Internet Telephony over Wireless Links

vorgelegt von Diplom-Ingenieur
Christian Hoene

von der Fakultät IV - Elektrotechnik und Informatik
der Technischen Universität Berlin
zur Erlangung des akademischen Grades
Doktor der Ingenieurwissenschaften
– Dr.-Ing. –
genehmigte Dissertation

Promotionsausschuss:

Vorsitzender:	Prof. Dr. Heiß
Berichter:	Prof. Dr.-Ing. Wolisz
Berichter:	Prof. Dr.-Ing. Steinmetz
Berichter:	Prof. Dr.-Ing. Sikora

Tag der wissenschaftlichen Aussprache: 16. Dezember 2005

Berlin 2006
D 83

Christian Hoene
Jahnstr. 22
72144 Dußlingen
Germany
hoene@ieee.org

As a small child, you never spoke that clearly, no wonder that you want
to improve the speech perceptibility.
My mother, after hearing my thesis topic.

Abstract

This thesis presents algorithms to enhance the efficiency of packetized, interactive speech communication over wireless networks. The results achieved are the following:

We present an improved approach to assess the quality of voice transmissions in IP-based communication networks. We combined the ITU E-Model, the ITU PESQ algorithm, and various codec and playout schedulers to analyse VoIP traces. Parts of this algorithm have been included in ITU standards. By using this assessment approach we derived design guidelines for application and data-link protocols. Also, we developed a quality model to parametrise adaptive VoIP applications. Later results received a best-paper award.

If highly compressed packetized speech is transported over packet networks, losses of individual packets impair the perceptual quality of the received stream differently, depending on the content and context of the lost packets. We introduce the idea of the *Importance of Individual Packets*, which is defined by the impact of VoIP packet loss on speech quality. We present real-time and off-line algorithms to measure this importance. Using the concept of importance of packets we show that only a fraction of all speech packets needs to be transmitted if speech intelligibility is to be maintained. By applying this concept for Internet telephony over wireless links, significant transmission energy savings on wireless phones can be achieved, because fewer packets need to be transmitted.

At the MAC layer we provided an open-source simulation model of IEEE 802.11e EDCA, which is used in many research projects and is often cited. Also, the ability of WLAN support voice traffic was studied qualitatively and quantitatively.

Last, we proved that the distance between two WLAN nodes can be determined by packet round trip time measurements. This approach outperforms the previously used signal strength indications.

Overall one can summarize, that this thesis contains relevant innovations and novel algorithms, which have a high potential to influence future research and product development.

Zusammenfassung

Diese Dissertation präsentiert innovative Algorithmen, um die Übertragungseffizienz von Telefonaten über drahtlose IP-Verbindungen zu steigern. Die folgenden Ergebnisse wurden erreicht:

Ein Algorithmus wurde entwickelt, der die wahrgenommene Qualität von Internet Telefongesprächen instrumentell bewerten kann. Hierfür kombinieren wir das ITU E-Modell, den ITU PESQ Algorithmus und unterschiedliche Sprachkodierungen und Ausspielpuffer. Teile dieses Algorithmus wurden in die Standards der ITU aufgenommen. Mit Hilfe dieses Bewertungsalgorithmus leiten wir darüber hinaus wichtige Designentscheidungen für Transport- und Vermittlungsschichtprotokolle analytisch her. Auch entwickelten wir ein Qualitätsmodell, das für adaptive VoIP-Applikationen eingesetzt werden kann. Diese Arbeiten wurden mit einem Best-Paper-Award ausgezeichnet.

Wenn hochgradig komprimierte Sprachdaten über das Internet übertragen werden, kann der Verlust eines Paketes sehr unterschiedliche Auswirkungen haben. Dies hängt unter anderem vom Inhalt und Kontext des Sprachpakets ab. Wir definieren *die Paketwichtigkeit als die Verschlechterung der Sprachqualität nach Verlust eines Sprachpakets.* Wir entwickelten Verfahren, welche die Paketwichtigkeiten offline und in Echtzeit quantitativ bestimmen. Wendet man das Konzept der Paketwichtigkeiten an, zeigt sich, dass nur ein Bruchteil aller Sprachpakete übertragen werden muss, wenn zumindest Sprachverständlichkeit gewährleistet werden soll. Wird das Konzept der Paketwichtigkeiten bei drahtlosen Internet-Telefonsystemen angewendet, lässt sich der Energieverbrauch der Funktelefone signifikant reduzieren, da weniger Pakete übertragen werden müssen.

Unsere Arbeiten auf der Vermittlungsschicht beinhalten ein open-source Simulationsmodell des IEEE 802.11e EDCA MAC-Standard sowie qualitative und quantitative Kapazitätsuntersuchungen von VoIP über WLAN. Unser Simulationsmodel wurde in vielen anderen Forschungsprojekten eingesetzt und ist häufig zitiert.

Schlussendlich zeigen wir, dass die Entfernung zwischen zwei Wi-Fi Geräten einfach und genau gemessen werden kann, indem man die Laufzeiten der Pakete bestimmt. Unser Verfahren stellte sich den bisher verwendeten Feldstärkemessungen als überlegen heraus.

Zusammenfassend kann man sagen, dass diese Arbeit neue Algorithmen und relevante Innovationen enthält, die ein hohes Potential haben, zukünftige Forschung und Produktentwicklungen zu beeinflussen.

Acknowledgements

I would like to thank *Professor Adam Wolisz* for his advice during my doctoral program and for supporting my research throughout the lasts years. He always provided inspiring ideas and challenging feedback. Often, he was a key source of motivation and caused me to work hard.

Also, I would like to thank my colleagues. *Dr. Henning Sanneck* wrote his PhD thesis on packet loss recovery and control for VoIP transmissions. His work inspired me a lot. Together with *Professor Holger Karl* I wrote a paper that won a best-paper award. I have to admit, you made the publication perfect! *Professor Günther Schäfer* was a great help in times of hard deadlines. He is somebody you always can rely on. Thanks a lot! My student *Sven Wiethölter* while doing this Studienarbeit has implemented the ns/2 EDCA model, which is cited so often, and helped me to write many publications. *Andre Günther*, it was a pleasure to work with you, listening to your stories and jokes was always fun. It is great to see somebody evolving like you. Together, we wrote two publications. *Iacopo Carreras* from Italy was my first diploma student. He wrote an email saying: “I want to do my master thesis, which should cover something with signal processing, something with programming and something with networking.” One morning I woke up and I had in mind his topic: “Prioritise speech packets over WLAN.” Iacopo wrote an excellent thesis and recommended to me the topic I shall study in my dissertation. *Tuya Dulamsuren-Lalla* has conducted the formal listening-only tests. It is a pleasure to work with you! Also *Berthold Rathke*, *Cornelius Mahlo*, and many more were a great help. Ein großes Dankeschön geht an *Ian Marsh*, KTH Stockholm, der die mühsame Arbeit hatte, das Englisch meiner Dissertation zu verbessern. Ich hätte keinen Besseren finden können, denn er beherrscht ein vorzügliches Englisch und hat darüber hinaus herausragende Fachkenntnisse.

Last but not least, I would like to thank my family members and friends, who have been giving me constant support, encouragement and happiness that was indispensable for me to make this thesis become true. Especially my father, *Dr. Ernst-Ludwig Hoene*, has provided excellent feedback to my research and publications.

Contents

List of Figures

List of Tables

Abbreviations, Acronyms, and Terms

3SQM Single Sided Speech Quality Measure is a new algorithm, developed for non-intrusive voice quality testing [110, 185].

802.11 IEEE 802.11 or Wi-Fi denotes a set of Wireless LAN standards developed by working group 11 of the IEEE LAN/MAN Standards Committee (IEEE 802). The term is also used to refer to the original 802.11 [95], which is now sometimes called ”802.11legacy” [216].

802.11b This IEEE standard uses the unlicensed 2.4 Gigahertz (GHz) band with transmission rates of 1, 2, 5.5, 11 MBit/s [216].

802.11e This IEEE standard defines a set of Quality of Service enhancements for LAN applications, in particular the 802.11 Wi-Fi standard. The standard is considered of critical importance for delay-sensitive applications, such as voice over wireless IP and streaming multimedia [96].

AC Access Category: A label for the common set of enhanced distributed channel access (EDCA) parameters that are used by a QSTA to contend for the channel in order to transmit MSDUs with certain priorities [96].

ACK Acknowledgment: A packet to acknowledge the reception of a previous packet [216].

A/D An Analogue-to-Digital converter (abbreviated ADC, A/D, or A to D) is a device that converts continuous signals to discrete digital numbers [216].

ad hoc network A network composed solely of stations within mutual communication range of each other via the wireless medium. An ad hoc network is typically created in a spontaneous manner. The principal distinguishing characteristic of an ad hoc network is its limited temporal and spatial extent [95].

ADPCM Differential (or Delta) pulse-code modulation (DPCM) encodes the PCM values as differences between the current and the previous value. Adaptive DPCM is a variant of DPCM that varies the size of the quantization step [216].

AP Access Point: Any entity that has station functionality and provides access to the distribution services, via the wireless medium for associated stations [95].

AIFS Arbitration Inter Frame Space [96]. Refer to IFS.

AMR Adaptive Multi-Rate is a lossy audio data compression scheme optimized for speech coding. AMR is adopted as the standard speech codec by 3GPP [216].

Bluetooth is an industrial specification for wireless personal area networks. It provides a protocol to connect and exchange information between electrical devices via a secure, low-cost, globally available short range radio frequency [216].

BSS Basic Service Set is a set of stations controlled by a single coordination function [95].

CBR Constant Bit Rate. Compare with VBR [216].

CELP Code Excited Linear Prediction describes a class of speech coding algorithms [216].

CFB Contention Free Bursting sends multiple small packets as a burst without intermediate contention as soon as the station gains access to the medium. Refer to Section 8.2.4 and [205].

CFP Contention Free Period is a term of the PCF MAC protocol. It is time period when the right to transmit is assigned to stations solely by a PC, allowing frame exchanges to occur between members of the BSS without contention for the wireless medium [96].

CP Contention Period is the time period outside of the CFP in a point-coordinated BSS. In a BSS where there is no PC, this corresponds to the entire time of operation of the BSS [96].

CSMA/CA Carrier Sense Multiple Access with Collision Avoidance is a network control protocol used in 802.11 [216].

codec A combination of encoder and decoder [3]. Refer to encoder and decoder.

coder Refer to encoder.

CW Contention Window [95].

D/A A Digital-to-Analogue converter (DAC or D-to-A) is a device for converting a digital (usually binary) code to an analogue signal (current, voltage or charge) [216].

DECT Digital Enhanced (former European) Cordless Telecommunications is an ETSI standard for digital portable phones, commonly used for domestic or corporate use. DECT can also be used for wireless data transfers [216].

DCF Distributed Coordination Function is a MAC protocol modus of IEEE 802.11, in which each station tries to compete for access [95].

decoder A device for the translation of a signal from a digital representation into an analogue format [3].

DIFS Distributed (Coordination Function) Interframe Space [95]. Refer to IFS.

downlink A unidirectional link from an AP to one or more non-AP stations. [96].

drop One speech frame has been dropped by intention. Refer to loss.

DFT Discrete Fourier Transform is a method of transforming a time domain sequence into a corresponding frequency domain sequence [3].

DTX Discontinuous Transmission is a mechanism, which allows the radio transmitter to be switched off most of the time during speech pauses. Refer to AMR and VAD.

EDCA Enhanced Distributed Channel Access is the prioritized CSMA/CA access mechanism used by QSTAs in a QBSS. This access mechanism is also used by the QAP and operates concurrently with HCCA [96].

EDCF Renamed to EDCA.

E-Model is a computational model that can be used as a transmission planning tool for telecommunication systems (ITU G.107, [105]). Refer to Section 3.1.2.

encoder A device for the translation of a signal into a digital representation [3].

EP Error propagation. Refer to Section 7.2.

FEC Forward Error Correction is a system of error control for data transmission performed by adding redundant data.

FFT Fast Fourier Transform is efficient implementation of the DFT algorithm [3].

frame In thesis this term is often used in the meaning of speech frame.

G.711 is an ITU-T standard for audio compression. It is primarily used in telephony and represents 8 bit compressed PCM samples for signals of narrow-band voice, sampled at the rate of 8000 samples/second. G.711 encoder will create a 64 kbit/s bit stream [216].

G.729 G.729 is an audio data compression algorithm for voice that compresses voice audio in frames of 10 milliseconds. It is mostly used in Voice over IP (VoIP) applications and usually operates at 8 kbit/s [216].

GSM Global System for Mobile Communications is the most popular standard for mobile phones in the world. GSM phones are used by over a billion people across more than 200 countries [216].

HC Hybrid Coordinator is a type of coordinator – defined as part of the QoS facility – that implements the frame exchange sequences and MSDU handling rules defined by the hybrid coordination function. The HC operates during both the CP and CFP. The HC performs bandwidth management including the allocation of TXOPs to QSTAs. The HC is collocated with a QAP [96].

HCCA HCF Controlled Channel Access. The channel access mechanism utilized by the HC to coordinate contention free media use by QSTAs for downlink unicast, uplink and direct link transmissions [96].

HCF Hybrid Coordination Function is a coordination function that combines and enhances aspects of the contention-based and contention-free access methods to provide QoS stations (QSTAs) with prioritized and parameterized QoS access to the wireless medium (WM), while continuing to support non-QoS stations for best-effort transfer. The HCF includes the functionality provided by both EDCA and HCCA [96].

HSDPA High-Speed Downlink Packet Access is a mobile telephony protocol. It is a packet-based data service in W-CDMA downlink with data transmission up to 8-10 MBit/s (and 20 MBit/s for MIMO systems) over a 5 MHz bandwidth [216].

MSDU MAC Service Data Unit. Refer to packet.

Hz The Hertz (symbol Hz) is the SI unit of frequency. It is named in honour of the German physicist Heinrich Rudolf Hertz who made important contributions to science in the field of electromagnetism [216].

IEEE The Institute of Electrical and Electronics Engineers (pronounced as eye-triple-ee) is an international non-profit, professional organization for the advancement of technology related to electricity. It serves as a major publisher of scientific journals and a conferences organizer. It is also a leading developer of industrial standards in a broad range of disciplines [216].

IETF The Internet Engineering Task Force is charged with developing and promoting Internet standards. It is an open, all-volunteer organization, with no formal membership nor membership requirements [216].

IFS Interframe Space [95]. Refer to AIFS, SIFS, DIFS, and PIFS.

impact of loss Refer to importance.

importance The importance of frame losses is the difference between the quality due to coding loss and the quality due to coding loss plus frame losses, times the length of the sample. In this thesis we introduce two mathematical definitions of importance.

Refer to Section 5.4 and Equation 5.1 for its first definition. Refer to Chapter 6 and Equation 6.6 for an enhanced definition.

IP The Internet Protocol is a data-oriented protocol used by source and destination hosts for communicating data across a packet-switched network. Data in an IP inter-network are sent in blocks referred to as packets or datagrams (the terms are basically synonymous in IP). In particular, in IP no setup is needed before a host tries to send packets to a host it has previously not communicated with [216].

ITU The International Telecommunication Union is an international organization established to standardize and regulate international radio and telecommunications. Its main tasks include standardization, allocation of the radio spectrum, and organizing interconnection arrangements between different countries to allow international phone calls [216].

LAN A Local Area Network is a computer network covering a local area, like a home, office or small group of buildings such as a college. Standardization efforts by the IEEE have resulted in the IEEE 802 series of standards. There are now two common technologies for a LAN, Ethernet and WLAN [216].

loss One speech frame can get lost by random processes like wireless link errors. Refer to drop.

loss impact Refer to importance.

MAC Media Access Control is the lower sublayer of the data link layer. The MAC is different for the various physical media (such as Ethernet, token ring, WLAN) [216].

MMX is a trademark. Most IA-32 CPUs implement the MMX instruction set to enhance their multimedia performance.

MNRU Modulated Noise Reference Units is an artificial generated white-noise signal as described in [101]. Usually it is used as a reference distortion for listening-only tests.

MOS Mean Opinion Score describes the perceived quality of a service according to a standardized quality assessment process (e.g. [100, 107]). Often the quality is described by a value, which ranges from 1 (bad) to 5 (excellent). Refer to Section 3.1, MOS-LQO, and MOS-LQS.

MOS-LQO The score is calculated by means of an objective (non-human) model which predicts the subjecting (human) rating results. Objective measurements made using the model given in ITU-T Rec. P.862 state results in terms of MOS-LQO [108].

MOS-LQS A score collected in a laboratory test by calculating the arithmetic mean value of subjective (human) judgments on a 5-point quality scale, as it is defined in ITU-T Rec. P.800. Subjective tests carried out according to ITU-T Rec. P.830 give results in terms of MOS-LQS [108].

ns-2 The Network Simulator version 2 is a discrete event simulator targeted at networking research. Ns-2 provides substantial support for simulation of TCP, routing, and multicast protocols over wired and wireless (local and satellite) networks [204].

packet A packet is the fundamental unit of information carriage in all modern computer networks. A packet consists of a header, which contains the information needed to route the packet from the source to the destination, and a data area, which contains the user's information. A good analogy is to consider a packet to be like a letter; the header is like the envelope, and the data area is whatever the person puts inside the envelope [216].

PC Point Coordinator [95].

PCF Point Coordination Function [95].

PCM Pulse-Code Modulation is a modulation technique. It is a digital representation of an analogue signal where the magnitude of the signal is sampled at uniform, regular intervals [216].

PESQ The Perceptual Evaluation of Speech Quality algorithm predicts human rating behaviour for narrow band speech transmission [107]. Refer to Section 3.1.1.

PF Persistence Factor (outdated term used in EDCA).

PIFS Point (Coordination Function) Inter-Frame Space [95]. Refer to IFS.

pitch period is the inverse of the fundamental frequency of a periodic signal. The pitch period is the smallest repeating unit of a signal. One pitch period thus describes the periodic signal completely. [216].

PLC Packet Loss Concealment is a technique used to mask the effects of lost or discarded packets. PLC algorithms typically extrapolate previous speech samples to generate the lost speech segments.

PSTN The Public Switched Telephone Network describes the world's public circuit-switched telephone networks. Originally a network of fixed-line analogue telephone systems, the PSTN is now almost entirely digital, and includes mobile as well as fixed telephones [216].

QAP QoS Access Point is an AP that supports the QoS facility specified in the IEEE 802.11e standard [96].

QBSS QoS Basic Service Set is a BSS that provides the QoS facility. An infrastructure QBSS contains a QAP [96].

QoS Quality of Service (QoS) refers to the probability of the network meeting a given traffic contract, or in many cases is used informally to refer the probability of a packet passing between two points in the network within a given time [216].

