An Architecture for a Next Generation VoIP Transmission Systems

Christian Hoene[1], Kai Clüver[2], Jan Weil[2]
[1]Wilhelm-Schickard-Institute, University of Tübingen, Germany
[2]Fachgebiet Nachrichtenübertragung, Technical University of Berlin, Germany

1.1 Abstract

Packetized speech transmission systems implemented with Voice over IP are gaining momentum against the traditional circuit switched systems despite the fact that packet switched VoIP is two to three times less efficient then its circuit switched counter part. At the time same time it is supporting just a rather bad "toll" quality. We believe that it is time to for a new architecture developed from the scratch. An architecture that includes an Internet enabled speech codec and its transport system. This architecture manages the perceptual service quality while using the available transmission resources to its best. The transmission of speech is managed and controlled in respect to its speech quality, month-to-ear delay, bit-rate, frame-rate, and loss robustness. Beside the architecture, we describe the requirements for the Internet speech codec and its transport protocol and present the description of an interface between speech codec and transport protocol.

2 Introduction

Internet Telephony is a mature technology that has gained increasing popularity against the traditional PSTN systems. Voice over IP (VoIP) is replacing the PSTN service on broadband access networks such as cable modems and DSL, as it is more cost efficient to use IP broadband access also for Internet telephony. In addition, future wireless broadband access networks such as the 3GPP's Long Term Evaluation (LTE) radio technology will support telephone services only via VoIP [1].

Despite the success of Internet Telephony it has a fundamental drawback. It is much less bandwidth efficient than its classic circuit switched counterpart. VoIP requires two to three times more physical gross bandwidth than a modern circuit switched speech transmission in DECT, GSM, or UMTS networks. If more bandwidth is required, other performance parameters are be sacrificed, too: the typical talk time or its transmission range of a mobile, portable VoIP telephone is shorter because more energy is required to support the transmission of packetized voice.

If we compare commercial, modern mobile and cordless phones, one can see that a DECT telephone using circuit switched technologies has a talk time at least three time longer than a WLAN cordless phone – assuming similar battery capacities. Also, using the circuit switched GSM technology the transmission range is 10 to 100 times larger than using VoIP-WLAN technology, if the telephones have the same battery capacities and talk times[1].

Taking these facts in consideration, one can say that a circuit switched based telephone call is far more efficient than its VoIP-WLAN counter part. Because also other portable VoIP based phones have similar operational specifications, we believe the lacking of efficient transmissions of the current VoIP architecture is fundamental and valid regardless of any implementation details and product models.

Traditionally, VoIP uses speech compression schemes, which have been designed for circuit switched telephone systems in mind, such as ISDN or GSM, and have a static frame rate and packet loss robustness. In the Internet, many more transmission parameters need to and can be controlled and managed. These include – beside the bit rate of the speech coder – the frame and packet rate, the loss robustness, and the algorithm delays. We believe that it is necessary to develop both a speech codec and a transport protocol that are optimized for the path characteristics of the Internet. They shall be aware about the current transmission resources and the perceptual quality of the ongoing telephone call in order to adapt their transmission parameters autonomously.

---

[1] These statements are based on a comparison of the specifications of commercial phones. As an exemplary DECT based cordless phone, we have chosen the Siemens Gigaset S44, which comes with a battery of 750 mAh, has a talk time of 10h, and has a transmission range up to 300m. As example for both GSM and VoIP-WLAN we take the Nokia E70 model, which has a battery capacity of 970mAh. In the GSM mode, it has a talk time between 3.3 and 6.4 hours and a transmission range up to 35 km. In the VoIP/WLAN mode using IEEE 802.11g it has a talk time between 3 and 3.2 hours and a transmission range similar to the DECT phone.

Recent research results, to which we will refer in the following sections, have shown that current VoIP systems can indeed be significantly enhanced, both in terms of efficiency and quality. To gain efficiency we cannot be backward compatible nor support the classic speech coders or transport protocols such as ITU G.729 or IETF RTP. Instead, we need to break with the past and make a new start. If one took the freedom to design a new VoIP system from the scratch, how would it look like?

