University of Würzburg
Institute of Computer Science
Research Report Series

Measurement and Analysis of Skype VoIP Traffic in 3G UMTS Systems

Tobias Hoßfeld[1], Andreas Binzenhöfer[1], Markus Fiedler[2], Kurt Tutschku[1]

Report No. 377 December 2005

[1] University of Würzburg
Department of Computer Science
Am Hubland, D-97074 Würzburg, Germany
{hossfeld,binzenhoefer,tutschku}@informatik.uni-wuerzburg.de

[2] Blekinge Institute of Technology,
Department of Telecommunication Systems.
SE-371 79 Karlskrona, Sweden.
markus.fiedler@bth.se

Measurement and Analysis of Skype VoIP Traffic in 3G UMTS Systems

Tobias Hoßfeld, Andreas Binzenhöfer, Kurt Tutschku
University of Würzburg
Department of Computer Science
Am Hubland, D-97074 Würzburg, Germany
{hossfeld,binzenhoefer, tutschku}@informatik. uni-wuerzburg.de

Markus Fiedler
Blekinge Institute of Technology,
Department of Telecommunication Systems.
SE-371 79 Karlskrona, Sweden.
markus.fiedler@bth.se

Abstract: *Voice-over-IP (VoIP)* telephony becomes more and more popular in the wired Internet because of easy-to-use applications with high sound quality like *Skype*. UMTS operators promise to offer large data rates which should also make VoIP possible in a mobile environment. However, the success of those application strongly depends on the user perceived voice quality. In this paper, we therefore analyze the achievable and the actual quality of IP-based telephony calls using Skype. This is done performing measurements in both a real UMTS network and a test environment. The latter is used to emulate rate control mechanisms and changing system conditions of UMTS networks. The results show whether Skype over UMTS is able to keep pace with existing mobile telephony systems and how it reacts to different network characteristics. The investigated performance measures comprise the *Perceptual Evaluation of Speech Quality* (PESQ) to evaluate the voice quality, the packet loss, the inter-packet delay, and the throughput to capture network-based factors. In this context, the concept of the Network Utility Function (NUF) is applied to describe the impact of the network on the voice quality as perceived by the end-user.
Keywords: VoIP, Skype, UMTS, measurement, Network Utility Function

1 Introduction

Voice-over-IP (VoIP) telephony becomes more and more popular in the wired Internet because of easy-to-use applications with high sound quality like *Skype* [1, 2]. In conjunction with high speed access technologies, like *Digital Subscriber Line (DSL)*, and flat-rate tariff models, VoIP telephony has grown to a strong competitor to the existing *Public Switched Telephone Network (PSTN)*. UMTS operators, however, are still searching for new applications which both exploit the potential of the UMTS 3G technology and motivate the user to adopt the technology. VoIP applications like the Peer-to-Peer (P2P) based Skype system might be an interesting candidate for such an application.

Due to its P2P technology, Skype might offer rapid access to a large base of users, seamless service operation across different types of networks (wireline and and wireless) and a serverless and cost efficient operation of a new service. As a result, a German UMTS operator has recently announced the official support of Skype in its system [3].

UMTS operators promise to offer large data rates which should make mobile VoIP possible. In the context of mobile communication systems, however, the perceived voice quality has to be evaluated for IP-based telephony. In this paper, the achievable and the actual quality of voice calls using Skype is analyzed by performing measurements in a real UMTS network and additionally in a test environment. The testbed is used to emulate rate control mechanisms and changing radio propagation conditions in UMTS. The results show how Skype reacts to network influences and if Skype over UMTS keeps pace with existing mobile telephony systems. The investigated performance measures are the *Perceptual Evaluation of Speech Quality* (PESQ) for evaluating the voice quality, the packet loss, the inter-packet delay, and the throughput for describing the network-based factors. The concept of the Network Utility Function (NUF) [4] is applied to describe the impact of the network on the perceived voice quality.

The remainder of this work is organized as follows. In Section 2, the setups for the bottleneck LAN measurements and for the UMTS measurements are shown. Section 3 describes the performance objectives and measures. The measurement results and the analysis of the Skype application in the different scenarios are given in Section 4. In particular, Section 4.1 covers the bottleneck LAN which emulates changing wireless conditions by adapting the transmission rate. The measurements in the real UMTS networks are evaluated in Section 4.2. In Section 4.3, we apply the network utility function which uses network-based parameters, like packet loss or delay, as input in order to derive the user perceived quality. Finally, Section 5 concludes this work and gives an outlook for the next steps of the performance evaluation of Skype.

2 Measurement Setup and Scenario

The general measurement setup is the following: Skype user A sends audio data to Skype user B. We used an English spoken text without noise, a sample rate of 8 kHz, encoded with 16 bits per sample which is a standard audio file for evaluating VoIP and available at [5]. The wav-file is played with the Winamp audio player on machine A. The output of Winamp is used as input for Skype (instead of a microphone). On sender A and receiver B, Windows XP is the OS, Skype v1.20.37 is installed and a packet trace is captured with TCPDump on each machine. In order to save the received audio data, the output of Skype has to be forwarded with an audio cable from B to another machine.