QSTA QoS station: A station that implements the QoS facility [96].

quality model A formula and/or set of instructions for how the obtained quality measures are to be interpreted to draw conclusions about the quality of a software entity, product, process, resource or transmission.

RaDiO The Rate-Distortion Optimized streaming techniques takes into account packet importance and knowledge about the channel to optimize service quality. It is a packet scheduling technique introduced by Chou and Miao [41].

R-D Renamed to RaDiO.

RDTSC In Intel x86 CPUs the ReaD TimeStamp Counter instruction counts the number of CPU cycles [97].

RED Random Early Detection is a queue management algorithm. RED monitors the average queue size and drops packets based on statistical probabilities [216].

R factor The transmission Rating factor describes telephone quality and is calculated by the E-Model. A higher R-factor corresponds to a better telephone quality, 0 being the worst value, 70 the minimal quality of telephone calls ("toll quality"), and 100 the best value.

RFC A Request For Comments document is standard defining Internet protocol and conventions [216].

RSSI Received Signal Strength Indication [95].

RTCP RTP Control Protocol (RTCP) is the sister protocol of RTP [186, 216]. It exchanges state and connection information to describe the parallel RTP connection.

RTP The Real-time Transport Protocol defines a standardized packet format for delivering audio and video over the Internet [187, 216].

sample Two definitions are used: In digital signal processing a sample refers to the value attained from an continuous signal when converted to a discrete signal. In addition, a sample refers to any piece of sound.

SID SIlence Descriptor [2] describes the sound of background noise during periods of silence.

SID frame A frame that conveys information about the acoustic background noise [2].

SIFS Short Inter-Frame Space [95]. Refer to IFS.

SIP Session Initiation Protocol is an IETF proposed standard for setting up sessions between one or more clients. It is the leading signalling protocol for Internet telephony.

speech frame A speech frame describes one unit of data, which is typically used in the encoding, decoding and concealment of speech.

speech quality This term describes the subjective or objective impression of speech transmission in relation to its original. The original can be existent or virtual. For example, PESQ can measure the speech quality to rate it with an MOS value.

STA Station [95].

TC Traffic Category. A label for those MSDUs that have a distinct user priority, as viewed by higher-layer entities, relative to other MSDUs provided for delivery over the same link. Traffic categories are only meaningful to MAC entities that support QoS within the MAC data service [96].

TOS The Type Of Service field in the IP packet header. It is used to for Explicit Congestion Notification (ECN) or for Diff-Serv [216].

TXOP Transmission OPportunity: A time interval when a particular QSTA has the right to initiate frame exchange sequences onto the wireless medium. A TXOP is defined by a starting time and a maximum duration. The TXOP is either obtained by the QSTA by successfully contending for the channel or is assigned by the HC [96].

UDP The User Datagram Protocol is a minimal message-oriented transport layer protocol [168]. Its primary use is to multiplex-demultiplex different connections on the same host.

UMTS Universal Mobile Telecommunications System is one of the third-generation mobile phone technologies [216].

uplink A unidirectional link from a non-AP STA to an AP [96].

VAD Voice Activity Detection or Voice Activity Detector is an algorithm used in speech processing, wherein the presence or absence of human speech is detected from the audio samples. The main uses of VAD are in speech coding and speech recognition. A VAD may not just indicate the presence or absence of speech, but also whether the speech is voiced or unvoiced, sustained or early, etc. [216].

VBR Variable Bit Rate. Compare with CBR.

VoIP Voice over IP, also called IP Telephony and Internet telephony, is the technology that enables voice conversations over the Internet or a dedicated Internet Protocol (IP) network instead of PSTN voice transmission lines [216].

WIFI Wi-Fi (or Wi-fi, WiFi, Wifi, wifi), short for ”Wireless Fidelity”, is a set of standards for wireless local area networks (WLAN) based on the IEEE 802.11 specifications. Certified products can use the official Wi-Fi logo, which indicates that the product is interoperable with any other product showing the logo [216].

WiMAX WiMAX an acronym that stands for Worldwide Interoperability for Microwave Access [216] and is defined by the IEEE 802.16 working group for point-to-multipoint broadband wireless access.

WLAN A Wireless LAN is a wireless local area network that uses radio waves as its carrier. Usually WLAN refers to IEEE 802.11 compliant product but it includes also Hyperlan and Bluetooth among others [216].

1. Introduction

Improving the transmission performance of the Internet is a worthwhile objective. It is especially important in wireless communication networks, because they often have a low capacity, tight energy constrains, and time varying channel qualities. Wireless access is frequently used but it is a bottleneck in current and will likely remain a bottleneck future broadband communication systems.

Wireless access works well for cordless and mobile phones. Millions of telephone calls are conducted over cordless and cellular telecommunication systems every day. The transmission of voice over wireless links is highly optimised. The common wireless systems such as DECT, GSM, and UMTS are highly cost effective and efficient. These technologies are based on substantial research results in the field of communication and signal processing theory and perform best when one application – such as telephony – is transmitted over one channel – such as a wireless link – using a dedicated circuit switched link[1].

Internet allows the joint transport of many different multimedia services such as web, games, video and audio. Multiple applications can be transmitted concurrently. The transmission can take place over multiple links in row and even on multiple routes in parallel. But the Internet, as other packet-switched networks too, cannot be as resource efficient on wireless links because packet-switching comes at the cost of controlling and negotiating the transmission schedule of each packet. Thus, a single IP-based telephone call requires more communication resources than a circuit-switched based call. But due to the statistical multiplexing gain of packet-switched networks and considering the overall system costs, Internet based communication might be cheaper and will be important in future – even for telephony services.

In this thesis, we address the question on how to increase the efficiency of IP-based telephony over wireless links. The goal is to enable the usage of wireless, mobile technologies for Internet based services, including telephone, with an equal or better level of user satisfaction as DECT, GSM and UMTS can already achieve.

[1] This statement is affirmed by theoretical research results considering joint source-channel coding [36, 128], which jointly optimizes the source (e.g. speech) for a single channel (e.g. the wireless link).

1.1. Ideas

This thesis is based on two fundamental ideas and concepts that guided this research throughout the last years.

The first concept is: *Optimise the service quality as perceived by its user.* The rationale behind this statement is straightforward. Most telephony calls are between humans[2]. Thus, optimising the communication according to the requirements of the psychoacoustics of humans is a beneficial goal. That means that each traffic flow containing a conversation shall be optimised with its usage and the perceptual service quality in mind.

The second concept can be summarised as follows: *Assuming that individual packets do not contribute to the perceived quality equality, we have to transmit each packet according to its needs.* Important packets should be transmitted with a higher priority, with a lower loss rate, and/or with a lower delay than packets of lower importance. This concept requires understanding the meaning of the concept of packet importance, knowing the importance of packets and controlling the communication systems on a per-packet basis.

Of course, these concepts come at additional costs. The psychoacoustic optimisation using instrumental assessment algorithms might have a higher computation complexity than IP-oriented optimisation goals. Transmitting each packet according to its need also introduces higher complexity requiring more resources. Thus, in this thesis we also address the trade-off between optimal transmission in respect of service quality and the resources.

1.2. Design scope

The ideas and concepts of this thesis may lead to the development of an entirely new communication technology such as 4G. The development of such revolutionary new systems is desirable and will continue over the next decades. Alternatively, an evolving approach can be chosen which enhances existing systems.

WLAN has not been developed with telephony in mind and its efficiency for this application is not sufficient. There have been many research contributions to enhance telephony on IEEE 802.11 (see Section 8.3). Mostly they proposed to modify standards of the IEEE 802.11 group or RFC recommendations. But changing standards or introducing completely new systems is a difficult and expensive approach. Especially, if one considers the world-wide basis of installed systems, the fact that WLAN users roam globally and Internet telephone calls are conducted globally, too. Thus, the introduction of new systems is an approach, which is unlikely to be successful in the short-term. Modifications covering only the software are much easier to achieve.

[2] Human to machine conversations, such as conversational systems for human/computer interaction [127], provide only a small fraction of all calls. Machine to machine communications, such as dial-up modems, are not relevant, too. Especially, if one considers that IP-based access networks are available.

Instead of new standards or technologies, we suggest another research objective: The quality of the speech communication should be improved under the basic assumption of existing or emerging standards (Telephony, WLAN, and IP). We adapt dynamically protocol parameters, which are not fixed and free to choose, to enhance the call quality. This approach might not achieve the same performance optimisations as a development of entirely new systems; however, it can be deployed easily.

1.3. Market development

The Internet has an enormous impact on our social life, business and communication [14]. It has a high impact as it facilitates communication with friends, companies and government institutions [55]. Already, most working places have Internet connections and it is widely expected, that households in developed countries will have broadband Internet access as often as they are equipped with (mobile) telephones, television, and radios. At the beginning of the 2005, 56% of all Germans have Internet access [154]. Many of them have high speed connections.

Whereas the spectrum of communication technologies will be broad and many existing technologies will continued to be used [51](e.g. PSTN, fax, DVB-T, GSM, IP, ...), the Internet is the only technology with the potential to support the transmissions of video, audio, letters, books, newspapers, emails and web pages over the common IP-based protocol stack.

Already, more and more PSTN telephone lines are being replaced by IP broadband access and Internet telephony applications. The reason for the technology replacement is more economical than technical as the PSTN technology is mature. The efficiency, quality and reliability of PSTN based communication is high, often considered higher than IP based solutions. Using existing IP broadband access for telephony services is straightforward, as its usage is simple and the costs of telephone calls are often lower.

Another market development is the increasing usage of wireless and mobile communications. Many telephones are either mobile or cordless phones. An increasing number of personal computers are notebooks with wireless LAN technologies. Whereas cellular and cordless communication networks are often based on PSTN centric technologies like GSM, UMTS and DECT, IP centric wireless technologies are increasing. Common examples are IEEE 802.11, WiMAX, Bluetooth, GPRS, and HSDPA solutions.

We consider the market of voice over wireless IP – or more specific VoIP over Wi-Fi – as important and growing. This opinion is fostered by many market studies and high industrial interest. During the development of the concepts presented in this thesis, many WLAN companies started to develop Wi-Fi voice products and they showed much interest in the research results.

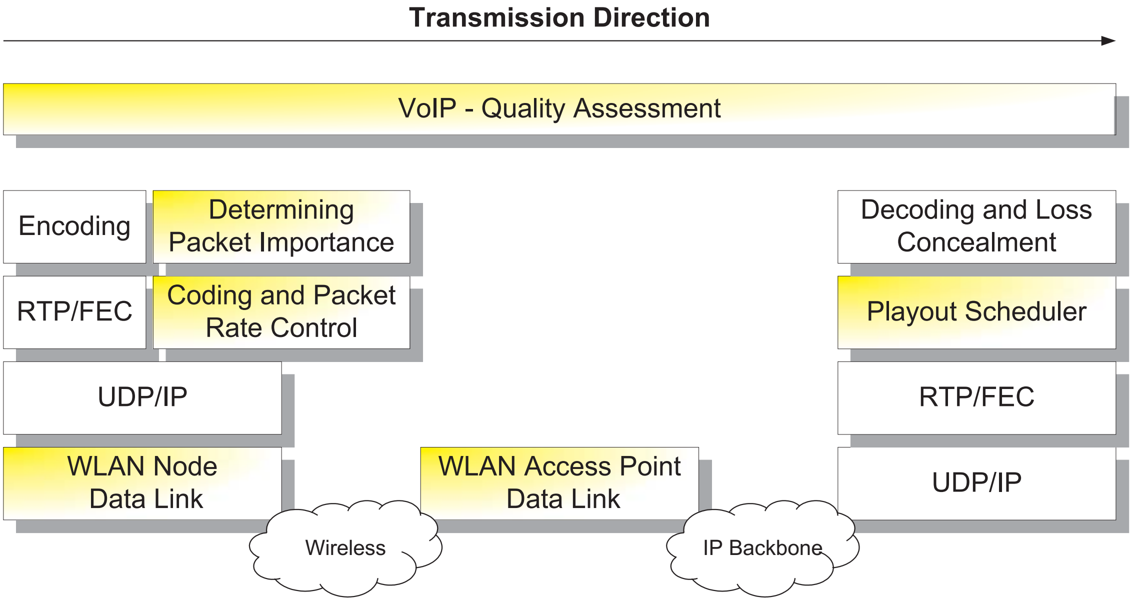

Transmission Direction
VoIP - Quality Assessment
Encoding
Determining Packet Importance
RTP/FEC
Coding and Packet Rate Control
UDP/IP
WLAN Node Data Link
Wireless
WLAN Access Point Data Link
IP Backbone
Decoding and Loss Concealment
Playout Scheduler
RTP/FEC
UDP/IP

Figure 1.1.: An architecture to support wireless Internet telephony is displayed. The highlighted boxes show areas where new results have been developed.

1.4. Architecture of a Wireless-VoIP system

In general, we assume a wireless Internet access scenario used for providing Internet telephony as displayed in Figure 1.1. It consists of a WLAN based wireless access link and an IP backbone, connected via an access point. The sender and receiver contain the classic VoIP protocol stack including encoding, decoding, concealment, playout scheduler, RTP, UDP, IP, and the data link. In the following we briefly described protocol components and algorithms that this thesis introduces or enhances.

An essential part of each optimisation problem is a metric that describes the performance of the system under study. A new optimisation criterion had to be developed because the existing assessment algorithms were not sufficient. An instrumental, algorithmic approach is presented that assesses the perceptual quality of voice transmissions in IP-based communication networks. This approach has been verified with formal listening-only tests.

Using this approach, we derived general control and design guidelines for application and data-link protocols. To be used in real-time to control the transmission parameters for a VoIP flow, we present a new quality model that evaluates VoIP call quality. We apply it in two scenarios: Selecting the ideal coding and packet rate in bandwidth-limited environments and deciding, whether to adapt the playout of speech frames to temporal increased transmission delays.

We introduce a novel algorithm which determines the impact of losing single speech frames.

This thesis contains real-time and off-line algorithms to calculate the importance. Also, it introduces a metric to describe quantitatively the loss impact of speech frames.

On the WLAN data link we measure experimentally IEEE 802.11b link characteristics and determine the distance between two nodes by measuring the packets' round trip times. Using simulation we analyse the capacity of WLAN access networks when used for Internet telephony.

1.5. Structure of this thesis

This thesis is divided into two parts. The first part contains the algorithms to assess call quality and to determine the importance of speech frames. The second part describes application and data-link protocols, which enhance the performance of Internet telephony.

After this introduction the thesis continues with Chapter 2, which gives the technical background that is relevant to both parts of this thesis. Together with a glossary it introduces and defines important terms, abbreviations and concepts. Its aim is to provide the reader with the knowledge to understand the following chapters.

Part one starts with the related work chapter covering relevant and related literature covering the assessment of VoIP and the classification of speech frames. It demonstrates that the questions posed here have not been previously answered. Next, Chapter 4 presents our VoIP quality assessment algorithm, including the results of two formal listening-only tests to verify the algorithm's prediction performance. We studied the impact of losing one single speech frame or VoIP packet in Chapter 5. Also, we show that only a fraction of frames need to be transmitted to maintain good speech quality, if those frames are the important ones. In Chapter 6 we present a linear metric to quantify the importance of a frame, which allow us to calculate the speech quality after multiple frame loss by aggregating the importance of the respective individual frames. Chapter 7 addresses the issue of how to determine the importance of speech frames with low algorithmic delay and complexity. The aim is to use these algorithms in real-time to determine the important packets.

Part two of this thesis starts with a section describing the related work on application, network, and data-link algorithms and protocols that support and optimize VoIP flows. In Chapter 9 we present an application of our VoIP quality assessment algorithm. We used it for a real-time quality model for adaptive VoIP applications. Next, Chapter 10 studies the physical and MAC-layer of WLAN if used for cordless telephony services. Often, a WLAN phone is moved during a call. We explored to what extent slow user motion influences the wireless link quality. We conducted extensive measurements with speech over commercial WLAN equipment using an experimental environment enforcing controlled motion. In Chapter 11 we quantify the capacity and quality of call on WLAN access networks. We studied the

Table 1.1.: Structure of this thesis and list of publications that disseminate the results.

Topic	Chapter	Analysis	Simulation	Experimental
Perceptual Quality Assessment	4	[87, 88]	[90]	[90, 92]
Impact of Single Frame Losses	5	[89, 93]	[93]	[92]
Adding Frame Importance Values	6	[93]	[93]	-
Real Time Classification of Frames	7	-	-	-
Application Layer Control	9	[87, 88]	[138]	[90]
Impact of Slow User Motion	10	-	-	[86]
VoIP Capacity of WLAN	11	-	[213–215]	-
Determining the distance between two WLAN nodes	12	[67, 68]	-	[67, 68]

MAC protocols distribution coordination function (DCF), enhanced DCF and contention free bursting (CFB).

Finally, we show that the distance between two WLAN nodes can be determined if one measures the round trip times of MAC layer packets.

This thesis ends with a conclusion discussing the results of all chapters and an outlook on future research challenges.

Many of the chapters in this thesis are based on publications. Table 1.1 briefly lists the chapters and shows, which publications disseminate the research results. Also, we show which different research methodologies have been applied to achieve the results.

2. Background

2.1. Internet telephony

Internet Telephony allows to offer voice services across networks using Internet protocols [22, 77, 156]. It is an alternative to the classic public switched telephone network (PSTN). IP Telephony consists of signalling and transmission protocols [77, 197]. The signalling protocols (ITU-T H.323 [109] or IETF SIP [181]) establish, control and terminate a telephone call. In this thesis we do not address any signalling aspects.

The principle components of a Voice over IP (VoIP) system, which cover the end-to-end transmission of voice, are displayed in Figure 2.1. First, at the source the analogue processing, digitalization, encoding, packetisation, and protocol processing are performed. Then, the resulting packets are transmitted through the network, comprising of IP networks. At the receiver, protocol entities process the packets and deliver them to the playout scheduler/buffer. In the next step, the speech frames are decoded and played out. Because telephony consists of bidirectional transmission a similar technique is taking place in the opposite direction. In the following, we will discuss the principal components of VoIP systems in detail.

2.1.1. Acoustic processing

The acoustic processing is highly important for the perceptual quality and often neglected in the implementation of VoIP phones [26]. It is required to regulate the level of the input and gain of the output signal in order to guarantee a constant and pleasant loudness of the audio signal by using adaptive gain control (AGC). Another aspect is the presence of background noise, which deteriorates the performance of many encoding algorithms. Therefore, an appropriate background noise suppression has to be implemented so that the human voice of the speaker is filtered out from the acoustic signal [50, 54]. Last not least, often the acoustic output is fed back to the microphone, so that a talker echo is noticeable. Hence a local echo cancellation is required if no headset is used [74]. Often the acoustic processing tasks are implemented in combined algorithms [76].