In the following section we will propose a new architecture on how to develop an efficient speech transmission system including a speech coding framework and a transport protocol. Also, referring to previous research results, we will describe the motivation behind our design decisions. In section 4 we will go into details and describe an interface between speech codec and transport explaining which parameters are exchanged. Finally, we will give an outlook to the upcoming the design and implementation of the new for the Internet optimized speech codec and its corresponding transport protocol.

3 Architecture

The next generation VoIP architecture shall consist of a speech codec, optimized for the Internet, and a corresponding transport protocol. The transmission shall be bidirectional as telephone calls are bidirectional as well. Figure 1 gives a first overview on the component of the architecture. It displays just one side of the transmission. However, the other side shall be build similarly. In the following, we describe the components individually.

3.1 Quality of the Telephone Call

In order to optimize the transmission of the telephone call perceptual quality models, which simulate the human rating of the quality of telephone calls, shall be applied. The foremost quality model to mention is the ITU's E-model that is intended to as a planning instrument for telephone systems [5]. It considers most of the parameters that have an effect of the transmission quality, such as the loudness of speech signal, the noise levels, the loudness of echoes, the speech quality, and the acoustic mouth-to-ear (M2E) delay. It calculates an overall quality rating called the R factor that ranges from 0 (worse) to 100 (very good). Beside its primary purpose to plan transmission systems, it can also be applied at real time to control a transmission and set the various transmission parameters [6].

In the novel VoIP architecture a quality model similar to the E-Model is of the utmost importance as it gives an overview on which parameters need to be optimized to achieve a high transmission quality. Also, a trade off between speech quality and delay will be possible.

We can also derivate the first building blocks of the architecture, namely the control of loudness with an adaptive gain control (AGC), the cancellation of echoes by an acoustic echo cancellation (AEC), and the determination of the intrinsic delay of a telephone, which are the sum of all delays that the telephone adds to the overall month-to-ear delay. In order to properly approximate the mouth-to-ear delay, the telephone shall determine the intrinsic latency of the speech signal. For example, the AEC can be used to determine this delay.

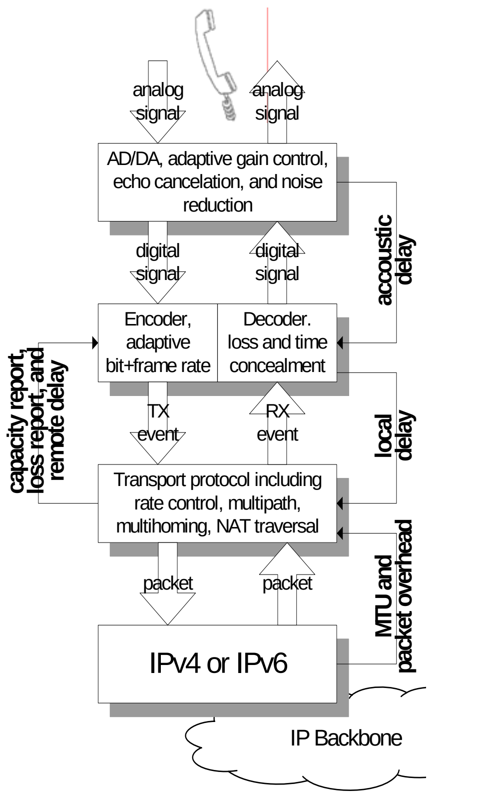

Figure 1: Architecture for a Next Generation VoIP transmission system.

3.2 Speech Codec and Concealment

In the last years many speech codecs, comprising of speech encoder, speech decoder, and loss concealment algorithms, have been developed and are applied in PSTN, cellular networks, and VoIP networks. The speech codecs include ITU G:711, ITU G.729, ETSI GSM-EFR, 3GPP AMR, 3GPP AMR-WB, 3GPP2 VMR-WB, and IETF iLBC. They have optimized to provide a superior speech quality, a low algorithmic delay, a low computational complexity, and a high packet loss robustness. At the same time, they require a low transmission bit rate. If this was the case, why should we consider the development of new speech coders if the existing ones are perfect?