2.1 Bottleneck LAN Setup

In a UMTS system, the conditions of the wireless channel are changing over time because of radio propagation effects or (slow and fast) fading, especially for a moving UMTS user. Additionally, the system load affects the available bandwidth of an arbitrary user. On rate-controlled dedicated channels or using HSUPA/HSDPA, this results in the bandwith adaptation of a UMTS subscriber. In the bottleneck LAN scenario, we emulate a UMTS user who starts a Skype VoIP call and has an initially assigned bandwidth of 16 kbps. During the call, the network conditions change and the user is assigned a higher bandwidth. We increased the bandwidth up to 384 kbps which reflects the currently offered downlink bandwidth of the public UMTS operator. In order to evaluate and compare the perceived voice quality for the different assigned bandwidths, the same audio data was transmitted for each measured bandwidth. The bandwidth restriction was realized by using a traffic shaping router, cf. Figure 1.

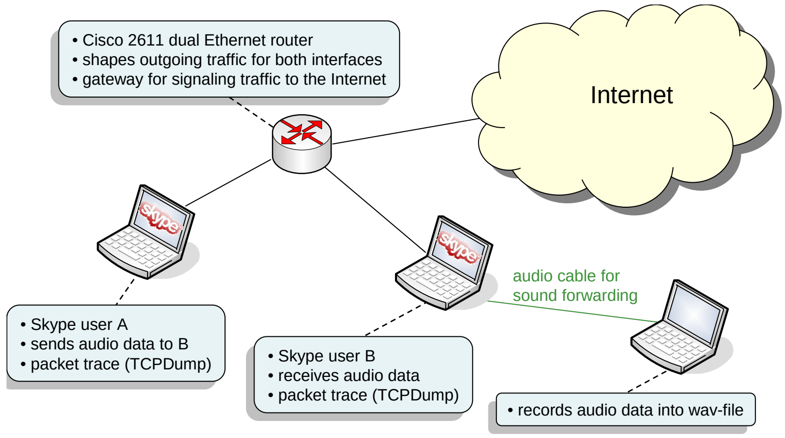


Figure 1: Bottleneck LAN scenario using a traffic shaping router

Another possibility to restrict the bandwidth in the LAN is using the dummynet software [7] by setting an appropriate packet loss value, cf. Figure 2. In the dummynet scenario, Skype detects that packet loss occurs and interpretes this as congestion in the network. Therefore, Skype tries to push away TCP traffic by increasing the throughput of the sender. The throughput increase is realized by transmitting packets with the same frequency, but more payload. Figure 3 shows the througput of the sender and the throughput at the receiver. As soon as Skype detects packet loss it reacts by increasing the sender throughput from 24 kbps to roughly 39 kbps.

The bottleneck LAN scenario using a traffic shaping router and using the dummynet software both lead to the same received throughput and the same packet loss, Skype reacts differently in both scenarios. While in the dummynet scenario the sender's througput is increased, in the traffic shaping router scenario the sender sends constantly with roughly 24 kbps (which will be discussed later in Section 4.1. The reason for this behaviour is the way how packet loss occurs in both scenarios. Therefore, we consider the *inter packet loss distance*. The distance L between two consecutive packet losses is referred to as inter packet loss distance. In the