2.1.2. Speech codecs

Speech coding is employed to reduce the bandwidth of digitized speech signals. All speech coders perform lossy compression. They use fewer bits to represent the speech signal whilst

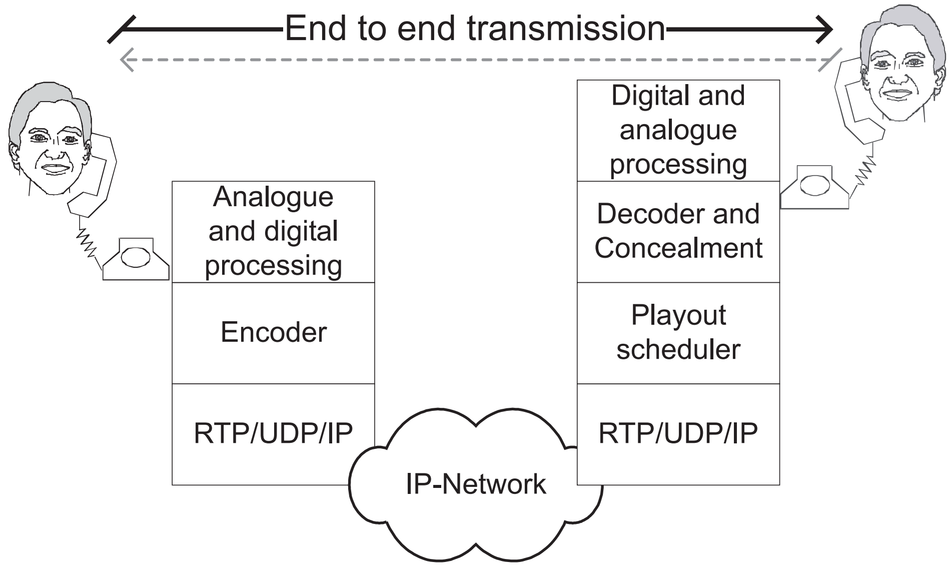


Figure 2.1.: A VoIP telephone call.

maintaining desired level of speech quality [155]. The common, standardized encoding algorithms (G.711, G.723.1, G.726, G.729, GSM, AMR, AMR-WB) differ in their coding rate (bits/s), frame rate (Hz), algorithmic latency (ms), complexity and speech quality (MOS). Table 2.1 gives a brief overview on the codec and their bit rates, applications, and perceptual quality on the MOS scale (refer to Section 3.1).

An important optimization opportunity for speech codecs is the fact that human speech consists of periods of voice activity and silence [31, 42, 113]. Some coding schemes lower the packet rate during silence to send only background noise descriptions (SID). This operating mode is called discontinuous transmission (DTX).

Decoders also contain packet concealment algorithms (PLC) which try to appropriately fill gaps caused by losses. This can be done for example by repeating the previous sound or by extrapolation [32, 44, 47, 160, 196].

In this thesis we have chosen three common speech-coding algorithms namely ITU's G.711, G.729, and ETSI's Adaptive-Multi-rate (AMR), which have been used in all experiments. In the following they are briefly described.

G.711: ITU G.711 [98] is applied for encoding telephone audio signal at a rate of 64 kbps with a sample rate of 8 kHz and 8 bits per sample. G.711 can operate in two modes, A-law (European) and μ-law (US). The μ-law mode is applied in this thesis. The ITU standardized a packet loss concealment (ITU G.711 Appendix I [104]), which limits the impact of transmission losses. The PLC algorithm works on frame sizes of 10 ms.

Table 2.1.: These most common, standardized encoding algorithms and their coding rate, speech quality, and applications.

Codec	**Coding rate [Kb/s]**	**Speech quality [MOS]**	**Applications**
G.723.1	5.3 or 6.8	3.8	Video telephony, VoIP
G.729	8.0	4.0	Mobile telephony, VoIP
G.711	64.0	4.5	Fixed telephone systems
G.726	16, 24, 32, 40		DECT and others
GSM Half Rate	5.6	3.5	GSM / 2.5G networks
GSM EFR	12.2	4.0	GSM / 2.5G networks
GSM Full Rate	13.0	3.5	GSM networks
AMR	4.75 - 12.2	3.5 - 4.0	Third generation mobile networks
AMR-WB	6.6 - 23.85	up to 4.5	Third generation mobile networks

G.729: The G.729 is a hybrid codec, specified by the International Telecommunications Union (ITU), and employs the CS-ACELP (Conjugate Structure Algebraic Code Excited Linear Prediction) algorithm. The speech is partitioned into frames, which frames are 10 ms long, corresponding to 80 bits. The encoder accepts 16-bit linear PCM data sampled at 8 kHz as input data and produces 8 kbps coded data. The codec includes loss concealment at the receiver in order to deal with occasional frame losses.

Adaptive Multi-Rate: The Adaptive Multi-Rate speech codec (AMR) was originally developed for GSM; however it has become the mandatory speech codec for third generation UMTS systems. Similarly to G.729, it provides toll quality and employs CELP-based coding. In addition it allows the dynamic change of bit rates allows so it can adapt to the capacity of the transmission channel. The coding rate can be selected between 4.75 and 12.2 kbps. Respectively, the frame sizes vary between 95 and 244 bits. The frame bits of AMR are ordered into three classes with regard to their significance. Therefore even if some part of the frame is corrupted, other parts may be successfully decoded. In UMTS the bit rate of AMR codec and the wireless channel coding are adapted jointly. A lower AMR bit rate can be transmitted with a higher amount of forward error correction data, enabling the flow to be robust again errors on the wireless link. A higher AMR bit rate can be transmitted with less redundancy, thus improving the speech quality. AMR can be employed not only for UMTS but also for Voice over IP. It has a receiver side loss concealment to cope with packet losses and a sender side voice activity detection to limit the bandwidth during silence.

2.1.3. Transport and network layer protocols

One or multiple speech frames are concatenated in one packet. The Real time Transport Protocol (RTP) [180, 186, 187, 192], User Datagram Protocol (UDP) [168], Internet Protocol (IP) [49, 169]) packet headers are added to the speech segments before the data is sent to the receiver. Optionally, forward error correction (FEC) can be included in the packet. FEC adds redundancy to the transmission so that lost packets can be recovered, as long as sufficient packets are received successfully. Redundancy can be either media-independent or media-dependent. Media-independent FEC is supported in an Internet recommendation [161]. Media-dependent FEC [27,72] uses multiple coding modes to compress the content at different rates (e.g. both G.711 and G.723.1).

2.1.4. IP network/backbone

The network transmits packets from the sender to the receiver. In the Internet, packets can get lost due to congestion, (wireless) transmission errors or connection outages [115]. The transmission delay of packets, the time needed to transmit a packet from the sender to the receiver, is variable and depends on the current network conditions and the routing path [24, 28, 70, 118, 141, 143]. VoIP packets may be transmitted in parallel over multiple paths [133].

2.1.5. Playout scheduler

At the receiver, protocol entities process the packets and deliver them to the de-jittering buffer (also called playout scheduler or packet voice receiver), which temporally stores packets so that they can be played out in a timely manner [11, 13]. They try to playout the speech time-regular manner to conceal variations in the transmission delay also known as jitter.

If packets are too delayed to be played out on time, they are usually treated as lost. Consequently, losses as seen by the application are in fact a superposition of real losses and excessive delays, where “excessive” is used in terms of playout buffer dimensioning. As the playout buffer contributes to the end-to-end delay it should not store packets longer than necessary. Instead, the playout buffer should drop packets that arrive too late to be played out at the scheduled time.

The playout scheduling can be static: If packets exceed a given transmission time they are discarded (we will refer to this scheme as *fixed playout buffer*). Alternatively, *adaptive playout buffers* redefine the playout time in accordance to the delay process of the network [146,173]. We refer to this kind of adaptation as *rescheduling*. The scheduling of playouts can be adjusted most easily during silence periods because then it is not noticeable. Adjustments during voice activity require more sophisticated concealment algorithms [132, 135]. Some early work on the perception of playout jitter can be found in [123, 195].

For each implementation of a multimedia receiver any playout scheduler can be chosen. Playout schedulers are not standardized. Thus, it cannot be predicted a priori how receivers will react to packet losses and delay jitter.

2.2. Adaptive VoIP

Voice over IP applications can adapt the *VoIP configuration* dynamically to the current state of the network in order to enhance the call quality [20,26]. For example, in cases of congestion, it has been proposed to change the coding rate [10, 188, 222]. Thus, the bandwidth of a VoIP flow is lowered and the probability of further packet losses due to congestion is decreased.

Whereas congestion control in general avoids excessive packet losses, it does not avoid packet losses at all. Therefore, an error control scheme should be used [25, 27, 167]. For example, FEC is a good candidate for end-to-end error control of interactive speech transmission. The IETF has standardized an FEC scheme that adds a redundant copy of speech frames to the following packets [161]. If a packet is lost, the receiver reconstructs the lost speech frame after receiving the following frames. Of course, FEC increases both the required bandwidth and the algorithmic delay. Thus, it is beneficial to jointly optimize adaptive FEC and playout scheduling [30, 179].

The number of frames in one packet can be changed to adapt the packet rate and link utilization. For example, the publication [206] has proposed an adaptive packetisation for Voice over WLAN. Alternatively, frame grouping is described in [122] to combine multiple voice flows in a single IP packet.

Numerous publications [129, 132, 135, 146, 165, 173, 193] study how to choose the ideal time of playing out the received frames. The size of the de-jittering buffer should be adjusted so that most packets are not received too late and packet losses are minimized.

Part I.

The Importance of Speech Frames

3. State of the Art

3.1. Quality assessment of telephony

The perceived quality of a service can be measured with subjective tests. Humans evaluate the quality of service according to a standardized quality assessment process [100]. Often the quality is described by a *mean opinion score (MOS)* value, which ranges from 1 (bad) to 5 (excellent). More precisely, values which origin from human test results are called MOS-Listening Quality Subjective (MOS-LQS). In listening-only tests usually speech samples are used, which have a lengths ranging from 6 to 12 s. *Listening-only tests* are time consuming because many subjects have to be asked (refer to Section 4.2 and 4.3).

In the last few years considerable effort has been made to develop instrumental measurement tools, which predict human rating behaviour. We will highlight the published approaches in the following paragraphs.

3.1.1. Perceptual evaluation of speech quality (PESQ)

The *Perceptual Evaluation of Speech Quality (PESQ)* algorithm predicts human rating behaviour for narrow band speech transmission [107]. It compares an original speech fragment with its transmitted and thus degraded version to determine an estimated MOS-LQO (Listening Quality Objective) value. One should note that PESQ can only be applied for distortion types which have been known before its development. These are coding distortions due to waveform codecs and CELP/hybrid codecs, transmission/packet losses, multiple transcoding, environmental noise and variable delay. Benchmark tests of PESQ have yielded an average correlation of R=0.935 with the corresponding MOS values under these conditions. PESQ may have to be changed before it can be applied for low-rate vocoders (below 4 kbps), digital silence, dropped words or sentences, listener echo, and wide-band speech. In the following we will describe essential details of PESQ as they are required to understand the upcoming sections.

Overview: PESQ transforms the original and the degraded signal input to internal representations of a perceptual model. If the degraded signal is not time aligned, e.g., due to jitter or delay, it is first adjusted to the original signal [176]. Next, the perceptual difference between the original signal and the degraded version is calculated [16] considering the human

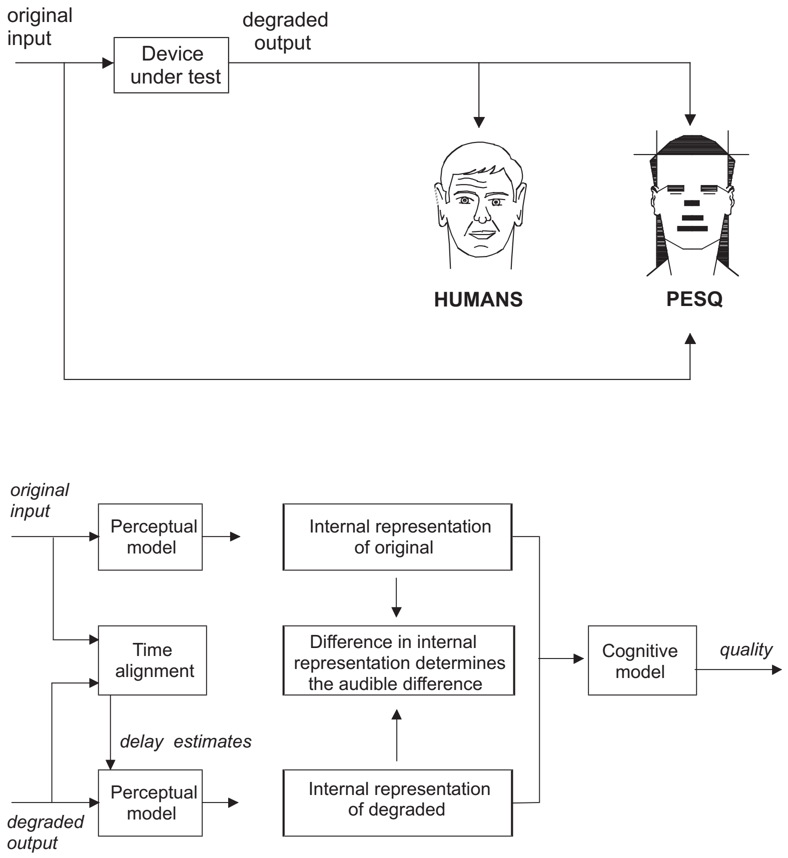


Figure 3.1.: Overview of the basic architecture of PESQ [107].

cognition of speech. Finally, PESQ calculates the perceived speech quality of the degraded signal (see Figure 3.1) .

Computation of the PESQ MOS score: The final MOS score is simply a linear combination of the so called *normal* and *asymmetrical disturbance*. In most cases, the output range will be a MOS-like score between 1.0 and 4.5, the normal range of MOS values found in human subjective experiments:

$$PESQ_{MOS} = 4.5 - 0.1 \cdot D_{indicator} - 0.0309 \cdot A_{indicator} \tag{3.1}$$

$D_{indicator}$ is the normal disturbance and $A_{indicator}$ is the asymmetrical disturbance. Before this final calculation is done the following seven processing steps are conducted:

1. Level alignment: Before the original and disturbed signals can be compared, an adaptation of the power level (loudness) of both signals is conducted.

2. Input filtering: The input is filtered with a frequency envelop that is typical for narrowband telephone networks.

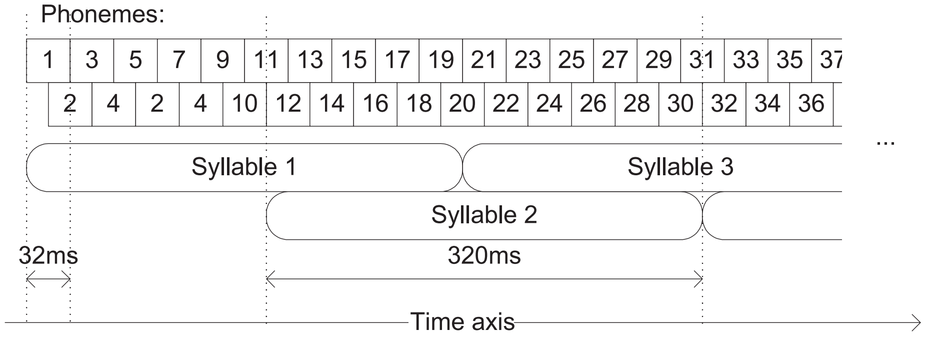


Figure 3.2.: PESQ: Structuring of the speech into phonemes (32 ms) and syllables (320 ms).

3. Time alignment: The time alignment calculates and compensates disturbances caused by transmission delays. This step is important because otherwise the psychoacoustic model would not produce precise results.

4. Auditory transform (psychoacoustic model): The original and degraded signals are transformed into a presentation which models the human hearing property. Here non audible parts of the signal are removed.

5. Time-frequency decomposition: PESQ's perceptual model performs a short term FFT on the speech samples that have been divided into 32 ms *phoneme*s. The phoneme overlap each other with 50% so that each position within the sample is covered by exactly two phonemes. PESQ calculates the spectral difference between original and degraded to calculate the distortion of a given phoneme.

6. Asymmetric effect: If a high correlation between PESQ and the subjective listening-only ratings should be achieved, an asymmetric effect has to be considered [15]: Humans do not know the quality and spectrum of the original speech as they only hear the degraded speech samples. Actually, they compare the degraded speech with an imaginative original, which differs with the original by lacking certain spectrum components. This is caused by the fact that the listener adapts to the constant limitations of the transmitted signal spectrum and considers these limitations as normal.

PESQ models this behaviour and separately calculates two perceptual differences for both the normal and the asymmetric, limited signals. Both disturbances are aggregated separately over time. Finally, they are combined.

7. Weighting of disturbances over time: PESQ uses a two layer hierarchy to group phonemes to *syllables* and to aggregate these syllables over the entire sample length (Figure 3.2). Twenty phoneme disturbances are combined into one syllable distortion with Equation 3.2. Phonemes are aggregated by an exponent of 6 to model a cognitive effect: Even if only one phoneme is distorted, it becomes impossible to recognise the syllable. The authors of PESQ argue that this is a cognitive effect that needs to be considered for high prediction performance [17].

$$syllable_{indicator}^{AorD}\left[i\right]=\sqrt[6]{\frac{1}{20}\sum_{m=1}^{20}phoneme_{disturbance}^{AorD}\left[m+10i\right]^{6}} \tag{3.2}$$

A syllable's length is 320 ms. Similar to phonemes, syllables are also 50% overlapping and cover half of the previous and following syllables. The syllables are aggregated with Equation 3.3 over the entire speech sample. Syllables are aggregated by an exponent of 2 because disturbances occurring during active speech periods are more perceived than those during silence periods [17].

$$AorD_{indicator}=\sqrt{\frac{1}{N}\sum_{n=1}^{N}syllable_{indicator}^{AorD}\left[n\right]^{2}} \tag{3.3}$$

3.1.2. The E-Model

The E-Model (ITU G.107, [105]) is a computational model that can be used as a transmission planning tool for telecommunication systems. A detailed description can be found in [149]. One distinguishing feature of the E-Model is the assumption that the psychological effect of uncorrelated sources of impairment is additive. This assumption is based on the empirical results in the psychophysical research field, which relate magnitudes of physical stimuli to perceptual magnitudes [5].

The E-Model considers the speech quality, the end-to-end delay, echoes, side-tones, loudness and other factors to calculate a so called R-factor. A higher R-factor corresponds to a better telephone quality, 0 being the worst value, 70 the minimal quality for telephone calls ("toll quality"), and 100 is the best value.