Three arguments, based on recent research results, have given us the insight that the current speech codecs might not be perfectly matched for the requirements of the Internet. The first is based on the observation that the losses of speech frame can have a quite different impact on the speech quality and that many low rate speech codecs still allow a high loss rate without a hearable degradation of the speech quality. The second is based on the observation that low bit rate it not the only transmission parameter that is of importance in a packetized network. The third argument simply accounts for the observation that telephones are not only used for human to human conversation but increasingly frequent for music listening and music exchange.

3.2.1 The unequal impact of losing speech frames

For a long time it has been known that the impact of speech frame losses can differ widely. Some losses, even during voice activity, are hardly hearable. Others have a notable negative impact on the speech quality. Just recently, one of the authors has investigated systematically this effect [7]. A measurement procedure has been developed to quantify the impact of single packet or speech frame losses. This measurement procedure has been verified by formal listening-only tests to ensure its precision. A metric was also developed that describes the impact of losses on speech quality quantitatively.

Using the importance of speech frames, simulation and listening tests show that many speech frames can be dropped during active voice because the receiver side loss concealment works so well that the losses are hardly notable [7]. These studies were conducted for G.711, G.729, and AMR encoded voice and loss rates up to one third (during voice activity) still allow understandable speech transmissions. Thus, knowing the importance of speech frames, significant performance gains can be achieved if only important packets are transmitted.

As a result of these research studies, one can say that the speech coders under study still contain a high level of redundancy because many speech frames need not to be transmitted (or can be dropped intentionally). Also, the information about the speech is unequally distributed among the speech frames as some frames are important and others are not. Would it not be better if all speech frames had the same importance and all speech frames contained the same amount of information? Then, each packet loss would have a similar impact on the degradation of speech quality.

It can only be achieved if the size of the speech frames are variable(such as in the 3GPP2 VMR-WB speech codec [3]) or if the rate of frames varies over time. Then, if the current speech signal contains a lot of new information, the encoder would produce larger or more speech frames, otherwise the encoder would produce smaller or less speech frames[2].

We assume that future speech codecs, optimized for the Internet, will generate speech frames of similar importance. The speech codecs will have variable frame size and/or variable frame rates.

3.2.2 Bit rate and frame rate

Many speech codecs of today support multiple bit rates. For example, the AMR codec supports eight compression rates ranging from 4.75 to 12.2 kbps. Others, like the speex Codec, support a bit range from 2.15 to 44.2 kbps. If in VoIP system a highly efficient transmission shall be achieved because, for example, bandwidth or energy is scarce, then often the lowest bit rate is chosen. A low bit rate has low bandwidth requirements and fewer bits per second need less transmission energy.

Again, recent research results have shown that the bit rate is not the only factor that influences the transmission efficiency: The packetisation can be of equal importance. Packetisation describes how many speech frames, produced by the speech encoder, are put into a VoIP packet before the packet is transmitted. Many speech coders produce every 10, 20, or 30 ms a speech frame. Many VoIP telephone transmit those frames in VoIP packets every 20, 40, or 60 ms. Thus, one VoIP packet contains one or multiple speech frames.

If more speech frames are put into one VoIP packet, a longer time has to be waited before the VoIP packet can be transmitted. Thus, the algorithmic delay of the packetisation increases. On the other side, if less VoIP packets are transmitted per second, then the gross bandwidth is reduced because less protocol headers such as IP, UDP, and RTP need to be transmitted.