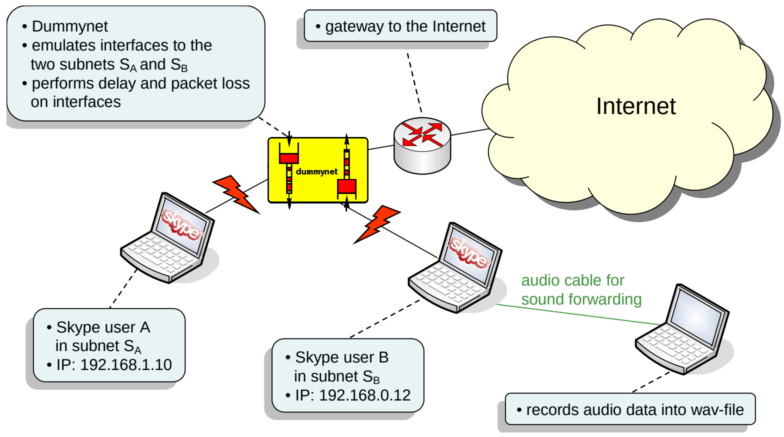


Figure 2: Bottleneck LAN scenario using dummynet software

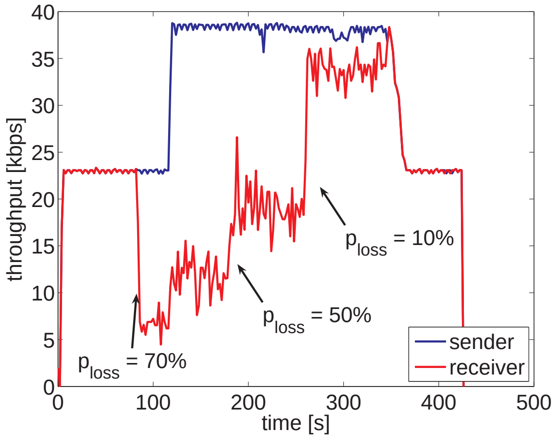
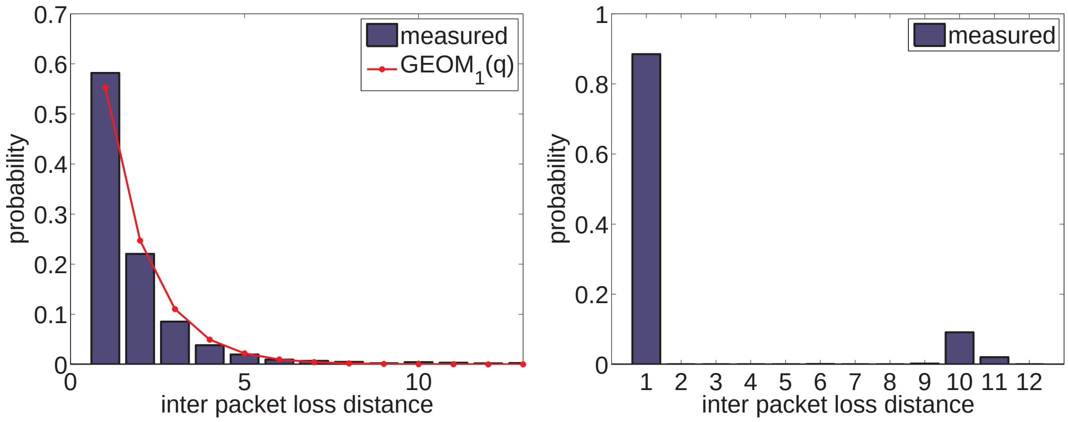


Figure 3: Using dummynet, Skype reacts on packet loss by increasing the sender throughput

dummynet scenario, the random variable L_{dummy} follows a geometric distribution shifted by one: $L_{dummy} \sim \text{GEOM}_1(q)$ with parameter $q = \frac{\mu-1}{\mu}$ and a mean measured distance μ. The number of consecutive packets without any packet loss is denoted as K. It holds: $K = L - 1$.

Figure 4 compares the inter packet loss distance for the dummynet and the traffic shaping router scenario. In the dummynet scenario, the packet loss probability was set to 50 percent. In the traffic shaping router scenario, the bandwidth on the link was restricted to 16 kbps which results in a packet loss ratio of 48 percent. The dummynet software leads to a geometric distribution with $q = 0.5$ in Figure 4(a). This means each packet gets lost with propability q. The traffic shaping router, however, leads to a completely different behaviour. Packets do not get lost independent of each other, but in bursts, which can be seen from Figure 4(b). Due to the nature



(a) Dummynet

(b) Traffic shaping router

Figure 4: Distance between two consecutive packet losses

of wireless channels and in overload situations in a system, error bursts might better model the packet loss behavior of Skype in 3G networks. Therefore, in Section 4.1 we only consider the bottleneck LAN scenario using the traffic shaping router.

2.2 UMTS Setup

In the UMTS scenario, one Skype user is connected to the Internet using a public UMTS operator. We used a Vodafone Mobile Connect UMTS PC-card as modem for the machine. Currently, only dedidacted channels with fixed bandwidths are used. While the uplink capacity is limited to 64 kbps, in the downlink direction a bandwidth of 384 kbps is available. The other Skype user is connected via DSL and has capacity of 128 kbps in the uplink and 1024 kbps in the downlink. We investigated the *uplink scenario* in which the UMTS subscriber sends the audio data with 64 kbps to the DSL user. In the *downlink scenario* the DSL user sends its data over the 128 kbps link to the UMTS user.

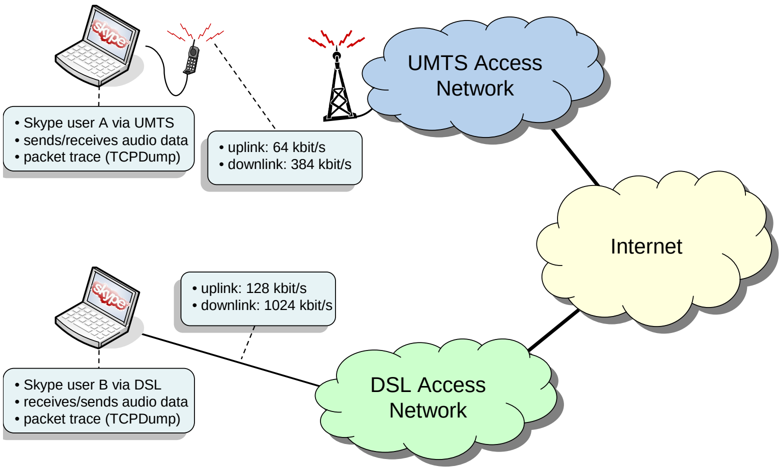


Figure 5: Measurement setup for UMTS scenario

3 Performance Objectives

The performance objective of the measurements is the evaluation of the perceived voice quality. The International Telecommunication Union (ITU) recommends the Perceptual Evaluation of Speech Quality (PESQ) method described in ITU-T P.862. PESQ was specifically developed to be applicable to end-to-end voice quality testing under real network conditions, like VoIP, POTS, ISDN, GSM etc. PESQ requires the sent audio wav-file and the received wav-file as input and returns as result a value between -0.5 (worst) and 4.5 (best), though in most cases it is between 1 and 4.5. However the PESQ score tends to be optimistic for poor quality speech and pessimistic for good quality speech. The degradation of the voice quality because of the Skype iLBC codec (with 30ms frames) and the measurement setup for saving the received audio data is illustrated in Figure 6. Hence, the PESQ value of 3.93 has to be used as reference value for the measurements results in Section 4.