The transmission rating factor R is comprised of five terms, which combine different types of impairments. The I-terms below refer to impairment factors.

$$R=R_o-I_s-I_d-I_e+A \tag{3.4}$$

R_o represents the transmission rating of the basic signal-to-noise ratio. Circuit noise, room noise at the sender and receiver, the side-tone, which is the sound of the speaker's own voice as heard in the speaker's telephone receiver, and the noise floor, which is generated by the device itself, are factors that are taken into account. The default value of R_o equals 93.2 [199].

The factor I_s is the sum of all impairments which occur simultaneously with the voice transmission: An overly loud voice signal, quantization (A/D and D/A conversion, logarithmic PCM coding, ADPCM coding) and a non-optimum talker side-tone.

Transmission delay also impairs the quality of a telephone system (see Figure 4.2). The factor I_d represents this delay impairment, which is strongly affected by the talker and listener echoes. If echoes are present, the delay can be more easily noticed.

Whereas the previous I factors cover mainly classic PSTN related quality impairments, I_e takes into account all impairments caused by more complicated equipments. It is mainly used for predicting the coding distortion of low-rate speech codecs. Because the influence of frame losses depends largely on the type of coding and loss concealment, the frame loss rate also influences I_e. The value of I_e can be gathered from subjective auditory tests [106].

The last factor A is based on the knowledge that the quality of a telephone call is judged differently if the user has an advantage of access. For example, the availability of wireless, cellular, and satellite connections is appreciated and users judge their connection quality higher than standard tethered access. Cellular phone users do not expect the same quality level as in PSTN telephone calls. If the Internet access is cheap or even free, VoIP will also have an advantage of access. Typical values of A range from 0 to 20.

3.1.3. Other quality models

Clark [43] identified the effect that humans remember the quality only for a certain time until the older impressions are replaced by recent. More precisely, the impact of a distortion decays exponentially with time. This effect is called "recency effect". He also introduced the notion of packet loss event, which we have adopted.

Mohammed et al. [145] proposed a model to measure speech quality in real-time in order to control the transmission of VoIP. The author uses a neural network that is trained to estimate speech quality. However, transmission delay is not considered.

Takahashi et al. [202] verified the E-Model with subjective conversational tests. Based on he results he proposed an enhanced model which improves the rating performance from $R = 0.763$ to $R = 0.793$. The authors identified further potential areas of improvement.

Humans can judge the quality of a speech sample without even the knowing of the original whereas PESQ requires both the degraded and the original speech sample. Recently, a new psychoacoustic model called 3SQM has been developed and standardized that does not require the original sample. Its rating performance achieves nearly the same level as PESQ [110,185].

Hammer et al. [71] suggest using PESQ to assess the speech quality of a VoIP packet trace. He proposes splitting the trace into overlapping subparts. The benefit of the proposed approach is that different coding schemes and also packet marking algorithms can be judged. Also, FEC or different playout schedulers can be supported.

An approach that also considers interactivity is presented by Sun and Ifeachor [198]. The authors suggest combining the E-Model and PESQ and describe a set of linear equations. The equations approximate the predictions of PESQ and E-Model considering delay, coding, and loss rate.

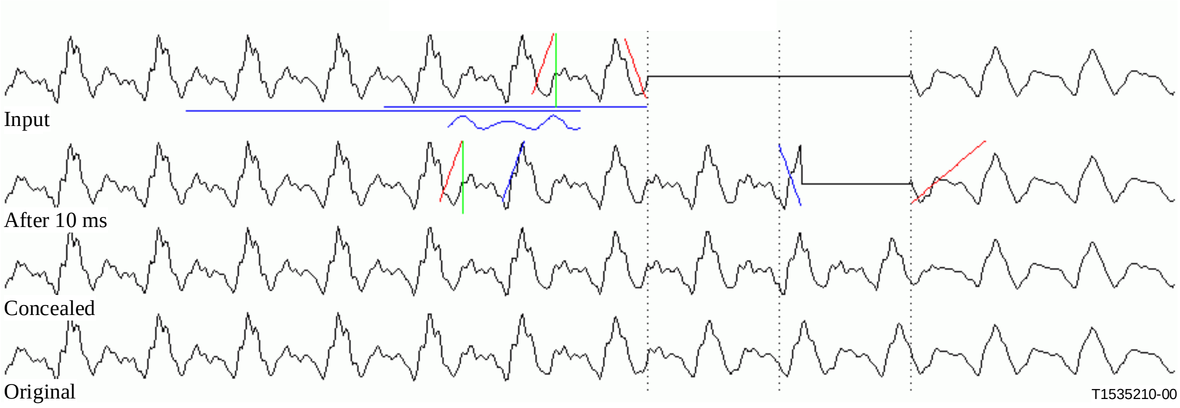


Figure 3.3.: Concealment algorithm in G.711 Appendix I [104].

3.2. A packet loss concealment algorithm

In Section 7.5 we apply the ITU G.711 Appendix I algorithm [104], which is called “A high quality low-complexity algorithm for packet loss concealment with G.711”. This algorithm generates a synthetic speech signal to cover missing frames in a received stream (Figure 3.3). Since speech signals are often locally stationary and do not change much over a short time period, it is possible to use the signal’s past history in order to generate an approximation of the missing frame(s).

Successful received frames: If a frame is received it is decoded. In addition the concealment requires two changes at the receiver. First, a copy of the decoded output signal is stored in a circular history buffer that is 48.75 ms (390 samples) long. This allows synthesizing the concealed signal in the case of frame loss. Second, the output signal is delayed by 3.75 ms (30 samples) before being played. This algorithmic delay, used for an Overlap Add (OLA) at the start of an erasure, is required for a smooth transition between the real and concealed signal.

First lost frame: If a frame is lost, first the pitch period [175] is estimated by finding the peak of the normalized cross-correlation of the most recent 20 ms of speech in the history buffer with the previous speech considering a segment length from 5 (40 samples) to 15 ms (120 samples). This corresponds to frequencies between 66 and 200 Hz.

If the first frame is lost, the concealed speech segment is generated by repeating the last 1.25 pitch periods. The loudness of the concealed segment is not changed. To insure a smooth transition between the real and the synthetic signal and between multiple pitch periods, an Overlap Add (OLA) operation is performed using a triangular window of one fourth of the pitch period, both at the start and the end of the lost frame.

Multiple lost frames: If more than one frame is lost, the synthesized signal contains not only the last pitch period, but also others. Also, the loudness is decreased.

3.3. Classification of speech frames

3.3.1. Voice Activity Detection

Speech frames differ greatly: One of the classic applications of the temporal characteristics of speech is the suppression of packet transmissions during silence. Periods of active speech alternate with periods of background noise (virtually silence). Periods of silence are less important for the perceptual quality of speech transmission. Therefore, the constant flow of frames can be interrupted until relevant audio content has to be transmitted again. Usually, a voice activity detection (VAD) is part of the encoder.

3.3.2. Voicing

Human generates two types of sounds: voiced and unvoiced. Voiced sounds have a regular pattern and usually high energy (e.g. “a”,”o”, . . .). Unvoiced sounds have a random nature (e.g. “h”,”sh”, . . .). The voicing decision usually can be generated by the codec at no additional cost. For example, Both G.729 and AMR decoders use the voicing information to adapt the packet loss concealment to the nature of the previous speech segment.

Petr et al. [162] suggested a method to mark speech frames as background with the lowest priority. The next higher priority is assigned to voiced speech segments, which are not at the beginning of the voiced sounds. The next higher priority is assigned to non-initial fricative (e.g. the “ch” in the German word Bach). All other frames including the initial voiced and fricative speech segments are marked with the highest priority.

3.3.3. Source-Driven Packet Marking

De Martin [48] has proposed an approach called Source-Driven Packet Marking, which controls the priority marking of speech packets in a DiffServ [23] network. If packets are assumed to be perceptually critical, they are transmitted in a premium traffic class. All other packets are sent using the best-effort traffic class.

The author describes a packet-marking algorithm for the ITU G.729 codec. For each frame, it computes the expected perceptual distortion, as if the speech frame were lost, under the assumption that no previous speech frames were lost. First, only speech frames with at least a minimal level of energy are considered to be marked as premium. Next, the marking algorithm takes the coding parameters (e.g. the gain, linear prediction filter, codebook indexes) and computes the parameters that would be computed by the concealment algorithm if the packet was lost. It then compares both parameter sets – the original and the concealed – in order to compute the perceptual quality degradation in case of loss.

If any of the following perceptual distance parameters exceed a given threshold, the packet is marked as premium. Depending on the voice/unvoiced state of the previous frame (as measured at the decoder), the thresholds used for *voiced* frames are:

- Adaptive-codebook index difference > 20%
- Adaptive-codebook gain difference > 5 dB
- Spectral distortion > 4dB

Instead, if the decoder expects a frame, only the fixed-codebook gains and the spectral distortion are used:

- Fixed-codebook gain difference > 5dB
- Spectral distortion > 4 dB

De Martin conducted formal listening tests, which showed that the source-driven packet marking enhances speech quality from MOS 3.4 to 3.7 in case of a loss rate of 5%. For comparison, if no packets are lost, the G.729 codec attains a speech quality of MOS 4.0. It is sufficient for 20% of all packets to be marked as premium.

The source-driven packet marking was enhanced by Masala et al. for video frames [144]. He proposes to compute the importance of multimedia frames. To predict the effect of receiver-side error concealment he uses an analysis-by-synthesis distortion computation.

Petracca et al. [164] presented a classification of AMR frames. His analysis-by-synthesis distortion evaluation calculates the spectral distortion in dB for the LP coefficients, the percentage difference for the long-term prediction coefficients and the difference in dB for the codebook gains. If any of these values is above a given threshold, an AMR frame is marked as premium.

3.3.4. SPB-DiffMark

Sanneck [184] analyzed the temporal sensitivity of VoIP flows if they are encoded with μ-law PCM and G.729: Single losses in PCM flows have a small sensitivity to the current speech properties. Multiple consecutive losses have a higher impact on the quality degradation than single, isolated losses.

The concealment performance of G.729, on the other hand, largely depends on the change of speech properties. If a frame is lost shortly after an unvoiced/voiced transition, the internal state of the decoder might be de-synchronized for up to the next 20 following frames [178]. Furthermore, voiced packets are more important than unvoiced packets. As a consequence, Sanneck proposed to mark packets with +1 (foreground), 0 (best-effort), and –1 (background traffic) depending on their speech properties. After an unvoiced/voiced transition, the so-called Speech Property-Based Differential Packet Marking (SPB-DiffMark) algorithm marks at most the next N packets with +1 and stops the marking with +1, if the packet is classified as unvoiced. All packets, which are not marked with +1, are marked with either with 0 or –1. As long as the number of +1 and –1 marked packets is not equal, packets are marked with –1. Afterwards, they are marked with 0. This is because of fairness requirements.

4. Instrumental Assessment of a Telephone Call's Perceptual Quality

When comparing, designing, implementing and optimizing communication networks for Internet telephony, a thorough approach of assessing the achieved performance is required. Classical approaches mainly focus on evaluating QoS in terms of networking metrics like packet loss, delay, and throughput. However, when it comes to fine-tuning or trade-off decisions are to be made (e.g. delay vs. jitter, or delay vs. packet loss), it is not possible to predict the actual QoS perceived by the user by only focusing on networking metrics. Therefore, we advocate the use of instrumental perceptual assessment methods that closely predict the behaviour of humans rating the quality of multimedia streams.

Perceptual quality assessment has to take into account the complete end-to-end transmission path, as this reflects human-to-human conversation. Thus, when studying the transmission of VoIP packets the entire transmission system has to be considered. Perceived QoS depends on the entire processing chain from source to sink, including encoding, routing across the Internet, de-jittering, decoding and playing the speech at the sink. For example, it is known that the end-to-end quality depends to a large extent on the de-jittering scheme [141].

In Section 3.1 we have described the instrumental perceptual assessment methods PESQ and E-Model. Both models consider most sources of impairment that can occur in a telephone system. For example, they can predict the impact of the mean packet loss rate on speech quality. However, they do not consider packet losses if the loss depends on the packets' content or importance. Also, they cannot be directly applied to traces of VoIP packets, which are produced by experimental measurements or network simulations.

To overcome these deficiencies we have developed a systematic approach that combines the ITU's E-model, the ITU PESQ algorithm, and various codecs and playout schedulers. Our algorithm encodes a speech sample, analyzes a given trace of VoIP packets, simulates multiple playout schedulers, and finally assesses the quality of telephone services (coding distortion, packet loss, transmission delay and playout rescheduling). Thus, it can determine the final packet loss rate, speech quality, mean transmission delay and conversational call quality.

Our approach outperforms previous algorithms as it considers not only the impact of playout rescheduling, but also takes transmission delay, speech quality and non-random packet loss distribution into account. Altogether, we are able to predict the quality of VoIP transmissions to a high precision.

Table 4.1.: Properties and features of quality models.

Features/Impairment	PESQ	E-Model	our approach
coding distortion	yes	yes	yes
mean packet loss rate	yes	yes	yes
non-random packet losses	yes	no	yes
absolute delay	no	yes	yes
delay variations	yes	no	yes
switching the coding mode	yes	no	yes
computational complexity	high	low	high
license-free	no	yes	uses PESQ

This chapter is structured in the following manner: Section 4.1 describes how to combine PESQ, E-Model and playout schedulers. Sections 4.2 and 4.3 contain the results of listening-only tests. Finally Section 4.4 draws conclusions.

4.1. The new quality model

Considering the characteristics of VoIP packet transmissions and the capability of perceptual models, we identify the following aspects as missing or incomplete (see also in Table 4.1).

- Perceptual quality assessment has to take into account the complete end-to-end transmission path, because only this reflects human-to-human conversation. Perceived QoS depends on the entire processing chain from source to sink, including encoding, routing across the Internet, de-jittering, decoding and playing it at the receiving side.
- The end-to-end quality depends largely on the playout buffer scheme [141]. However, an "ideal" playout scheduler has not yet been identified and a VoIP phone implementer can choose any scheme. Thus, to predict the impact of playout scheduling one has to consider all or at least the most common playout schedulers.
- Some playout schedulers change the playout time adaptively during the transmission [63, 136, 195]. This rescheduling deteriorates the speech quality because of temporal discontinuities. The E-Model does not take into account the dynamics of a transmission but relies on static transmission parameters, which do not change during run-time. PESQ considers dynamic playout adaptations, but does not include the absolute delay into its rating algorithm. PESQ has been designed to assess the impact of playout scheduling, but has not been validated for this purpose so far [176].
- Finally, the loss of packets might depend on their content and importance. Thus, one cannot relate the packet loss rate directly to speech quality, because packet losses can

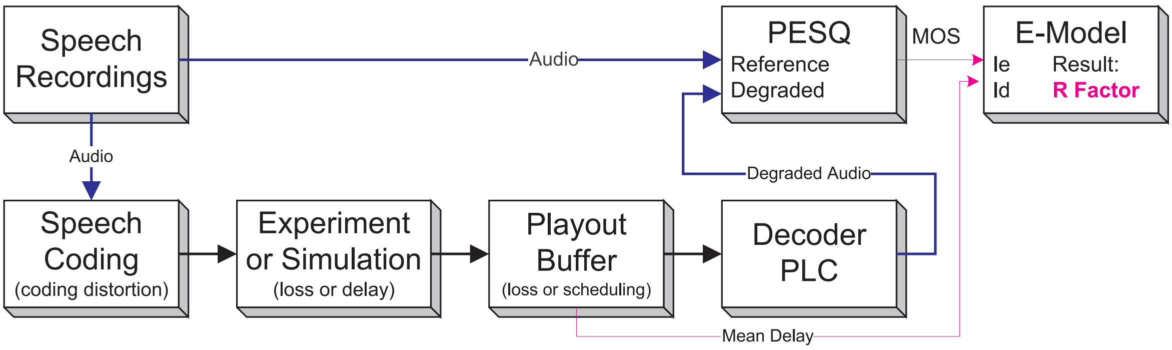


Figure 4.1.: Speech and delay assessment.

occur during silence or periods of low changes in the voice characteristics. The E-Model does not consider non-random packet losses. PESQ has not been verified for this kind of distortion so one cannot know its prediction accuracy.

To overcome these shortcomings we combine the E-Model, PESQ and playout schedulers as shown in Figure 4.1. To judge the quality of a telephone call, we use a similar procedure as for conversational tests. This procedure is described in recommendation ITU-T P.833 [106] and describes how to derive the E-Model's equipment impairment factor I_e from listening-option tests. However, we use fewer test cases and instrumental assessment tools, more precisely PESQ. Each quality judgement is based on multiple observations, which are gathered using the following seven steps, which are repeated several times with different parameterizations.

1. A speech recording is selected from a database. We use the ITU-T P.suppl 23 [102] speech sample database that contains 832 samples from different languages, speakers and sentences. Each sample has a duration of 8 s. Additional background noise is not present.

2. An ITU reference implementation of speech coder compresses the speech samples (currently G.711 and G.729 are supported).

3. The compressed speech frames and the VoIP packet traces are combined. For each speech frame the packet trace contains the information whether it is lost or how long is has been delayed during transmission. If the speech sample should not be used at the very beginning of the packet trace and the packet trace has a longer duration than the speech sample, the sample can be delayed. Also, an additional, constant *system delay* can be added to the transmission delays to simulate the algorithmic delay caused by the processing of the speech signal, codec and OS handling.

4. A set of the most common playout scheduler schemes (including fixed-deadline and adaptive algorithms [146, 173] of Van Jacobsen, Mills, Schulzrinne, Ramjee, and Moon)

calculate the packets' playout times. Also, the mean transmission delay is calculated. One should note that only speech frames during voice activity are considered, as during silence periods a human cannot identify the transmission delay.[1]

5. A speech decoder generates a degraded version of the speech sample and conceals lost frames or gaps due to network losses and due to playout delays.

6. Next, PESQ calculates the speech quality that depends on coding distortion, non-random packet loss and playout rescheduling. Because PESQ has not been verified for non-random packet loss and playout rescheduling, we conduct formal listening tests to verify its accuracy (see Section 4.2 and 4.3)

7. Last, both the speech quality and the mean transmission delay are fed into the E-Model to calculate the observations' R-Factors.

The distribution of R-Factors for a given packet trace is the resulting conversational quality, which can be used to judge the quality of the telephone system.