If now the bandwidth is limited, shall the coding rate be reduced or the packetisation increased in order to save bandwidth? Simulations have been conducted in [7] to answer this question. The results showed that the answer to this depends on the underlying technology. On a traditional circuit switched connection, which does not transmit packet headers, the reduction of the bit rate achieves the best quality. On switched Ethernet links using an AMR codec, both bit and packet rate shall be adapted. And, finally, on an IEEE 802.11b wireless LAN using an AMR codec it is sufficient to decrease only packet rate to save a significant share of the bandwidth.

[2] Indeed, the AMR's discontinuous transmission (DTX) algorithm produces smaller and less frequent speech frames during silence. However, during voice activity the frames have all a constant size and are produced every 20 ms.

The results show that Internet optimized speech shall not only support a low and variable bit rate. The frame rate is of similar importance. This means, a speech coder shall not produce frames at a constant rate but shall reduce the packet rate, whenever this is possible without sacrificing the perceptual service quality.

We believe that for the Internet optimized speech coders shall be able to produce speech frame at any point of time. For example, speech frames can be generated if the current change of speech characteristics requires to do so. An Internet speech codec must not follow the strict rule of a constant time interval.

3.2.3 Limitations of the frequency band

Quite frequently it can be seen that mobile phones are not only used for human to human communications but for many other purposes like listening to music, exchanging ring tones, listening to the radio, and many more. We assume that in future also telephones will be required to transmit, beside speech, also musical content.

Current speech codecs are intended for the transmission of human speech (and background noise). Recently, enhancement such as 3GPP's AMR-WB+, the AAC-Low delay, and Fraunhofers Ultra Low Delay (ULD) codec support the transmission of music at real time. However, current VoIP telephone uses codecs that support a "narrow" frequency bandwidth up to 3700 Hz or a "wideband" frequency bandwidth up to 7000 Hz. But in contrast to the traditional PSTN or cellular systems, VoIP has no technical constrains that limit the frequency spectrum. Instead of this, an Internet speech codec shall encode speech and music at the highest quality that the current transmission path can support to transmit.

3.2.4 Loss and time concealment

Packet loss concealment algorithms are placed at the receiving end of a transmission of speech and limit the effect of packet losses [9]. They extrapolate the last speech signal if the current speech frame has not been received. So they limit the negative effect of packet losses on the speech quality. Nowadays, they are often part of a speech codec's standardization document and part of the decoder.

Time concealment tries to cope with the effect of transmission jitter in a way of slowing down or increasing the speed of the current speech [10]. Time concealment algorithms have a positive effect on the service quality but they come at the cost of additional algorithmic delay. Also, if a speech frame has not been received on time, the decoder cannot decide whether to slow down the speech output or whether to conduct loss concealment. At this moment of time, the decoder cannot know whether the packet will still arrive or whether is has been lost.

On the other side, if the decoder would closely follow the delay process of the transmission path, then the overall mouth to ear delay could be reduced significantly. The buffering of speech frame in play out buffer, nowadays included in nearly all VoIP phones, could be omitted. Thus, we suggest to include the loss concealment, the time concealment, and the playout buffer into the decoder. The decoder shall then decide to playback the speech frames as they arrive and conceal, slow down, or fasten the speech, if required.

3.3 Transport protocol

The Internet optimized speech codec shall not operate on the traditional RTP/UDP protocol. Instead, it requires a transport protocol that informs him on the current state and quality of the transmission path. Only if the speech codec knows the current properties of the transmission path can it adapt its coding bit rate and packet rate to achieve a high perceptual transmission quality.

Forward Error Correction (FEC) shall not be a functionality provided by the transport protocol. It can be more easily implemented at the encoder. But then the transport protocol shall inform the encoder about the loss process in the network and the encoder shall change its loss robustness.

The transport protocol shall take advantage of the bidirectional nature of a telephone call and shall transmit speech frame bidirectionally. This has the advantage that control information, nowadays transmitted in signaling packets like RTCP, can be piggy back on the data stream. Thus, the packet rate is reduced further. Also, the transport protocol can implement feedback loops to implement rate and congestion control more easily. Optionally, the transport protocol can support other mechanisms such as multi-homing, mobility, multipath, or NAT traversal in order to increase the reliability and quality of the transmission.