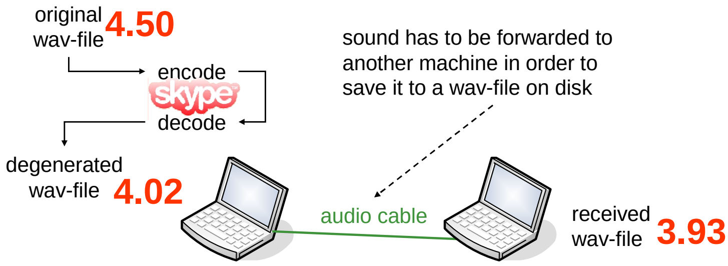


Figure 6: Reference PESQ value without influence of network

The network-based factors which influence the perceived voice quality are the received throughput, the packet loss, the delay, and the delay jitter. The throughput includes the payload as well as the UDP and IP headers. The headers of the data link layer and the physical layer are neglected in order to compare the throughputs in the bottleneck LAN scenario and in the UMTS scenario. The jitter is described by the *packet interarrival time* (PIT), i.e. the time difference between two consecutive packet arrivals.

While it is a trivial task to measure the network-based factors by capturing a packet trace at the receiver, the factors have to be mapped to a value describing the influence of the network on the speech quality. The PESQ reduction imposed by the network connectivity between sender and receiver is described by the *Network Utility Function* (NUF) U_{Netw} introduced in [4]: $PESQ_{rcvd} \simeq U_{Netw} \cdot PESQ_{sent}$. In general, the NUF consists of a number of factors, each reflecting an impairment on the quality, $U_{Netw} = \prod_i U_i \quad U \in \{0,1\}$. In [4] the latter is concretised as $U_{Netw} = U_m \cdot U_s$. The factor U_m is called *m-Utility Function* (m-UF) and captures changes of the mean throughput during an observation window of duration ΔW from m_{sent} at the sender to m_{rcvd} at the receiver, which is reflected by the *loss ratio* $\ell = \max\left\{1 - \frac{m_{rcvd}}{m_{sent}}, 0\right\}$. The factor U_s is denoted as *s-Utility Function* (s-UF) and captures changes of the standard deviation of the throughput from s_{sent} to s_{rcvd} also during an observation window ΔW. The throughput values used for calculating the standard deviation are averages during short intervals of duration ΔT. The relative change of the standard deviation is denoted by $\sigma = \frac{s_{rcvd}}{s_{sent}} - 1$. The concrete application of the NUF concept will be discussed in Section 4.3.

4 Measurements and Results

Skype uses different codecs to maintain reasonable call qualities at an available bandwidth of at least 32 kbps [2]. In our measurements, we only observed the use of the iLBC codec [6] indicated in the technical information field of the Skype application.

4.1 Bottleneck LAN for Emulating Various Wireless Channel Conditions

In this section we concentrate on the bottleneck LAN scenario, which is used to emulate the dynamically changing conditions of a UMTS system. We started with a link bandwidth of 16 kbps and increased the available bandwidth during the Skype call to 32 kbps, 64 kbps, 128 kbps and 384 kbps respectively. During the tests, we observed that the measured PESQ values are very sensitive to small changes in the range between 16 and 64 kbps. Therefore, we additionally measured bandwidth restrictions to 24 kbps, 28 kbps, 40 kbps, 48 kbps and 56 kbps.

The results are summarized in Figure 7, which plots the available bandwidth against the achieved throughput of the VoIP call. Note, that the throughput includes the payload (67 Bytes) as well as the UDP and IP headers (28 Bytes). Each scenario was repeated between five and ten times in order to produce credible emulation results. The figure shows the mean values of the different emulation runs as well as the corresponding minimum and maximum. The first interesting observation is that the sending rate is not adapted to the available bandwidth or the packet loss respectively. The sender constantly uses a bandwidth close to 26 kbps. The communication partner receives a throughput, which corresponds to the currently available bandwidth on the link. That is, all remaining packets are lost on the bottleneck link.

If we use a software tool, like dummynet [7] to generate packet loss, Skype increases the sending rate to roughly 40 kbps in order to push away other traffic, assuming packet loss to be caused by congestion in the network. In the bottleneck scenario, Skype does not adapt its sending rate to the packet loss. The reason seems to be that packets are not lost randomly (as generated by dummynet) but in bursts, what will be explained later.