4.1.1. Combining PESQ and the E-Model

In the following we describe a formula that calculates the overall R-Factor. It is based on the E-Model. If the acoustic processing is optimal, which we assume [87], we can simplify the E-Model to a model with only few parameters. The computation of R_{factor} is then given by:

$$R_{factor} = \mathrm{MOStoR}\left(\mathrm{MOS}_{\mathrm{PESQ}}\right) - I_d\left(t\right) \tag{4.1}$$

If neither talker nor listener echoes are present, the delay impairment I_d [105] can be reduced to the term I_{dd}: For an end-to-end delay $0 < T_a \leq 100\,\mathrm{ms}$, I_{dd} is 0. For any delay between $100\,\mathrm{ms} \leq T_a < 500\,\mathrm{ms}$ $I_{dd}$is

$$I_{dd}\left(T_a\right) = 25\left(\left(1 + X^6\right)^{\frac{1}{6}} - 3\left(1 + \left(\frac{X}{3}\right)^6\right)^{\frac{1}{6}} + 2\right) \tag{4.2}$$

with $X = \frac{-2+\lg T_a}{\lg 2}$ (refer to Figure 4.2).

The mean opinion score can be obtained from the R Factor with a conversion formula given in ITU G.107. For $6.5 \approx 80 - 30\sqrt{6} < R < 100$, this conversion formula can be inverted:

$$\mathrm{MOStoR}\left(m\right) = \frac{20}{3}\left(8 - \sqrt{226}\cos\left(h + \frac{\pi}{3}\right)\right) \tag{4.3}$$

[1] Indeed, some playout schedulers change the playout time at the start of a talk spurt. Others change it at the beginning of silence periods. Both have to be considered as equal with respect to the transmission delay.

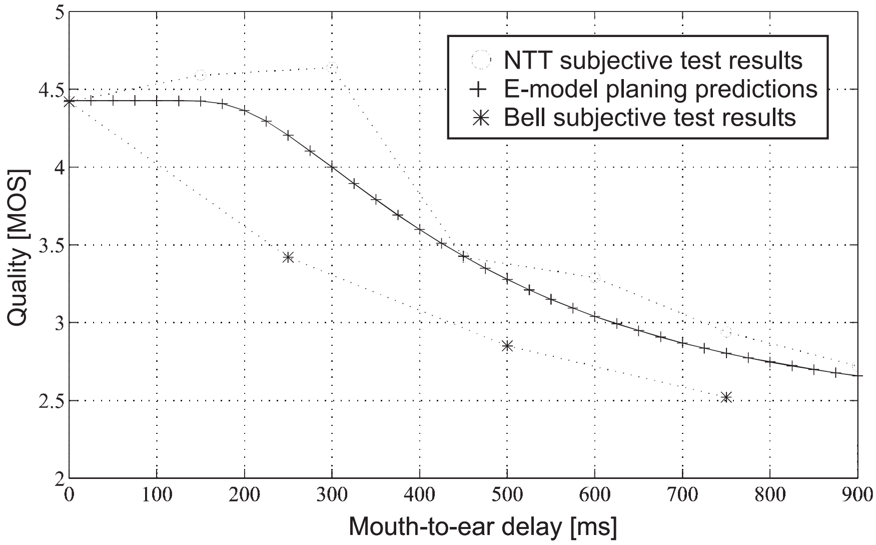


Figure 4.2.: Impact of delay on call quality displaying the E-model and human test results [151].

with

$$h = \frac{1}{3} \arctan 2 \left(\mathrm{x} = 18566 - 6750m, \mathrm{y} = 15\sqrt{-202500m^2 + 1113960m - 903522} \right)$$

4.1.2. Software package

We implemented the approach explained and provide it as open-source to the research community. The software implements the digital processing chain of VoIP. It supports the

- Quality assessment using ITU's E-Model and PESQ.
- Encoding of audio samples with G.711 μ-law and G.729.
- Decoding of audio files, including the rescheduling of the playout time.
- Playout scheduling as described by Moon, Ramjee and others.
- Parsing of packet trace files generated by Snuffle [80] or ns-2 [204] to get data about a packet's delay or loss.
- Generating packet trace files artificially to study the impact of delay spikes.

To be fully operational, the PESQ algorithm and a G.729 codec had to be bought from its owners (e.g. Opticom GmbH). Alternatively, they can be downloaded at no cost from ITU's web page, however for trials only. Further information can be found in the manual [91].

We verified the correctness of our software by several means. The publishing of this software including its source code ensures that more users will use it and study the code. Thus, the pace of finding potential errors will be increased. We have tested our toolset in various projects which include the assessment of voice over WLAN, the impact of handover and wireless link scheduling. Overall, we are confident on the quality of our implementation.

4.2. Verifying PESQ regarding single frame losses

PESQ is only a psychoacoustic model of the human hearing. Thus, it only simulates the human rating behaviour and it is in principle less precise than humans. On the other side, when humans rate the speech quality in listening-only tests, the results are precise only if the tests are carefully conducted. The ITU has set up a detailed description [100] of how to conduct listening-only tests in such a manner that they achieve the highest degree of accuracy. This procedure is referred to as a formal testing procedure. In the following description the results of formal listening tests are presented which verify the prediction performance of PESQ in the presence of a particular kind of distortion, namely single frame losses and non-random packet loss.

PESQ has been designed to consider the impairment due to multiple frame losses. Frame (or packet) losses occur if networks are congested or (wireless) links have transmission errors. The impact of frame losses is well measured by PESQ. It shows a high correlation with the results of formal tests (R=0.93) [159]. However this assumes that the frame losses are randomly distributed. This statement does not hold if single, specific frame losses are to be measured.

In [89] we have shown that objective quality models (such as EMBSD [221] and PESQ) score the same single frame losses differently. Thus, we need to verify whether PESQ measures the importance of single frame losses similarly as humans do. This verification is important because PESQ has not been designed for this kind of measurement and operates outside the scope of its operational specification.

The difficulty of the listening-only tests is the fact that humans often cannot hear the impairment of one frame loss. Humans can judge only the impact of multiple frame losses. Thus, if we want to verify PESQ's rating of single frames, we have to construct samples containing multiple losses of the same frame. However, it is not possible to generate samples which contain multiple losses of the same frame, because at least the frame's context will be different. Thus, we drop multiple, similar frames. If both PESQ and human tests yield the same results for multiple but similar frame losses, PESQ is verified single losses.

As long as frame losses do not occur shortly one after the other, we can assume that PESQ results scale linear with the number of lost frames (refer to Chapter 6). Thus, the loss impact of one frame is a N_{th} fraction of the loss impact of N multiple frames, as long as the frames are similar.

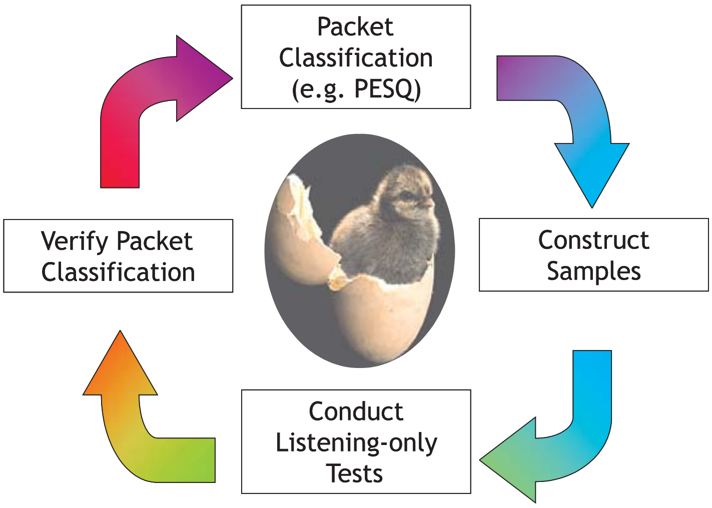

Packet Classification (e.g. PESQ)
Construct Samples
Conduct Listening-only Tests
Verify Packet Classification

Figure 4.3.: Circumstantial evidence is required to judge the quality of packet classification algorithms.

To identify similar frames, a packet classification method is required. Thus, to verify PESQ's ability to classify frame losses, we need a proper classification of frames. This circular problem definition makes verifications difficult (Figure 4.3). We have decided to classify frames according to their importance, as measured with PESQ, and to their different speech properties, (silence, active, voiced and unvoiced sounds). We also vary the coding scheme.

4.2.1. Experimental design

To verify PESQ, we construct artificially degraded samples and conduct both subjective and objective listening-only tests. Figure 4.4 displays the testing procedure.

4.2.2. Mongolia

The tool "Mongolia" (Figure 4.5) helps us generate degraded samples.[2] It works as follows: First, a reference sample is selected from ITU's database P.suppl 23. If requested, samples (and their degraded versions) can be played loudly. Next, a coding algorithm compresses the reference sample and PESQ calculates the degraded sample's MOS value. The tool supports three coding modes: G.711, G.729 and AMR. Next, the overall frame loss rate controls, how many frames are dropped. The packet length controls the burstiness of the frame losses. The later effect refers to the packetized transmission of speech as a VoIP packet can contain multiple speech frames. A random seed value controls the positions of these losses. The user can select whether important or less important frames are to be dropped. The *importance*

[2] The tool can be tested remotely on our web page [83].

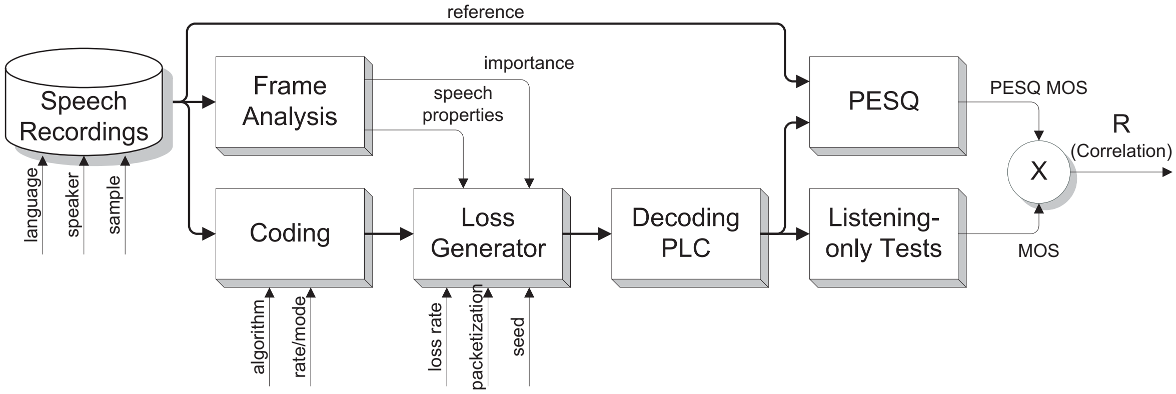


Figure 4.4.: Test design.

of a frame is the quality degradation that the frame's loss would induce. In Chapter 5 we describe in detail how the importance of a packet is calculated. High values refer to more important frames. Next, frames are selected according to their speech property:

- frames during silence (off)
- active voice (on)
- active frames containing unvoiced sounds (voiced)
- active frames containing voiced sounds (unvoiced)

Last, the packet loss statistics and the PESQ MOS value are displayed.

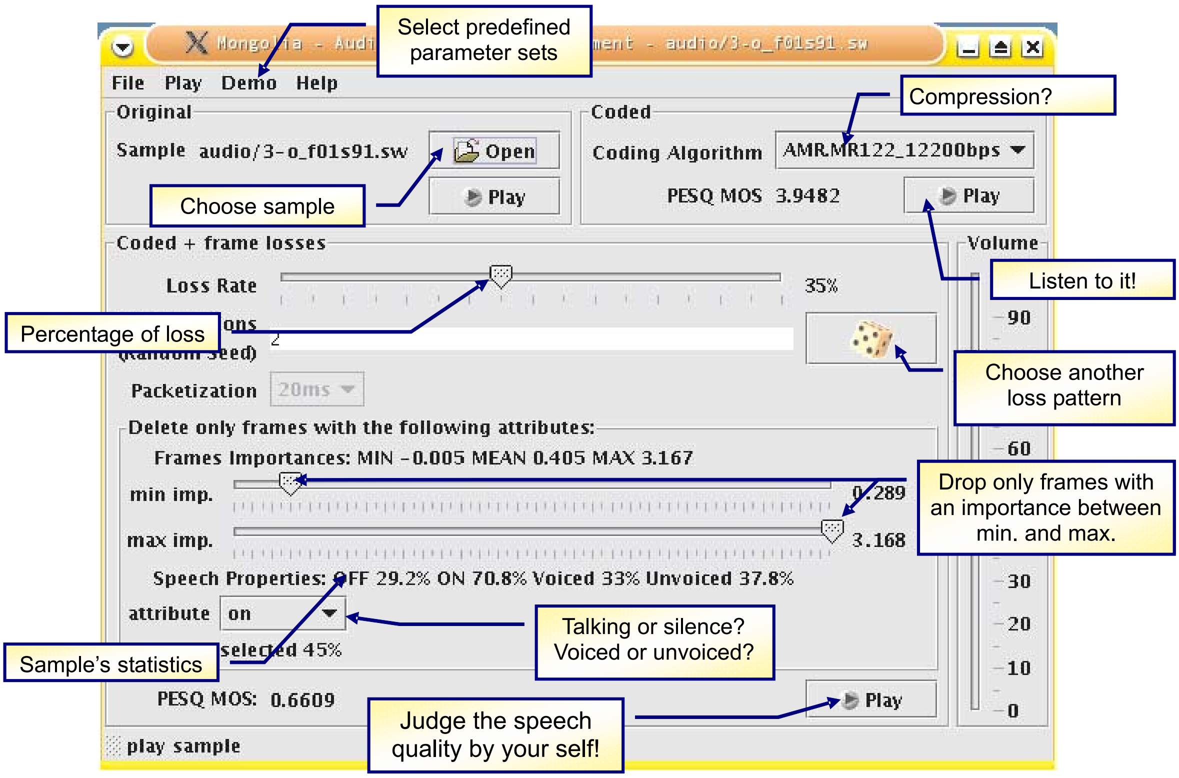
4.2.3. Test sample design

For our listening-only tests we construct samples from four English language speakers (2 male and 2 female subjects). We drop 3% of all frames but dropping only active frames. We do not analyse the trivial case of dropping silent frames.

We select four coding modes (G.711, G.729, AMR 4.75, and AMR 12.2) and choose the shortest packet length (10 ms for G.711 and G.729; 20 ms for AMR). We force the loss of either all, voiced or unvoiced segments. We also drop frames from either all, the most or the least important half of the frames. Altogether this test design consists of 4·4·3·3=144 test samples. As a reference we also generate 20 samples containing modulated noise reference units (MNRU) as described in [101].

4.2.4. Formal listening-only tests

Our listening-only tests took place in a professional sound studio (46 m^2, low environmental noise, etc.). Nine persons judged the quality of 164 samples. The samples' language is English,


Mongolia - Audi ment - audio/3-o_f01s91.sw
File
Play
Demo
Help
Original
Open
Play
Coded
Play
Coded + frame losses
Volume
90
60
30
20
10
0
Delete only frames with the following attributes:
Play

Sample audio/3-o_f01s91.sw

Coding Algorithm AMR.MR122_12200bps

PESQ MOS 3.9482

Loss Rate 35%

ons (Random seed) 2

Packetization 20ms

Frames Importances: MIN -0.005 MEAN 0.405 MAX 3.167

min imp. 0.289

max imp. 3.168

Speech Properties: OFF 29.2% ON 70.8% Voiced 33% Unvoiced 37.8%

attribute on

selected 45%

PESQ MOS: 0.6609

play sample

Select predefined parameter sets

Compression?

Choose sample

Listen to it!

Percentage of loss

Choose another loss pattern

Drop only frames with an importance between min. and max.

Talking or silence? Voiced or unvoiced?

Sample's statistics

Judge the speech quality by your self!

Figure 4.5.: Design tool Mongolia [83].

Table 4.2.: MOS results for modulated noise (MNRU).

MNRU	MOS-LQS	Norm. MOS-LQS	PESQ MOS-LQO	MNRU	MOS-LQS [132]
5	1.12	1.43	1.44	10	1.4
15	1.75	2.20	2.23	18	2.7
25	2.52	3.14	3.08	24	3.7
35	3.23	4.01	3.95	30	4.1
45	3.58	4.43	4.50	none	4.4

which all listeners understand.

The listening-only tests closely followed the ITU recommendation P.800 Appendix B [100] that describes methods for subjective assessment of quality. We did not follow the ITU recommendation if scientists have suggested modifications that improve the rating performance. For example, we used high quality studio headphones instead of an Intermediate Reference System, because headphones produce a better sound quality. Also, multiple persons have been in the room at the same time to reduce the duration of the experiment. We did not apply the "Absolute Category Rating" (1,2,3,4, and 5) because a discrete MOS scale makes it difficult to compare two only slightly different samples. We allowed intermediate values (e.g. 3.5 and 2.5) and used a linear MOS-LQS scale. PESQ can calculate a MOS-LQO value with a resolution of up to 10^{-6} at the MOS scale, too.

Finally, we analysed the results. We calculated the correlation of subjective and objective listening-only rating results to obtain a measure for the similarity (R). If R equals one, the results are perfectly related. If no correlation is present, R equals zero. When comparing absolute subjective and objective MOS values, we applied a linear regression to one set of values. The correlation R does not change after linear regression.

4.2.5. Results

First, we show the MNRU listening-only results. In Figure 4.6 and Table 4.2 we present MOS values from PESQ (labelled as "PESQ MOS"), our listening-only tests ("MOS") and from the tests described in [132]. We also included MOS-LQS values after linear regression ("scaled MOS"), which fit closely the PESQ MOS-LQO values. Subjective and objective results have a correlation of R=0.999 and thus match perfectly.

Next, we show the MOS ratings excluding the MNRU results. We calculate the mean scores over all listener ratings (9) and over all different reference samples (4, thus $4 \cdot 9 = 36$ trials). Table 4.4 contains the resulting MOS values. In Figure 4.7 we display PESQ MOS-LQO vs. MOS-LQS to get an impression of the measurement performances.

We analyse the prediction performance for difference kinds of impairment and partitioned

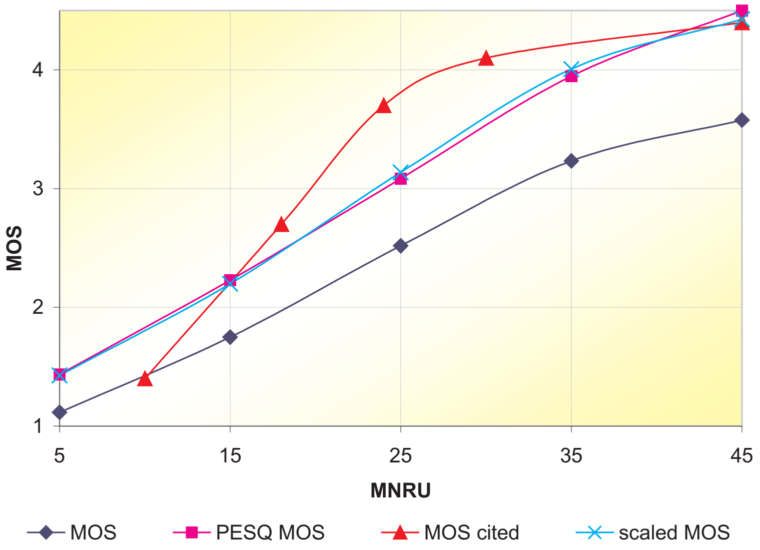


Figure 4.6.: Comparison of noise (MNRU) ratings from listening-only tests.
Human (MOS) and objective results (PESQ MOS) have a correlation coefficient of R=0.999 and are – after linear regression (scaled MOS) – nearly identical.