4 Interface Description

After the description of the architecture, this chapter describes how an interface between the speech codec optimized for the Internet and its corresponding transport protocol. This interface description is required, if both speech codec and transport protocol are to be developed separately or if codecs or transport protocols shall be exchangeable.

In this publication we are concentrating on the ongoing transmission of speech. State changes are notified by events. Events change parameters and data between the codec and the transport protocol. To describe the parameters that are exchanged between both entities, we use a Java like pseudo code notion.

4.1 Coding to Transport: Transmit Event

The speech coder notifies the transport layer every time a new frame has been generated. Beside the frame data, its length, and time stamp is required. The length and time stamp can both be dynamical because the speech coder might have a variable speech and coding rate (such as the proprietary codec iSAC from Global IP Sound and 3GPP2's VMR-WB).

```
class TransmitEvent {
   byte data[];  // speech frame and its length
   int ts;       // time stamps defining when the speech signal as been produced (local clock)
};
```

Time stamp is a novel feature but an important one because one cannot assume that speech frames are produced at regular intervals. Also, the time stamp shall be taken at the point of time the speech signal has been spoken or produced.

Given this information, the transport layer can calculate the current bit and frame rates generated by the encoding. Given a set of transmit events called $te[1]$ to $te[n]$ all time stamps shall be increasing. That means, for all $1 \le i < n$, $te[i].ts < te[i+1].ts$. Then, bit rate and the packet rate are calculated as

$$bitrate = \frac{8 \cdot \sum_{i=1}^{n} te[i].data.length}{te[n].ts - te[1].ts} \tag{1}$$

$$packetrate = \frac{n}{te[n].ts - te[1].ts} \tag{2}$$

The main task of the transmission layer is to transmit the frame data, its length, the time stamp, and its increasing index. These parameters shall be transmitted to one (or multiple) destinations. How the transport layer opens and tears down its connection and whether the transport layer uses multiple destinations to support multicast, multiple paths, or any kind of error correction is beyond the scope of this publication.

A second task is to estimate the variability of the flow of speech frames. According to the current situation of the conversation, the variability of speech one the one side and the interactivity of the other side can vary significant. Thus, the rate and size of speech frames can differ substantially. The transport protocol requires an estimate about the variability of transmission rates in order to calculate a safety margin regarding the transmission capacity.

4.2 Transport to Decoding: Receive Event

Similarly, just opposite, the transport protocol hands over speech frames to the decoder as soon as it receives them. It shall not buffer the speech frames. The data parameter includes:

```
class ReceiveEvent {
   byte data[];   // speech frame and its length
   int ts;        // time stamps defining when the speech as been spoken (remote clock)
   int jitter;    // time offset as compared to mean remote round trip time describe in section 3.3.
   short index;   // increasing index number of the speech frame
};
```

The receiver calculates the loss rates using a set of receive events called $re[1]$ to $re[n]$ with for all $1 \le i < n$, $re[i].ts < re[i+1].ts$ and $1 \le i < n$, $re[i].index < re[i+1].index$:

$$packetloss\,rate = \frac{n}{re[n].index - re[1].index} \tag{3}$$

Also, using the time stamps, the decoder can calculate the transmission delay variations. It enables him to get a statistics about the distribution of the transmission delays in order to adapt the play out of the speech frames accordingly.

4.3 Transport and Codec: Round Trip Times Delays

The classic RTP Control Protocol (RTCP) is a signaling protocol to provide feedback on the quality of the transport of multimedia data. The feedback is performed using the by the RTCP sender and receiver reports, which in regular intervals report information about time stamps, byte- und packet counts, loss rates, smooth mean deviation of interarrival times (jitter), and the round trip times [2].

Recently, an the Extended Reports (XR) have been added to RTCP to report more detailed statistics on the network characteristics or quality monitoring [8]. The data provided includes which packets have been lost and received, which packets have been received multiple times, when the packets have been received. Also, it provides the means to gather the network round trip time and the end system delay in order to calculate the acoustic round trip time.

