Uu," 3rd Generation Partnership Project, Tech. Rep. TS 26.102, 1999. [Online]. Available: http://www.3gpp.org

[23] J. Sjoberg *et al.*, "Real-Time Transport Protocol (RTP) Payload Format and File Storage Format for the Adaptive Multi-Rate (AMR) and Adaptive Multi-Rate Wideband (AMR-WB) Audio Codecs," *IETF*, 2002. [Online]. Available: http://www.ietf.org/rfc/rfc3267.txt

[24] ITU-T, "ITU-T rec. H.263 Video coding for low bit-rate communication," ITU, Tech. Rep., 2005. [Online]. Available: http://www.itu.int/rec/T-REC-H.263/en

[25] D. Hands, "A basic multimedia quality model," *IEEE Trans. Multimedia*, vol. 6, pp. 808–816, 2004.