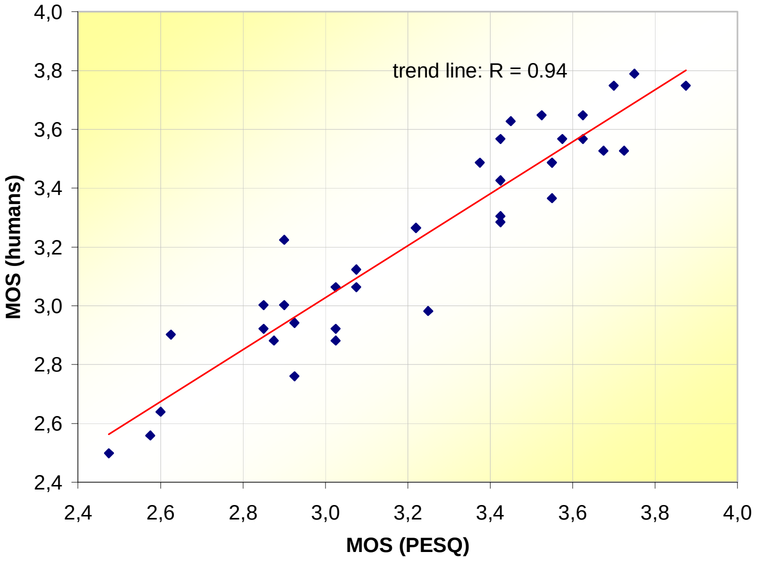


Figure 4.7.: Single packet loss case: Comparison of MOS-LQS and PESQ MOS-LQO.

Table 4.3.: Single packet loss: Listening-only test results.

Importance	Speech	Codec	MOS-LQS	Norm.	PESQ	MOS-LQO minus
	Properties			MOS-LQS	MOS-LQO	Norm.MOS-LQS
Min 50%	Voiced	AMR 12.2	3.387	3.366	3.550	**0.2**
All			3.022	3.124	3.075	0.0
Max 50%			2.656	2.882	2.875	0.0
Min 50%	Voiced	AMR 4.75	2.473	2.761	2.925	**0.2**
All			2.169	2.559	2.575	0.0
Max 50%			2.077	2.498	2.475	0.0
Min 50%	Voiced	G.711	3.814	3.648	3.525	-0.1
All			3.784	3.628	3.450	-**0.2**
Max 50%			3.692	3.567	3.575	0.0
Min 50%	Voiced	G.729	3.266	3.285	3.425	0.1
All			2.809	2.982	3.250	**0.3**
Max 50%			2.656	2.882	3.025	0.1
Min 50%	Unvoiced	AMR 12.2	3.631	3.527	3.725	0.2
All			3.570	3.487	3.375	-0.1
Max 50%			2.930	3.063	3.025	0.0
Min 50%	Unvoiced	AMR 4.75	2.930	3.063	3.075	0.0
All			2.839	3.003	2.850	-**0.2**
Max 50%			2.687	2.902	2.625	-**0.3**
Min 50%	Unvoiced	G.711	3.966	3.749	3.875	0.1
All			4.027	3.789	3.750	0.0
Max 50%			3.692	3.567	3.625	0.1
Min 50%	Unvoiced	G.729	3.570	3.487	3.550	0.1
All			3.479	3.426	3.425	0.0
Max 50%			3.174	3.224	2.900	-**0.3**
Min 50%	All	AMR 12.2	3.631	3.527	3.675	0.1
All			3.296	3.305	3.425	0.1
Max 50%			2.839	3.003	2.900	-0.1
Min 50%	All	AMR 4.75	2.717	2.922	3.025	0.1
All			2.717	2.922	2.850	-0.1
Max 50%			2.291	2.639	2.600	0.0
Min 50%	All	G.711	3.966	3.749	3.700	0.0
All			3.814	3.648	3.625	0.0
Max 50%			3.692	3.567	3.425	-0.1
Min 50%	All	G.729	3.570	3.487	3.550	0.1
All			3.235	3.265	3.220	0.0
Max 50%			2.748	2.942	2.925	0.0

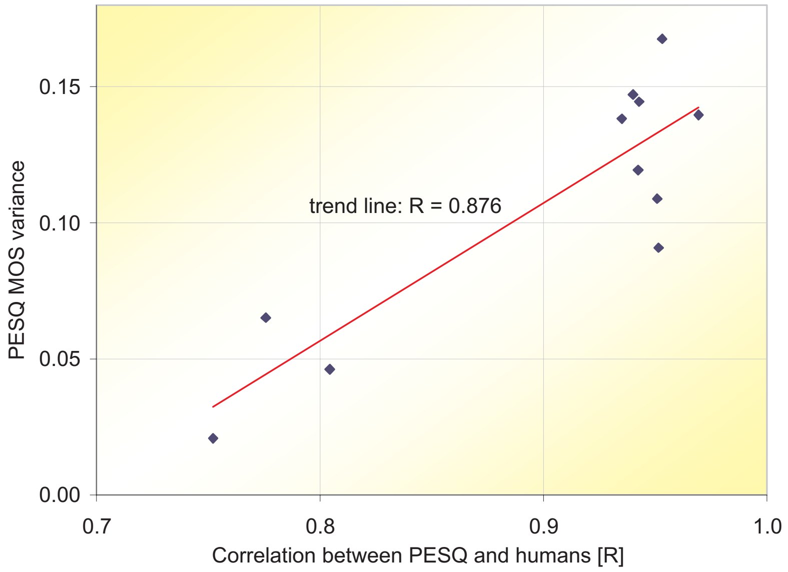


Figure 4.8.: Sample set variance vs. prediction performance.

all speech samples in different sample sets (Table 4.4 contains the correlation between MOS-LQS and MOS-LQO values for different sample sets). In general the correlation depends on the variation of a sample set (see Figure 4.8). If the samples in one set are largely different, both humans and PESQ rate the speech quality similarly and R is high. For example, PESQ predicts only partly the impact of packet losses considering only those samples, which are encoded with the same codec. The prediction performance for G.711, G.729, and AMR 4.75 is rather low. Those sample sets differ only slightly and their variance is low.

This dependence between sample set variance and prediction accuracy can be explained by "measurement noise" being present in subjective tests: Humans can judge highly different samples more precisely than samples with only minor variations (see Figure 4.11).

Overall, these experiments show that human ratings correlate with PESQ at a degree of R=0.94 (see Table 4.4). Thus, given the precision of speech quality measurements, we can assume equal quality of the subjective and instrumental ratings.

4.2.6. Analysis

A degraded speech sample might contain different sources of impairment such as frame loss, coding distortion, or random noise. If such sources need to be compared, PESQ might not assess them precisely in relation to each other. Thus, trade-offs like loss rate versus coding rate might be imprecise. In addition, informal listening-tests show that PESQ might not judge the effect of clipping – shortly before an ON-OFF transition – precisely. Further studies are required to identify problematic packet loss patterns (e.g. as done in [18]).

Table 4.4.: Single packet loss: Accuracy of PESQ.

Condition	Correlation (R)	Number of trials	Mean MOS-LQS	Mean norm. MOS	Mean PESQ MOS-LQO	PESQ MOS variance
All but MNRU	**0.940**	1296	3.189	3.235	3.235	0.147
MNRU	**0.999**	180	2.439	2.738	3.039	NA
AMR 12.2	0.951	324	3.218	3.254	3.292	0.109
AMR 4.75	0.804	324	2.545	2.808	2.778	0.046
G.711	0.752	324	3.828	3.657	3.617	0.021
G.729	0.776	324	3.167	3.220	3.252	0.065
Both voiced and unvoiced	0.969	432	3.210	3.248	3.243	0.140
Voiced	0.943	432	2.984	3.098	3.144	0.145
Unvoiced	0.953	432	3.375	3.357	3.317	0.168
Importance All	0.942	432	3.230	3.261	3.239	0.119
Importance upper half	0.935	432	2.928	3.061	2.998	0.138
Importance lower half	0.951	432	3.410	3.381	3.467	0.091

4.3. Verifying PESQ regarding playout rescheduling

In the following we verify whether PESQ can measure the impairment of playout rescheduling, the reason for this is because PESQ has not been designed for this kind of impairment and operates outside the scope of its operational specification [176]. Again, to verify PESQ, we construct artificially degraded samples and conduct both listening-only tests and instrumental predictions.

4.3.1. Experimental setting

Analyses of Internet traces have shown that packet delays can contain a sharp, spike-like increase [141, 173], which cannot be predicted. A delay spike is a short increase of the packet transmission times, which usually occur after network congestion or on a wireless link that suffers from fading. After the spike's gap the following packets arrive shortly one after the other until the transmission delay has returned to normal (Figure 4.9). We consider the question whether to adjust the playout of speech frames to deal with delay spikes (refer also to Figure 9.7). We concentrate on the non-trivial case of delay spikes during voice activity.

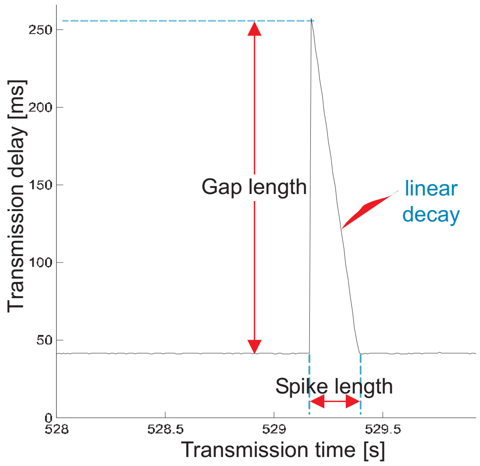


Figure 4.9.: Delay spikes are short, spike-like increases in transmission delays.

4.3.2. Speech sample design

For the test sample design we used software package described in Section 4.1.2. As well as judging the quality of VoIP packet trace, the software can generate artificial packet traces that contain delay spikes. The frequency, the height and width of delay spike can be controlled by changing the respective number of spikes, the maximal packet delay, and the time to return to normal.

In this section, three different playout strategies are analysed: First, we drop every packet that is affected by the spike. Second, the playout is re-scheduled so that packets are not lost. The playout time only increases. It never decreases. The third strategy is similar to the second, but any delayed playout schedule is adjusted during every silent period until the playout time returns to normal.

We construct 220 samples (length approx. 5-10 s), containing samples encoded with G.711, G.729 and containing one delay spike with a gap length of 50 to 300 ms and a spike length of 55 to 330 ms ($spike.length = 1.1 \cdot gap.length$). Again, the tests took place in a professional sound studio. Ten persons judged the quality of 221 samples. The samples' language was German, which all listeners understand.

4.3.3. Results of delay spike listening-only tests

The test subjects gave a total of 2210 judgements. We could use only 2033 judgments because the test people failed to keep track with voting rows. The rating performance during the second half of the test was significantly worse than during the first half because the persons got tired. We also compared a group of native speakers with a group of foreign students. Both showed a similar rating performance. This leads us to the conjecture that being concentrated is more important than being a native speaker.

Figure 4.10 displays the speech quality versus the spike height. We display the rating results of humans and PESQ for different adaptation policies. The black lines (drop) refer to the dropping of any late packet during the delay spike. The blue lines (adapt) display the results when delaying the playout following the delay spike. Last, the red lines (adapt&fallback) include the effect of falling back to the original playout time as soon as possible. The later rescheduling occurs only during periods of silence.

4.3.4. Analysis of delay spike listening-only tests

In Figure 4.10 the observations of different spike lengths are based on different sentences. The sample content has a large influence on the speech quality ratings. Thus, one cannot compare the MOS values across the horizontal axis.

The playout strategies are compared using the same set of sentences. Thus, the playout strategies can be compared against each other as long as the spike length remains the same:

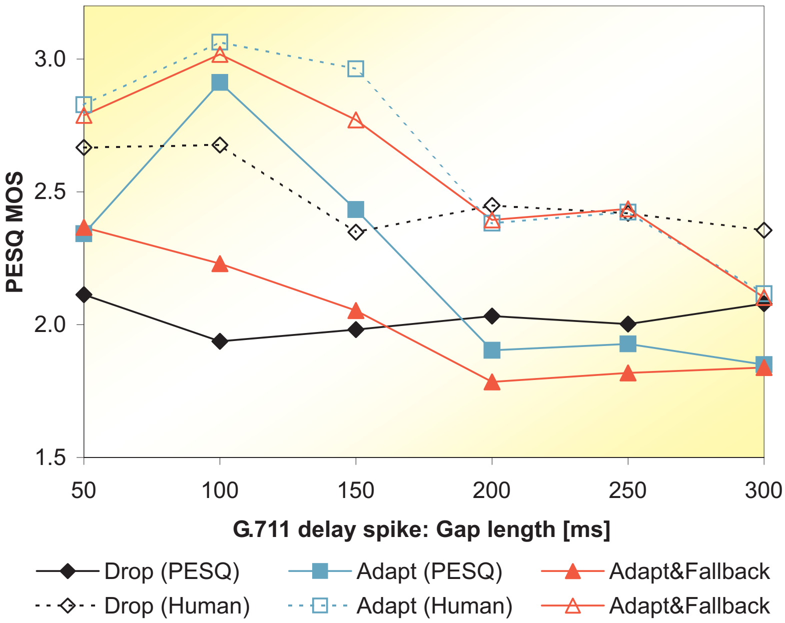
1. If the delay-spike's height is 200 ms or larger dropping of packets is more beneficial than delaying the playout.
2. The “adapt&fallback” ratings are worse than “adapt” rating results. Thus, the negative adjustments during silence periods lower the speech quality.

If one compares absolute MOS values' in Table 4.5, one can see a constant offset between instrumental and subjective MOS values. We assume that it is due to social behaviour and emotions of our listening personal.

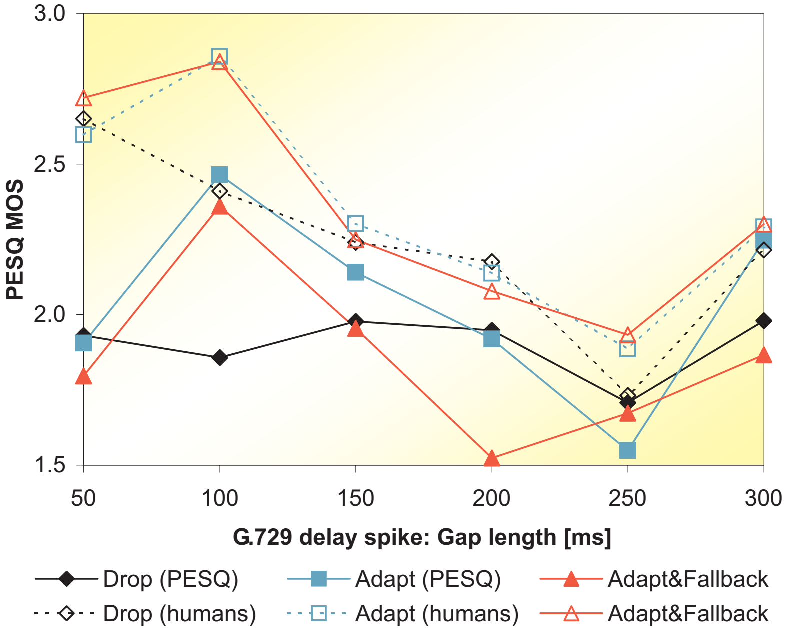
4.4. Summary

In this chapter we have presented an approach of how to assess the quality of VoIP transmissions. The following contributions have been made:

- We developed a formula on how to convert MOS values to R Factor, which the ITU has adopted as standard extension [87]. We use this formula to include PESQ into the E-Model.
- We verified whether PESQ can measure the impact of single frame losses or non-random packet loss – a source of impairment for which PESQ has not been designed. To construct samples for experimental tests, we developed a tool that controls the loss of specific frames, e.g. only important or voiced frames. We conducted subjective, formal listening-only tests to verify PESQ's prediction performance. The human ratings correlate with PESQ with a degree of R=0.94. Given the precision of speech quality measurements we can assume the equality of subjective and instrumental results.



(a) ITU G.711



(b) ITU G.729

Figure 4.10.: Playout strategies: delay spike height vs. speech quality.

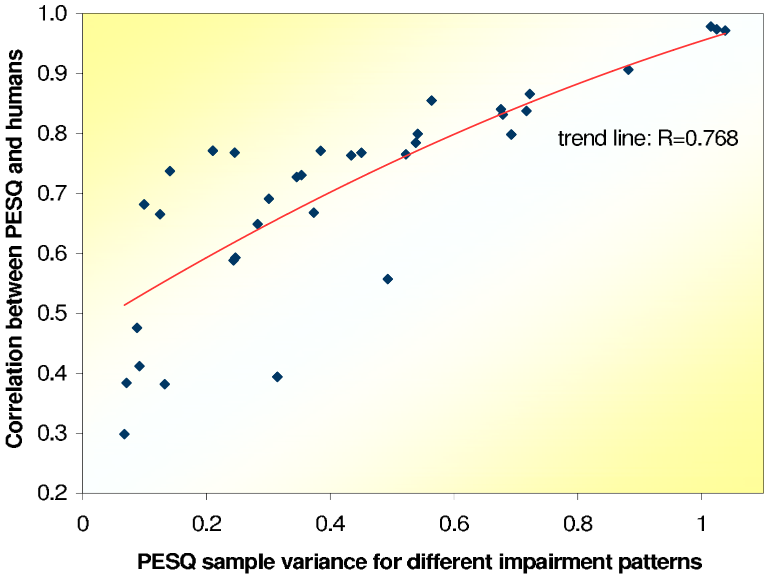


Figure 4.11.: Variance of different sample sets against the correlation between PESQ and humans. We took the data from the packet loss listening-only tests and split it to sets of varying variance (e.g. only G.711 coded or only unvoiced losses).

Table 4.5.: Listening-only test results showing the PESQ performance in cases of delay spikes.