As mentioned above, the mouth to ear delay is an important quality metric that influences the service quality of a telephone call, which needs to be optimized. More precisely, the metric under optimization is the acoustic round trip time, which is the sum of the mouth to ear delays of both transmission directions. Humans cannot distinguish which direction of the transmission contributes to the delay, thus the one-way delay needs not to be known.

The round trip time can be used for both the codec and the concealment. For example, if the RTT is below 150 ms, the codec increases its algorithmic delay to cope better with packet loss or with delay variations.

Both the codec and the transport protocol inform each other, if the mean acoustic delay of each side has changed. The following event format is applied:

```
class RTTchange {
   int delay;      // acoustic round trip time on the local or remote side
};
```

The sum of both values, from the codec and the transport protocol, is the overall acoustic round trip time and twice the mean mouth-to-ear delay. The events are just triggered if the delay has changed significantly, e.g. about more than 10 ms, to avoid unnecessary high number of updates.

4.4 Transmission Capacity

The transport protocol determines the rate at which rate the coder is allowed to produce data. It informs the codec about that this rate. In compliance with TCP, the rate is given in bits per round trip time, which means that the coder is allowed to send this maximal the given number of bits within the next round trip time. The coder is free to choose when he sends the data, either at the beginning, continuously during, or at the end of the RTT period. The capacity of the path can change highly dynamic. Thus, at any time an update regarding the transmission rate can occur.

Depending on the volatility of the coder's rate and the volatility of the network bandwidth, the transport protocol is free to reduce the transmission rate to add a safety margin or the increase the transmission rate in order to achieve a statistical multiplexing gain at the cost of a higher packet loss rate.

TCP sends packets at the maximal transfer unit (MTU) in order to achieve the highest throughput. If a service requires a low transmission delay, then it would not benefit from sending large packets containing a long speech segment but from short packets containing short speech segments.

Usually, the costs of sending many small packets is much higher than sending one larger packet, because each packet has additional packet headers on multiple layers. In addition, the medium access control requires additional resources to transmit a packet.

An example given in [7] studied the transmission over IEEE 802.11b at 11 Mbps in the DCF mode. The cost of the contention period, collisions, and the immediate acknowledgements contribute beside the headers of PLCP, MAC, link, IP; UDP, and RTP significant to the bandwidth requirements of a packet. In total, transmitting one packet, the physical medium of IEEE 802.11b is busy for about one microsecond in addition to the actual data transmission. Thus, the costs of one packet – regardless of its size – correspond to about 1000µs/11MBps ≈ 1500bytes/s in the IEEE 802.11b mode. Packet headers can be easily compressed to a few bytes by using IETF's IP header compression algorithms. But header compression cannot reduce the overhead of the MAC protocol.

In [4], the notion of packet overhead is introduce to determine the amount of overhead required to transmit a packet. If is defined as the gross bandwidth that is required to transmit a packet:

$$t_{overall} = t_{overhead} + \frac{ps_{pdu}}{rate} \quad \Leftrightarrow \quad t_{overall} \cdot rate = t_{overhead} \cdot rate + ps_{pdu} \qquad (4)$$

Defining $p_{overhead}=t_{overhead}\cdot rate$, the packet overhead is the number of bytes that each packet costs. It measures the gross number of bits on the physical medium.