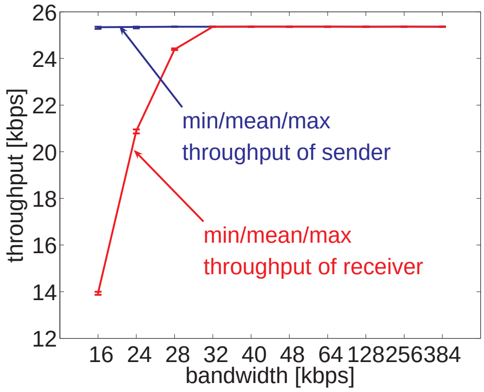


Figure 7: Throughput of sender and receiver

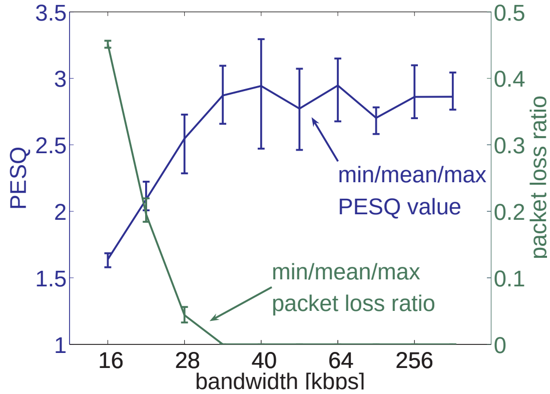
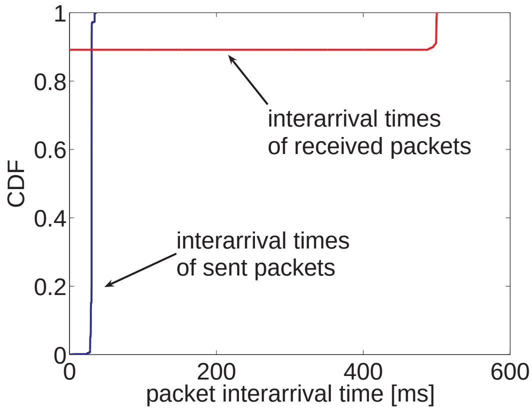


Figure 8: PESQ and packet loss probability

Before that, we have a closer look at the PESQ value in this scenario as illustrated in Figure 8. The figure shows the achieved PESQ value for different link speeds between 16 and 384 kbps and relates them to the observed packet loss. Obviously, the higher the packet loss, the lower is the corresponding PESQ value. There is no more packet loss above a link speed of 29 kbps since up from this point the throughput of the sender (109Byte/30ms) is smaller than the available bandwidth on the link. The corresponding PESQ values oscillate around a value of 2.9. Since the PESQ value is a very sensitive performance measure, the fluctuations can be explained by the stochastic influences of the network, like jitter.

In order to understand the details of the bottleneck in this scenario, we focus on a single emulation run. Figure 9 shows the CDF of the packet interarrival time for both the sender and the receiver of the VoIP call, using a bottleneck speed of 16 kbps. The sudden jump from 0 to 1 at the CDF of the sender illustrates an almost constant time of 30ms between two sent



	payload [Byte]	observed packets	interarrival time
sender	3	3	20.05 s
sender	67	1888	30.00 ms
rcver	3	3	20.25 s
rcver	67	1039	54.43 ms

Figure 9: Single run with 16 kbps bottleneck and 8000 bit buffer in the router

packets. At first glance, the CDF of the receiver has a very unexpected shape. About 90 percent of all packets have an interarrival time of practically 0ms, while the time between the remaining packets is about 500ms. The buffer in the router was set to 8000 Bit, while simultaneously limiting the speed of the link to 16 kbps. Skype used a total packet size of 872 Bit. Thus, at most 9 packets (872*9= 7848 Bit) fit into the buffer of the router. To emulate a link speed of 16 kbps, the router fills its buffer and delays the data for exactly 500ms. This way, a bandwidth of 8000 Bits/500ms = 16 kbps is achieved on a physical 100Mbit/s link. This has two major implications. At first the interarrival time of the packets within a burst is 872 Bit/100 Mbit/s, which is in the range of 10^{-6} s ≈ 0 ms and explains the shape of the CDF in Figure 9. Secondly, packet loss occurs in bursts during the 500ms, in which the buffer of the router is delayed.

Finally, the table in Figure 9 shows the payload size and the observed number of packets of this size for both the sender and the receiver, as well as the corresponding interrarival times. The packets with a payload of 67 Bytes represent the voice connection. Only 1039 of the 1888 sent packets arrive at the receiver, which corresponds to a packet loss of about 45 percent and explains the increase in the average interarrival time from 30ms to 54.43ms. An interesting observation is that Skype obviously sends some smaller packets (3 Bytes) every 20 seconds, which we believe to be quality feedback. This would also explain how a Skype client is able to display the packet loss of the remote side in its technical info window during an active call.

4.2 Measurements in UMTS Network

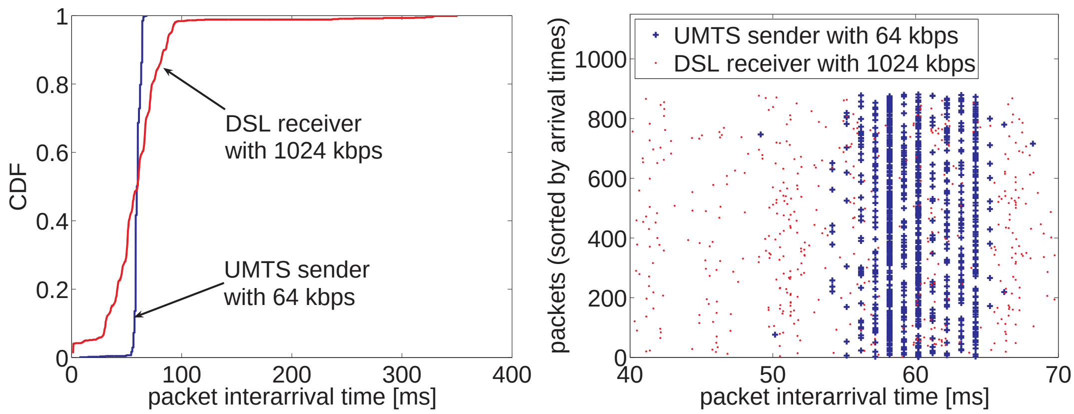
All experiments in this section were done using our UMTS setup as described in Section 2.2. During all measurements the clients used a different codec for the main audio connection, sending 108 Byte of payload every 60ms instead of 67 Byte every 30ms like in the previous scenarios. Since this codec was used starting with the first audio packet, Skype seems to choose this codec based on local information, like access type (modem or LAN) to the Internet, or due to exchanged packets before measuring the link quality. This assumption is supported by the fact, that emulating the link properties (delay, bandwidth) with dummynet did not cause Skype to use the same codec. There was nearly no packet loss in any of the experiments. In total, 11 out of