Selection criteria:	**all**	**MNRU**	**Coding**	
			G.711	**G.729**
Samples	113	13	33	63
MOS	2.518	3.013	2.565	2.284
PESQ MOS	2.280	2.823	2.277	2.028
PESQ var.	0.723	1.015	0.564	0.373
Correlation	**0.866**	**0.978**	0.856	0.668

Selection	**Gap length**			**Playout strategy**		
criteria:	**100ms**	**200ms**	**300ms**	**drop**	**adapt**	**&fallback**
Samples	15	18	18	32	32	32
MOS	2.844	2.228	2.280	2.220	2.453	2.469
PESQ MOS	2.680	1.873	1.998	2.039	2.243	2.058
PESQ var.	0.882	0.141	0.246	0.088	0.717	0.541
Correlation	0.906	0.737	0.768	0.476	0.838	0.799

- We conducted formal listening-only tests to verify the prediction performance of PESQ for impairments due to playout rescheduling, which are caused by jitter and adaptive playout scheduling. The overall correlations are R=0.87.
- Finally, we implemented the most common playout schedulers and provide them to the research community as open-source software [91]. Also, because perceptual speech quality assessment is computationally complex, we developed various performance optimizations. For example, we provide a tool that runs the calculations in parallel [81].

Our software package allows researchers to assess VoIP transmissions with a precision not reached before.

4.5. Outlook

Several issues could be addresses in future research. One should note that the E-Model is based only on an assumption that uncorrelated sources of impairment can be added on a psychological scale. This assumption was not verified with formal listening-only tests. Recently, Takahashi et al. [202] have conducted listening-only tests to study the interaction between transmission delay and speech quality (see Figure 4.12). Their results show that the E-Model overestimates the combined impairment of delay and speech quality. Thus, our quality model should replace the E-Model equation by an equation modelling Takahashi's results. Then, the accuracy would be improved further.

In general, PESQ indeed predicts well the speech quality. However, we found that in same cases PESQ has to be enhanced and or re-tuned. These improvements are beyond the scope of this thesis. We published the complete experimental data including all samples and ratings on our web page [82]. Thus, an improved algorithm can be tuned and verified using our experimental test results.

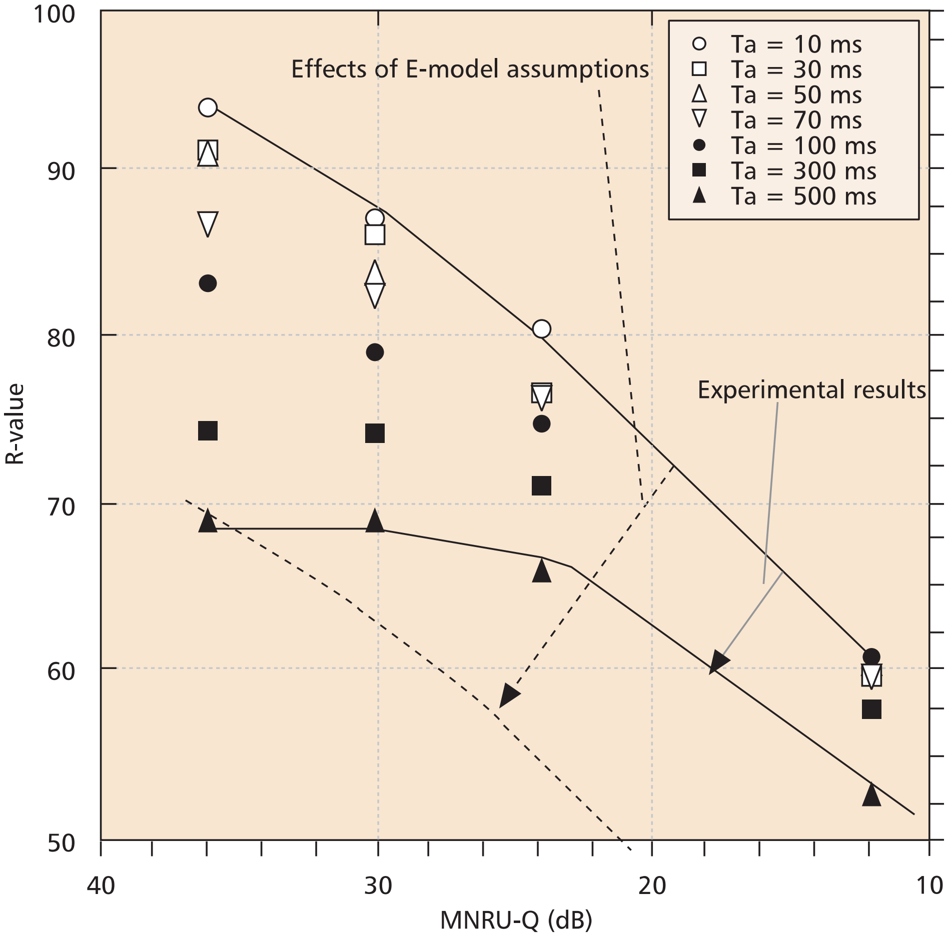


Figure 4.12.: Interaction between delay and speech quality (MNRU) [202].

5. The Importance of Individual Speech Frames

If highly compressed multimedia streams are transported over packet networks, losses of individual packets can impair the perceptual quality of the received stream by different degrees, depending on the content and context of the lost packets. In this chapter, we investigate the impact of individual speech frame and packet[1] losses on the perceptual speech quality.

An off-line measurement procedure is described, which measures the impact of loss on speech quality and quantifies the importance of frames. We used this method in an extensive experiment effort evaluating more than two million different, deliberately simulated packet and frame losses. Hereby we consider the most common standardized, narrow-band speech codecs and concealment algorithms, which are Adaptive Multi-Rate (AMR), G.711 plus Annex I[2], and G.729. This enables us to precisely determine the behaviour and statistics of frame losses. Also, we have validated our method with formal listening-only tests (refer to Section 4.2).

Applying the knowledge about frame importance, both simulations and informal listening-only tests show that only a fraction of all speech frames need to be transmitted if (at least) speech intelligibility is to be maintained.

This chapter is structured as follows: First, we describe the procedure to measure the impact of an individual frame loss. Based on this result, we introduce a metric describing the *importance* of a frame. In Section 5.5 we present an algorithm that preferentially drops unimportant frames in order to optimize the speech quality for a given frame loss rate. The interested reader can assess this algorithm via a public web interface in order to obtain a subjective impression of the different effects caused by different frame losses [83].

5.1. Experimental set-up

In order to study the impact of frame losses on the speech quality, we conducted experiments as depicted in Figure 5.1. They are similar to the experimental set-up described in [200].

We used speech recordings, taken from an ITU coded speech database [102] that consists of 832 files, each 8 seconds long, with 16 different speakers, 8 female and 8 male, spoken in four different languages, without any background noise. We chose this database to limit the influ-

[1] One VoIP packet consists of one or multiple speech frames.

[2] Its ”frame” length is set to 10 ms.

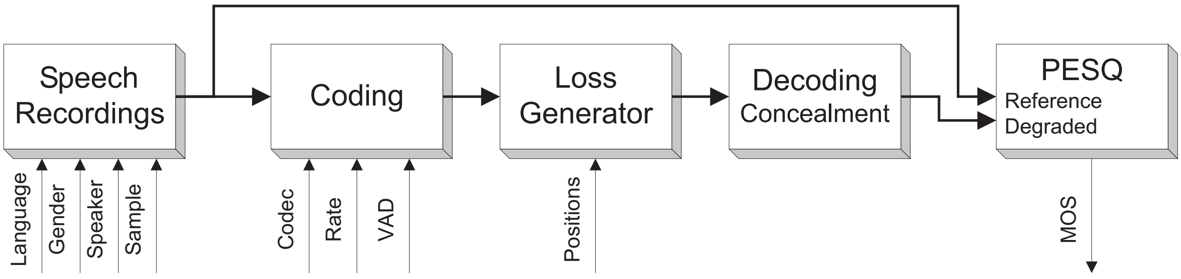


Figure 5.1.: Measuring one frame loss.

ence of specific languages [62], speakers, or samples. We chose three common narrow-band-speech-coding algorithms: ITU's G.711 and G.729, and ETSI's Adaptive-Multirate (AMR).

We simulated packet losses at different positions within the sample. For example, we dropped a single VoIP packet containing 10, 20, 40, 60, and 80 ms of speech data. We varied the coding scheme, the packet loss positions and the sample content, and generated for each test case a degraded audio sample. To assess the speech quality we applied the ITU's PESQ algorithm [107] to calculate a MOS value. In preparation of this thesis, some million PESQ rating results were gathered to achieve a high accuracy for statistical analysis.

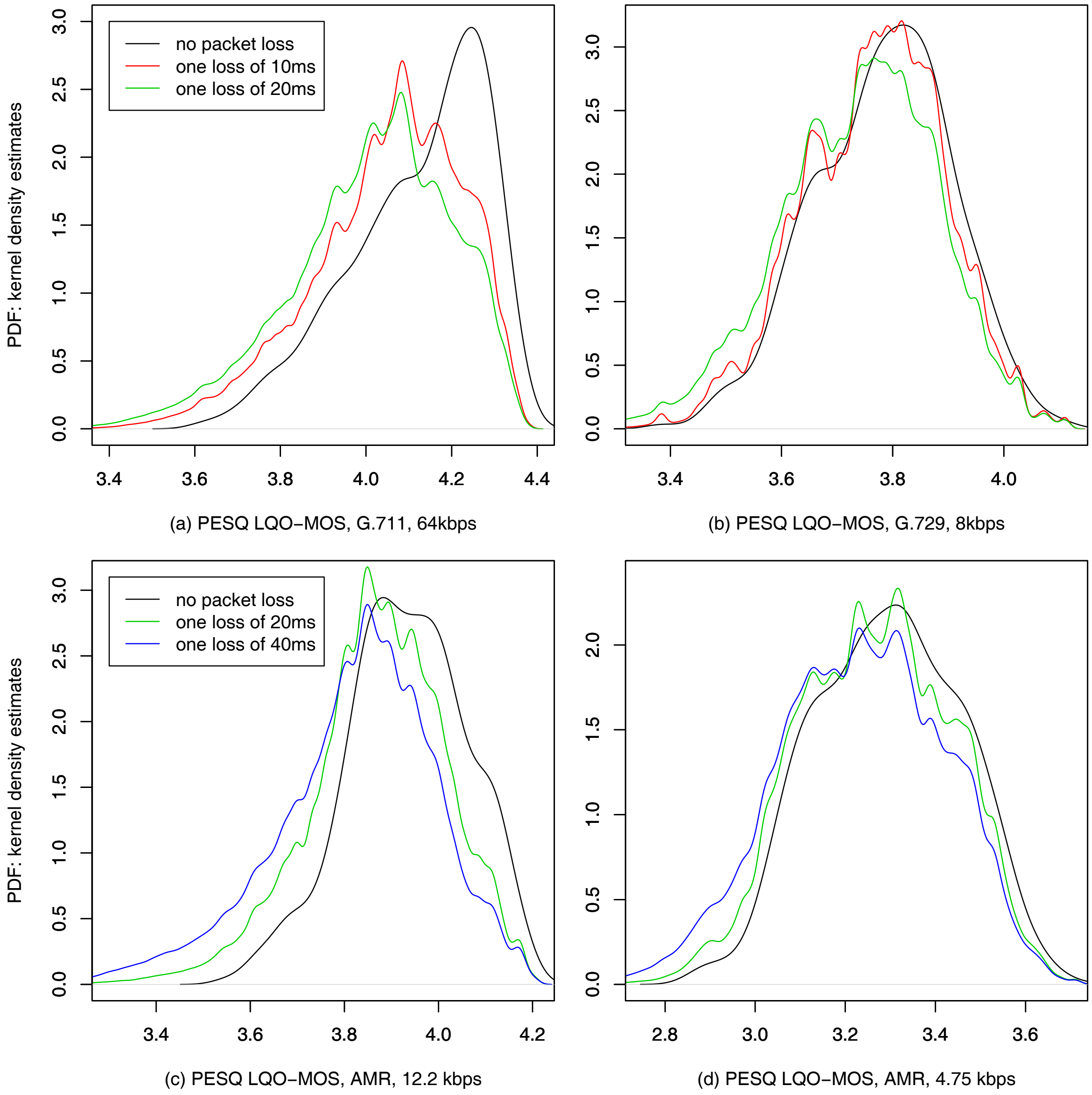
5.2. The impact of a single loss: Results

In Figure 5.2 we plot the distribution of MOS values by varying the speech sample content (e.g. sentence and speaker). The Figure is similar to a histogram with infinite small bars. Actually, we calculate the density estimation and use an automatic kernel bandwidth estimates [172, 191]. Thus, the coloured lines are curlier because the density estimates are based on a higher number of MOS values and the kernel bandwidth is smaller.

It is well known that the coding scheme has a large impact on the speech quality. But also the sample content has a large influence. Thus, one cannot assume that the perceptual *coding distortion* remains the same if the sample content is varied. We also display the loss of one and two frames. In cases of losses, the speech quality is decreased more than it is when experiencing encoding distortion only. However, when compared to varying coding distortion, the packet *loss distortion* is small.

5.3. Verification with listening-only tests

PESQ has been designed to take into account random frame losses as well as coding distortion and yields high correlations with human ratings [159]. In the case of single frame losses, however, PESQ operates beyond its specification [16] and its predicting performance is not known. Therefore, we have conducted formal listening-only tests and compared the human



(a) PESQ LQO−MOS, G.711, 64kbps

(b) PESQ LQO−MOS, G.729, 8kbps

(c) PESQ LQO−MOS, AMR, 12.2 kbps

(d) PESQ LQO−MOS, AMR, 4.75 kbps

Figure 5.2.: Impact of coding distortion and packet loss on speech quality. The probability density functions (PDF) are displayed for varying sample content (e.g. sentence). The samples have an equal length of 8 s. The black line shows the impact of coding distortion. The coloured lines display the impact of losing one and two speech frames.

ratings with PESQ predictions [92]. The result is given in Section 4.2. The overall correlation coefficient is R=0.94. Thus, human and PESQ ratings are very similar.

5.4. A metric for single frame loss

If the impact of frame loss is described with the MOS value, the MOS value largely depends on the sample content and the coding scheme. In order to eliminate this interfering dependence, we have developed the following heuristic metric[3] that will be applied in this chapter. Its definition is:

If a sample is encoded, transmitted and decoded, the maximal achievable quality of transmission is limited by the coding performance, which depends on the codec algorithm, its implementation, and the sample content. Some samples are more suitable to be compressed than others (see Figure 5.2). For a sample s, which is coded by an encoding and decoding implementation c, the quality of transmission is MOS(s,c). The sample s has a length of $t(s)$ seconds. One should note, that the length of a sample excludes the leading and subsequent periods of silence, which are usually not relevant to evaluate the perceptual quality.

In a VoIP system, the quality is not only degraded by encoding, but also by frame losses. If such losses occur, the resulting quality is described by MOS(s,c,$\{l_1$,l_2,$\dots\}$). The values of l_x describe the loss of a speech frame at position x. We define the frames importance as *"The importance of frame losses is the difference between the quality due to coding loss and the quality due to coding loss plus frame losses, multiplied by the length of the sample".* The following empiric equation describes how to calculate the importance.

$$\begin{aligned} &Imp^{*}\left(s,c,\{l_1,l_2,\ldots\}\right) = \\ &\left(MOS\left(s,c\right) - MOS\left(s,c,\{l_1,l_2,\ldots\}\right)\right)\cdot t\left(s\right) \end{aligned} \tag{5.1}$$

If a frame loss occurs in sample of 8 seconds length it is stronger than a frame loss in sample of 16 s. Thus, we have to normalize the perceptual difference with the sample length to get a statement on the loss impact independent of the sample length.

Using the data shown in Figure 5.2 we display the distribution of individual frame importance values in Figure 5.3. Many frames are not important at all.

5.5. Frame dropping strategies

The previous sections describe a method to classify the impact of individual and multiple frame losses. But how this can be applied? In the following we assume a scenario in which we know the importance of each frame and in which we can control the loss process - e.g. which

[3] This first intuitive definition of frame importance has been presented in [89]. As we showed, this definition does not scale linearly when considering more than one frame. Therefore we provide a better equation in the following chapter.

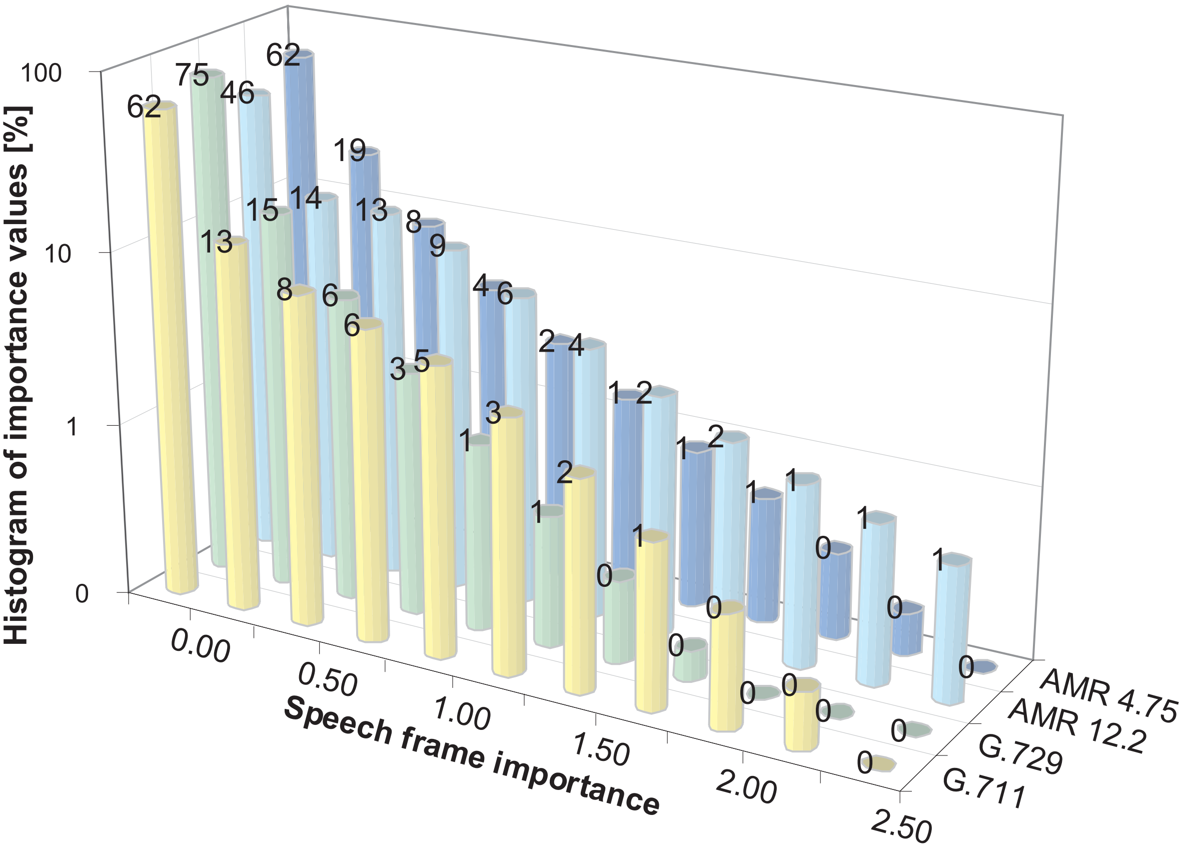


Figure 5.3.: Distribution of speech frame importances.
Many speech frames are not important at all.

frames can be dropped. We assume that we can drop frames at any position and show, how the *frame dropping strategy* influences the speech quality.