Of course, this value can change with the physical medium, the transmission rate, and many other parameters. If the packet overhead is not precisely known, the transport protocol can guess it by average the packet overhead of various, typical, and commonly used transmission technologies. For this interface description, we apply the notation of packet overhead: The transport protocol signals the coder the current transmission requirements as

```
class Capacity {
   int bps;         // mean bit per second the coder is allowed to produce at maximal during the next round trip time.
   int mtu;         // the maximal transfer unit, the largest packet size a coder is allowed to produce
   int overhead;    // costs of a single packet in bits
};
```

Thus, for the transmit events $i \in \{1;n\}$ within a period of t_{rtt}, the following conditions must be given:

$$te[i].data.length \leq capacity.mtu \tag{5}$$

$$capacity.bps \cdot 8 \cdot t_{rtt} \geq \sum_{i=1}^{n} \left(te[i].data.length + capacity.overhead \right) \tag{6}$$

4.5 Transport to coder: Packet losses

In the Internet packet losses occur during time of congestion. Also, on wireless links transmission error might cause packet losses. Following the solution given in the RTCP XR receiver reports [8] we report packet losses and packet receptions with a bit vector.

```
class PacketLossReport {
   short begin_index;  // the first index number that this event reports on
   short end_index;    // the last sequence number that this event report on plus one.
   int vector[];       // the array of integers is read from left to right, in order of increasing index number
                       // (with the appropriate allowance for a wraparound)
};
```

The coder requires the report about packet losses because then it could adapt its loss robustness and change the amount of redundancy. If many losses occur, the amount of redundancy shall be increased to help the packet loss concealment algorithm. But if the losses hold on for a long time and are bursty, then redundancy could not help and losses would be inevitably audible.

5 Summary and Outlook

We followed the following tenets in our architectural redesign of a VoIP transmission system:

1. Develop a speech codec that has a variable bit and a variable frame rate.
2. Closely couple the speech codec and the transport to achieve the benefits of a cross layer optimization strategy. They shall be aware about the current quality of the call in order to manage and control their transmission parameters.
3. Include Forward Error Correction into the encoder.
4. Combine decoding, loss and time concealment, and the playout buffer into a single Internet enabled speech decoder.
5. Do not stick to a narrow or wide frequency band because beside speech also music transmission will be required.

This publication shall help research to design and implement new architecture for the next generation of VoIP transmission system. But only if this system has been designed, implemented, and tested, can we see to what extent the new architecture can enhance the transmission efficiency and perceptual quality as compared to the classic VoIP system.

6 References

[1] 3GPP TR, „Feasibility study for evolved Universal Terrestrial Radio Access (UTRA) and Universal Terrestrial Radio Access Network (UTRAN)," version 7.1.0, Oct. 2006.

[2] H. Schulzrinne, S. Casner, R. Frederick, V. Jacobson, „RTP: A Transport Protocol for Real-Time Applications", IETF RFC 3550, Jul. 2003.

[3] 3GPP2 C.S0052-A v1.0 "Source-Controlled Variable-Rate Multimode Wideband Speech Codec (VMR-WB) Service Options 62 and 63 for Spread Spectrum Systems", 3GPP2 Technical Specification, Apr. 2005.

[4] C. Mahlo, C. Hoene, A. Rostami, A. Wolisz: Adaptive Coding and Packet Rates for TCP-Friendly VoIP Flows, Proc. 3rd Int. Symp. on Telecommunications (IST2005), Shiraz, Iran, Sep. 2005

[5] ITU G.107, "The E-model, a computational model for use in transmission planning", Mar. 2005.

[6] C. Hoene, H. Karl, and A. Wolisz, "A perceptual quality model intended adaptive VoIP applications", International Journal of Communication Systems, Wiley, Aug. 2005.

[7] C. Hoene "Internet Telephony over Wireless Links", PhD thesis, Technical University of Berlin, TKN, Dec. 2005.

[8] T. Friedman, R. Caceres, A. Clark, "RTP Control Protocol Extended Reports (RTCP XR)," IETF RFC 3611, Nov. 2003.

[9] K. Clüver and P. Noll, "Reconstruction of missing speech frames using sub-band excitation," in IEEE-SP International Symposium on Time-Frequency and Time-Scale Analysis, 1996.

[10] Y. J. Liang, N. Färber, and B. Girod, "Adaptive playout scheduling and loss concealment for voice communication over IP networks," IEEE Transactions on Multimedia, vol. 5, no. 4, pp. 532–543, Dec. 2003.