15417 packets were lost in the UMTS measurements. However, the PESQ values are lower than before because of the network jitter. In the following we concentrate on the packet interarrival times (PIT).

4.2.1 Uplink: UMTS subscriber sends to wireline user

In this uplink scenario the UMTS client uses a 64 kbps connection to send its data to the DSL user, which has a maximum download capacity of 1024 kbps.

Figure 10(a) shows the CDF of the packet interarrival time (PIT) for both the sender and the receiver. The UMTS client constantly sends a voice packet every 60 ms. However, due to the jitter in the network the PITs at the receiver side are spread around the mean. The almost symmetric shape of the CDF reflects the fact that for every delayed packet there is obviously a packet with a correspondingly smaller PIT.



(a) Cumulative distribution function

(b) Trace of packet arrivals

Figure 10: Packet interarrival times in the UMTS uplink scenario

To illustrate this effect, Figure 10(b) plots the PIT for each packet at the sender and the receiver. There were 878 packets in this scenario. The x-axis shows the PIT, the y-axis plots the packet of the corresponding number as sorted by their arrival time. That is, the plot shows how many packets arrived with a specific PIT. As expected, the packets of the DSL user are randomly distributed due to the jitter in the network. The UMTS packets, however, are sent at a discrete resolution of 1ms as can be seen by the vertical lines in the figure. Note that this discretization already happens at the sender and is thus locally influenced, probably by the PCMCIA UMTS card. We are therefore able to exclude buffer effects and the like in the core network for the same discretization in the downlink scenario.

Table 1 presents a more detailed view on some key performance measures for the uplink scenario using a ΔT of 300ms. In particular, it shows the average throughput (m_{sent}, m_{rcvd}) and the average deviation (s_{sent}, s_{rcvd}) for ten different runs of the experiment as well as the corresponding standard deviation for each measure. Since there is almost no packet loss in this scenario, the throughput of the receiver corresponds to the throughput of the sender. The

Table 1: Key performance measures for UMTS uplink scenario

	throughput m_{sent} [bps]	deviation s_{sent} [bps]	goodput m_{rcvd} [bps]	deviation s_{rcvd} [bps]	PESQ Q
μ	18071.58	2300.95	18055.23	3497.57	2.24
σ	8.84	568.87	21.20	858.38	0.16

Table 2: Received packets in the UMTS uplink scenario

payload	number	mean PIT	std. PIT
3 Byte	3	20.02 s	10.73 ms
108 Byte	847	61.97 ms	35.00 ms
112 Byte	28	1.92 s	27.05 ms

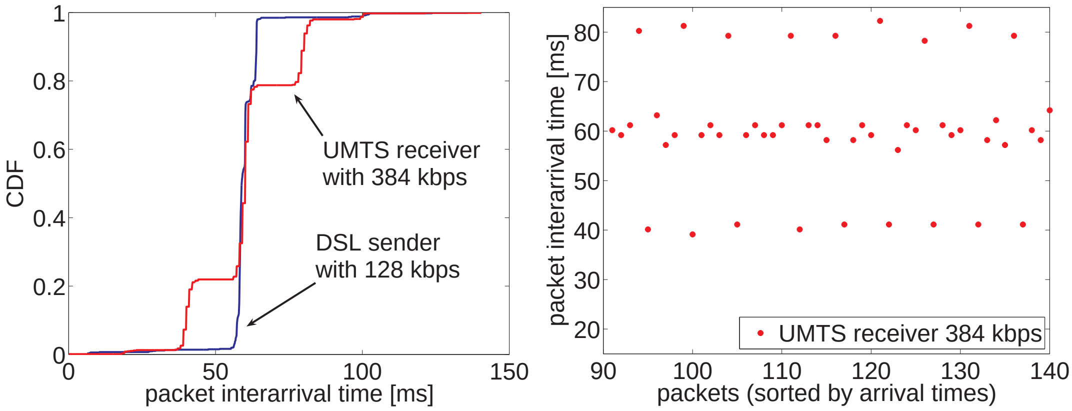
corresponding standard deviation (8.84 bps) of the individual runs is close to zero, as the same codec with a fixed payload size and PIT was used in each of the ten experiments. However, s_{sent} and s_{rcvd} differ by about 1200 bps in the uplink. Due to the jitter in the network the observed PITs are almost uniformly spread around the mean PIT, which is also reflected in the lower PESQ value (2.24) as compared to the bottleneck LAN scenario (2.95) with the bandwidth restriction to 64 kbps.

To highlight these effects in more detail, Table 2 shows the packets received at the DSL client during a single run of the experiment. As before, the 3 Byte packets are used for quality feedback. However, this specific Skype codec uses two types of packet (108 Byte and 112 Byte) in the main audio stream. Thereby every 32th packet is of size 112 Bytes, which explains why the mean PIT of the 108 Byte packets is 61.97ms instead of 60ms. The high standard deviation of the PIT of these packets confirms our previous statements. The same behavior was observed in all nine remaining experiment runs.

4.2.2 Downlink: Wireline user sends to UMTS subscriber

In this scenario we regard the opposite direction, where the DSL user sends its voice data over a 128 kbps link to the UMTS user, who has a downlink capacity of 384kbps. Thereby the interesting effects occur on the link from the base station to the UMTS mobile.

Figure 11(a) shows the CDF of the PIT at the DSL sender and the UMTS receiver. Like before, the packets are sent into the network exactly every 60ms. The UMTS receiver, however, registers a different behavior of the incoming packets. The PIT of the arriving packets is no longer uniformly spread around the mean PIT, but mainly takes three discrete values, 40ms, 60ms, and 80ms. The difference of these values, corresponds to a UMTS Transmission Time Interval (TTI) value of 20ms. As can be seen by the CDF, about 60 percent of all packets arrive with a PIT of 60ms at the UMTS receiver. Approximately every 5th packet misses the corresponding TTI, cf. Figure 11(b), and subsequently arrives with a PIT of 80ms. Therefore the next packet, which hits the correct TTI, has a PIT of 40ms instead of 60ms. This means that 20 percent of all packets have a PIT of 80ms and 40ms, respectively.



(a) Cumulative distribution function (b) Trace of received packets

Figure 11: Interarrival times of received packets in the UMTS downlink scenario

Table 3 gives a more detailed view of the key performance measures in the downlink scenario. The mean throughput of the receiver again corresponds to the throughput of the sender. This time, s_{sent} and s_{rcvd} do not differ as much as in the uplink scenario. Thus, the network should have less influence on the user perceived quality of the audio connection. The PESQ is indeed higher in the downlink scenario (2.49) than before in the uplink scenario (2.24). Note that the standard deviations in the last row of the table are slightly higher, since the number of quality feedback packets varied in the different experiment runs.

Table 4 summarizes the four different packet sizes at the UMTS receiver in this scenario. Again for the most part 108 Byte packets were used for the audio connection, while this time only every 54th packet had a payload of 112 Bytes. In exchange, there is a new packet type using 21 Bytes. This kind of packet was also used in the audio connection, replacing some of the 108 Byte packets. However, they were sent very irregularly as can be seen by the high standard deviation of their PIT. The same irregularity was obtained for the 3 Byte packets, which did not have a deterministic PIT of 20s but were sent every 10s on average with a standard deviation of 4.49s. What exactly triggers Skype to use this specific variation of the codec is subject to further study.

Table 3: Key performance measures for UMTS downlink scenario

	throughput m_{sent} [bps]	deviation s_{sent} [bps]	goodput m_{rcvd} [bps]	deviation s_{rcvd} [bps]	PESQ Q
μ	18023.77	1848.15	18007.08	2172.39	2.49
σ	48.16	282.70	51.64	284.97	0.085

Table 4: Received packets in UMTS downlink scenario

payload	number	mean PIT	std. PIT
3 B	6	9.46 s	4.49 s
21 B	14	1.73 s	3.58 s
108 B	817	61.32 ms	16.00 ms
112 B	16	3.20 s	45.02 ms

4.3 Network Utility Function

We investigate the impact of the loss ratio on the NUF, i.e. on the PESQ, by studying the bottleneck LAN scenario described in Section 4.1 and using $\Delta T = 150$ ms. By reducing the speed of the bottleneck link beyond the need of the Skype connection, we provoke loss whose intensity is roughly given by the missing capacity share. Figure 12 shows a scatter plot of PESQ versus the loss ratio ℓ. Even though the PESQ values are highly varying even for similar loss ratios, a linear trend can be observed in Figure 12 as ℓ increases (as assumed in [4]). Simple regression analysis for $\ell \leq 0.2$ (i.e. a link speed of at least 24 kbps) leads to $U_m \simeq 1 - 1.5\,\ell$. Interestingly enough, the change in the relative standard deviation jumps from $\sigma \simeq 0$ to $\sigma \simeq 0.3$ as soon as loss due to overload occurs, while the damping of the PESQ as a function of the overload intensity is sufficiently described by the m-utility function alone. This behavior is different from the one observed in [4], where the sources of loss and delay variations were independent of each other. On the other hand, there is an impact on the PESQ which is due to the fact that a network is used, which is captured by the *n-utility function* $U_n = 2.90/3.93 \simeq 0.74$ (cf. Sect. 3). Thus, the complete NUF in the bottleneck LAN case reads $U_{Netw} \simeq U_n \cdot U_m$ and $U_s = 1$.

In the UMTS scenario (cf. Section 4.2), no significant loss has been observed, which means $U_m \simeq 1$. However, as seen from the scatter plot shown in Figure 13, uplink and downlink

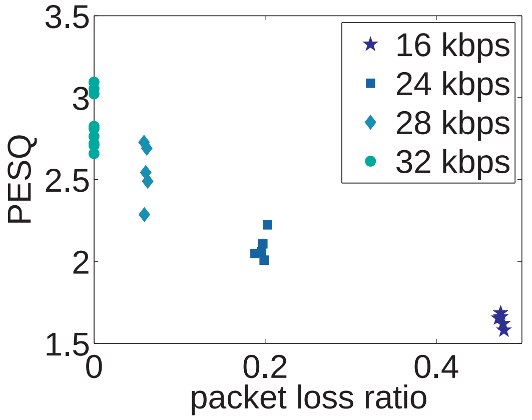


Figure 12: Packet loss probability ℓ vs. PESQ for the bottleneck LAN scenario

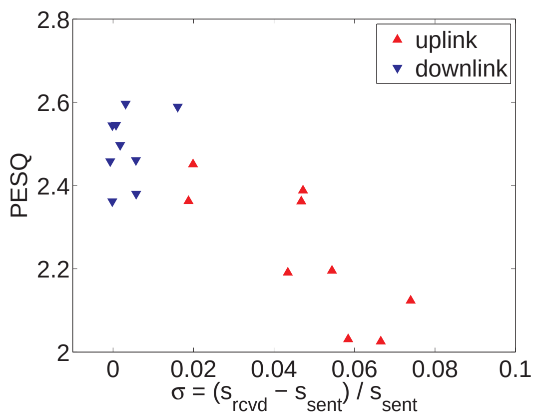


Figure 13: $\sigma = (s_{rcvd} - s_{sent})/s_{sent}$ vs. PESQ for the UMTS scenario

experience different intensities of throughput variations [8], which allows us to take approximate it as follows: $U_s \simeq 1 - 2\sigma\ (0 \leq \sigma < 0.1)$. As discussed in Section 4.2, the origin of the PESQ values is not the same as in the bottleneck LAN case (2.49 for $\sigma = 0$ instead of 2.90, cf. Fig. 13 with 12). We can capture this effect, caused by the mobile network and Skype's selection of an appropriate codec, by modifying the value of the n-utility function to $U_n = 2.49/3.93 \simeq 0.63$. Summarizing, the NUF as a whole reads $U_{Netw} = U_n \cdot U_m \cdot U_s$.

5 Conclusion and Outlook

VoIP telephony has become a strong competitor to existing telephone networks. In this study, the user perceived quality of Skype calls over UMTS was measured and analyzed in terms of PESQ values and network-based influence factors, like packet loss, throughput, and jitter.

We emulated the rate control mechanisms in UMTS by restricting the link bandwidth with a traffic-shaping router. In this case, Skype does not react to packet loss, as e.g. by increasing the sender throughput, but constantly sends audio data with roughly 25 kbps. The occurring packet loss degrades the PESQ value. For bottleneck links below 32 kbps, a linear dependency of the PESQ value on the loss was discovered, while it varies around 2.9 for links of higher bandwidth. The measurements in a public UMTS networks revealed that the capacity offered by UMTS is sufficient to make mobile VoIP calls possible. However, due to network jitter and the use of a different codec by Skype, the PESQ values are worse than those in the emulation of the bottleneck in a LAN environment. The used UMTS card sends and receives packets at discrete time instants in multiples of 1 ms. The packet interarrival times on the downlink are multiples of 20 ms, which corresponds to a common transport time interval TTI in UMTS.

Finally, we showed that the concept of the Network Utility Function can be applied to describe the impact of the network on the PESQ. Thereby the network-based influence factors are sufficient to approximate the PESQ values. The next step will be to investigate the dependency between loss and changing traffic characteristics and the corresponding impact on the PESQ and the utility functions.

Acknowledgements

The authors would like to thank Melanie Brotzeller for performing the measurements, programming all scripts, and the fruitful discussions; Thomas Zinner, and Frank Bennewitz for helping during the measurements; Stefan Chevul for the much appreciated input during the course of this work; and Frank-Uwe Andersen and the Siemens AG for supporting the measurements.

References

[1] Skype Technologies S.A., http://www.skype.com/.

[2] Baset, S., Schulzrinne., H.: An analysis of the skype peer-to-peer internet telephony protocol. Technical Report CUCS-039-04, Columbia University, New York (2004)

[3] Skype Technologies S.A., http://www.skype.com/company/news/2005/skype_eplus.html.

[4] Fiedler, M., Chevul, S., Radtke, O., Tutschku, K., Binzenhöfer, A.: The Network Utility Function: A practicable concept for assessing network impact on distributed systems. In: 19th International Teletraffic Congress (ITC19), Beijing, China (2005) 1465–1474

[5] Signalogic, Speech Codec Wav Samples, http://www.signalogic.com/index.pl?page=codec_samples.

[6] Andersen, S., Duric, A., Astrom, H., Hagen, R., Kleijn, W., Linden, J.: RFC 3951: internet low bit rate codec (ilbc) (2004)

[7] Rizzo L., Dummynet, http://info.iet.unipi.it/~luigi/ip_dummynet/.

[8] Isaksson, L., Chevul, S., Fiedler, M., Karlsson, J., Lindberg, P.: Application perceived throughput process in wireless systems, ICMCS’05, Montreal, Canada (2005)