If frames have to be dropped, which frames should be dropped? Classic approaches chose the frames randomly (line "random"). Discontinuous Transmission (DTX) algorithms detect the voice activity (VAD) to interrupt the frame generation. They reduce the transmission rate during inactive speech periods whilst maintaining an acceptable level of output quality. Thus, a DTX algorithm would first drop silent frames, and then active frames (line "DTX"). Using the metric of frame importance, we introduce two novel strategies: As a "worst" case we consider a dropping strategy that drops the most important frames first. The second called "best" loss strategy preferentially drops the less important frames. Only at high loss rates important frames are dropped.

In our simulations, we increase the dropping rate from 0% to 100% in steps of 2% using four coding modes. We use 832 different samples but consider only the mean MOS value over all different samples. For high loss rates, PESQ MOS values tend to increase again. One should note that PESQ has not been designed to measure very high frame loss rates. We therefore assume that PESQ in these ranges operates beyond its specification and the values should be interpreted as MOS=1.

In Figure 5.4 we display the speech quality depending on the frame loss rate. One can see that the "best" loss strategy performs better than the DTX and random case. In case of the worst strategy, the speech quality drops very fast. In order to allow a better comparison, we adapt the semantic of the X-axis (frame loss rate) in Figure 5.5 to count only active frames.

5.6. Coding rate versus frame dropping strategy

One question remains: Is it better to use a good frame dropping strategy or to lower the coding rate?[4] In Figure 5.6 we use the same measurement results as displayed in Figure 5.4 but we display on the X-axis the resulting bandwidth that we calculated using Equation 9.3. We display only the AMR 4.75 and 12.2 kbps coding and the best, DTX and random strategies.

One can see that for a bandwidth of exactly 4.75 kbps, the best frame dropping strategy using 12.2 kbps coding rate is as good as the 4.75 kbps coding rate without any losses. Interestingly the best dropping strategy with 12.2 kbps is always better than 4.75 kbps with random dropping. Thus, optimisations should include selecting the best coding rate and include a proper frame dropping strategy in order to allow for fine grain adaptations.

[4] A similar problem is studied in Chapter 9.4 considering only the random dropping strategy. Here we extend it to different frame dropping strategies.

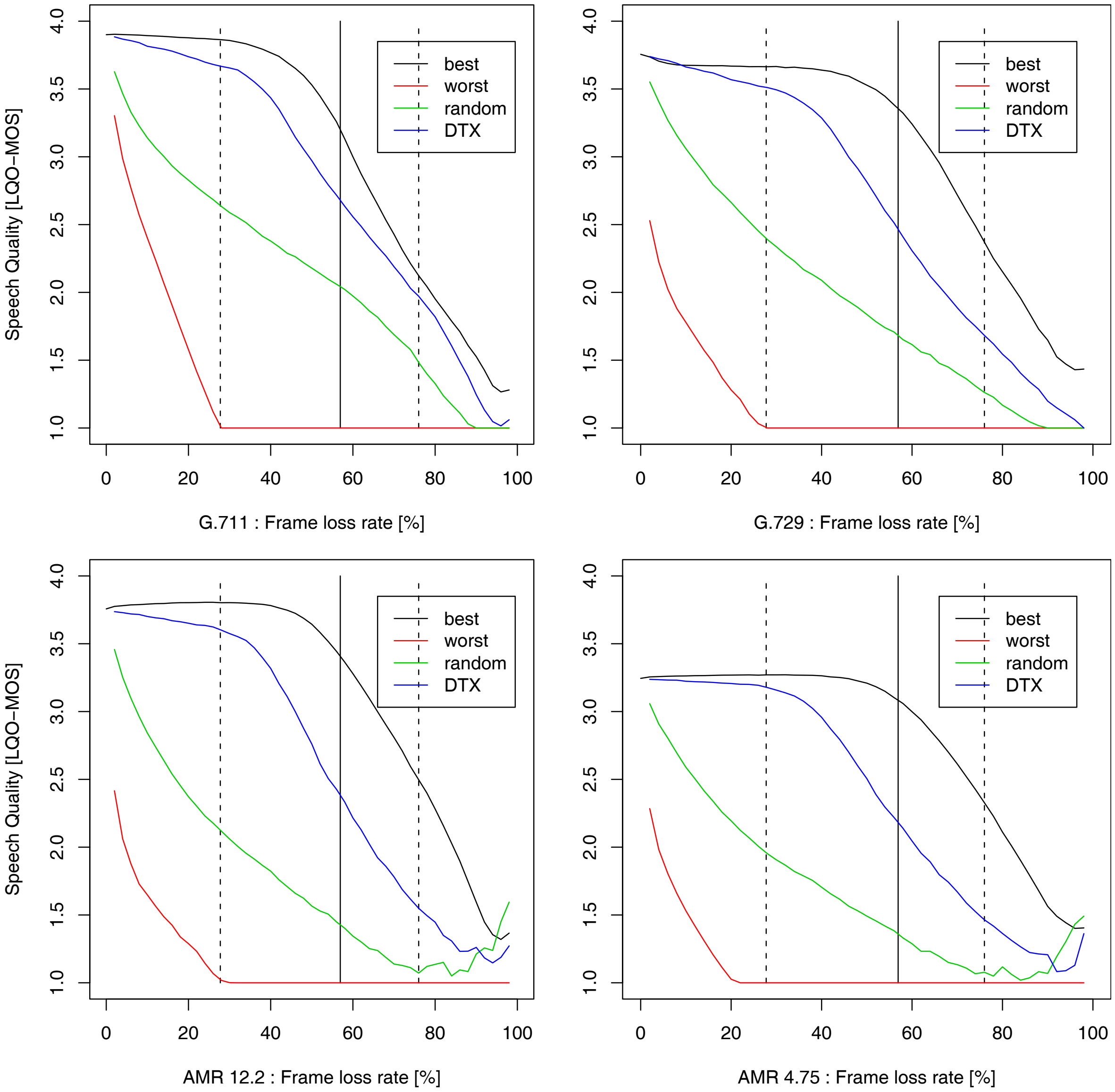


Figure 5.4.: Impact of dropping strategy on the speech quality.
The grey, vertical lines refer the minimal, mean, and maximal percentage of silent frames in the sample set. Strategy "best" drops first all the less important frames and at last the important once. Strategy "worst" drops the important first. "Random" contains drops frame without any distinguishing. "DTX" drops first randomly silent frames and then randomly active frames.

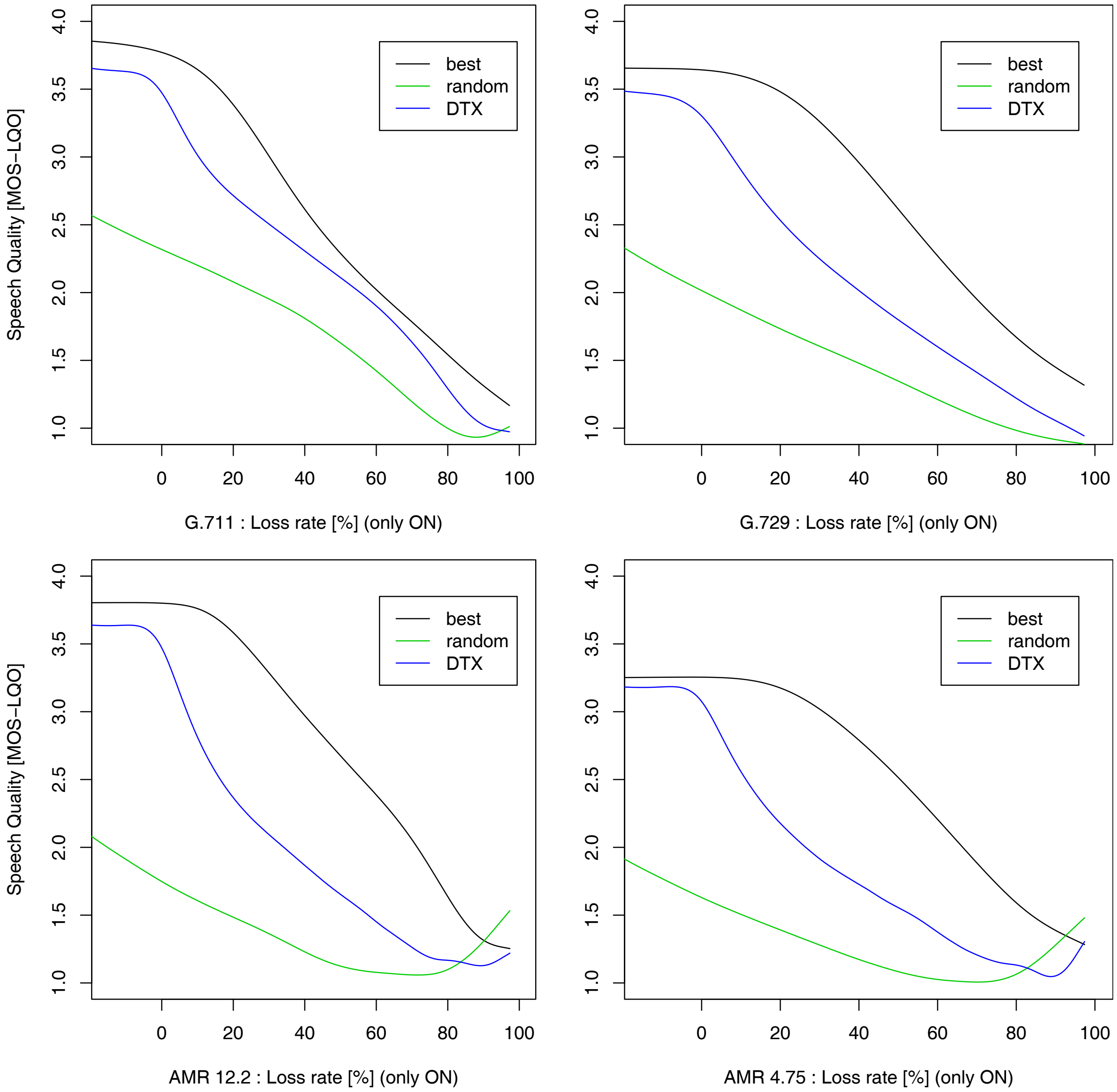


Figure 5.5.: Relation between frame loss rate (counting only active speech frames), dropping strategy, and speech quality.

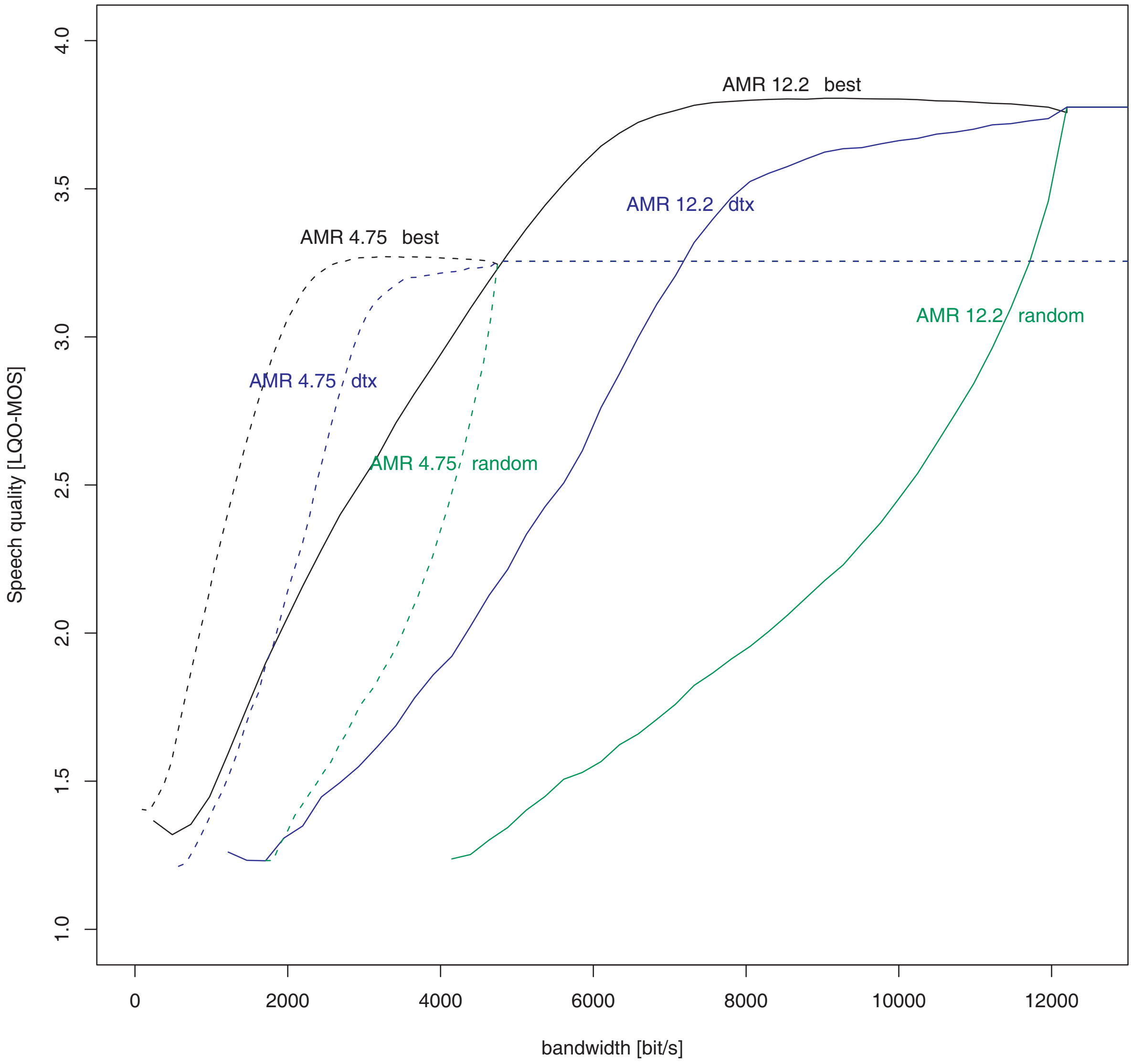


Figure 5.6.: Impact of dropping strategy and coding rate on the speech quality.

5.7. Impact of the prediction performance

As we will show in Chapter 7 it is not possible to predict the frame importance perfectly in real-time because the importance values are falsified by incomplete knowledge of the amount of error propagation. Also, if the algorithm to classify frames is optimised regarding computational complexity, this might introduce additional sources of error. This section identifies the impact of imperfect importance calculations.

For the sake of generality we try to model the prediction errors by a Gaussian distributed error. To each importance value a random error signal is added. Then, as in the previous sections the impact of the frame dropping strategy is studied but not with perfect importance values bt with noisy values.

Let us rate the predicting performance (the quantitative quality) of the algorithm by calculing the correlation coefficient value. We compared the imperfect with the reference values. The strength (width) of random noise is selected in such a way that the distorted importance values correlate to the reference values with a degree of $R \in \{0, 0.2, 0.4, 0.6, 0.8, 1\}$.

But how strong should the error signal be to obtain a given correlation coefficient? We have to find a way to determine the proper strength of the Gaussian error signal. Because it is not possible to find an analytical solution we had to implement an algorithm, which searches for the proper strength. It conducts a logarithmic search to find the approximate error strength. The algorithm works as follows.

1. A Gaussian error signal is generated for each frame importance value. To calculate the Gaussian error we use the following equation:

$$\begin{gathered} v = \sqrt{-2 \cdot \log u_1} \cdot \cos 2\pi u_2 \\ \text{with the random variables } u_{1,2} \in [0;1[\end{gathered} \tag{5.2}$$

2. Each importance value imp_{pos} at position pos is added with the random signal that has an initial value of $e = 1024$. The strength e changes with each iteration.

$$imp_{pos}^{new} = imp_{pos} + e \cdot v_{pos} \tag{5.3}$$

3. Next, the correlation coefficient is calculated and the following Algorithm, written in pseudo C-code, is executed. It conducts a search to find the proper R value, which has a precision of ±0.001 or better.

4. Finally, the distorted importance values are used for the frame dropping strategy. Of course, this time other frames are considered as important and the ranking of frames is changed.

Figures 5.7 and 5.8 display the results. They differ as the first figure compares the best

Algorithm 1 Finding the optional Gaussian error strength to achieve a given prediction performance, which is measured with the correlation coefficient R.

```
double findR(double target, double precision)
{
  double actual, noise;
  double min_noise = 0, max_noise = 1024;
  double min_r = 1, max_r = 0;

  forever {
    if(min_r < target+precision
       && min_r > target-precision) {
# Target R value is reached! It is the lower value.
      noise = min_noise;
      addNoise();
      return correlation(noise);
    }
   if(max_r < target+precision
      && max_r > target-precision) {
# Target R value is reached! It is the higher value.
      noise = max_noise;
      addNoise();
      return correlation(noise);
    }
# Test the mean noise value.
    noise = (min_noise+max_noise)/2;
    addNoise();
    actual = correlation(noise);
    if(actual<0) actual=0;
    if(actual<target) {
# The mean is now the maximal R value.
      max_noise = noise;
      max_r = actual;
    }
    else if(actual>target) {
# The mean is now the minimal R value.
      min_noise = noise;
      min_r = actual;
    }
  }
}
```

dropping strategy (R=1) to the random dropping (R=0), whereas the second compares best (R=1) with DTX (R=0). One can see that only a perfect prediction achieves remarkable performance gains. Even at a prediction accuracy of R=0.8 the effect is only about half as good as the best case. This leads to the conclusion that is highly important to predict the frame importance precisely.

5.8. Conclusions

The importance of speech frames differ widely. We developed an off-line method to quantify the impact of frame loss on speech quality. Using the knowledge of frame importance we showed that remarkable performance gains can be achieved if only those packets are transmitted that are important.

Using these algorithms on mobile phones, for example, can reduce transmission energy significantly and thus extend their battery lifetime: If we know the importance of speech frames, fewer frames need to be transmitted and the mean transmission power is reduced.

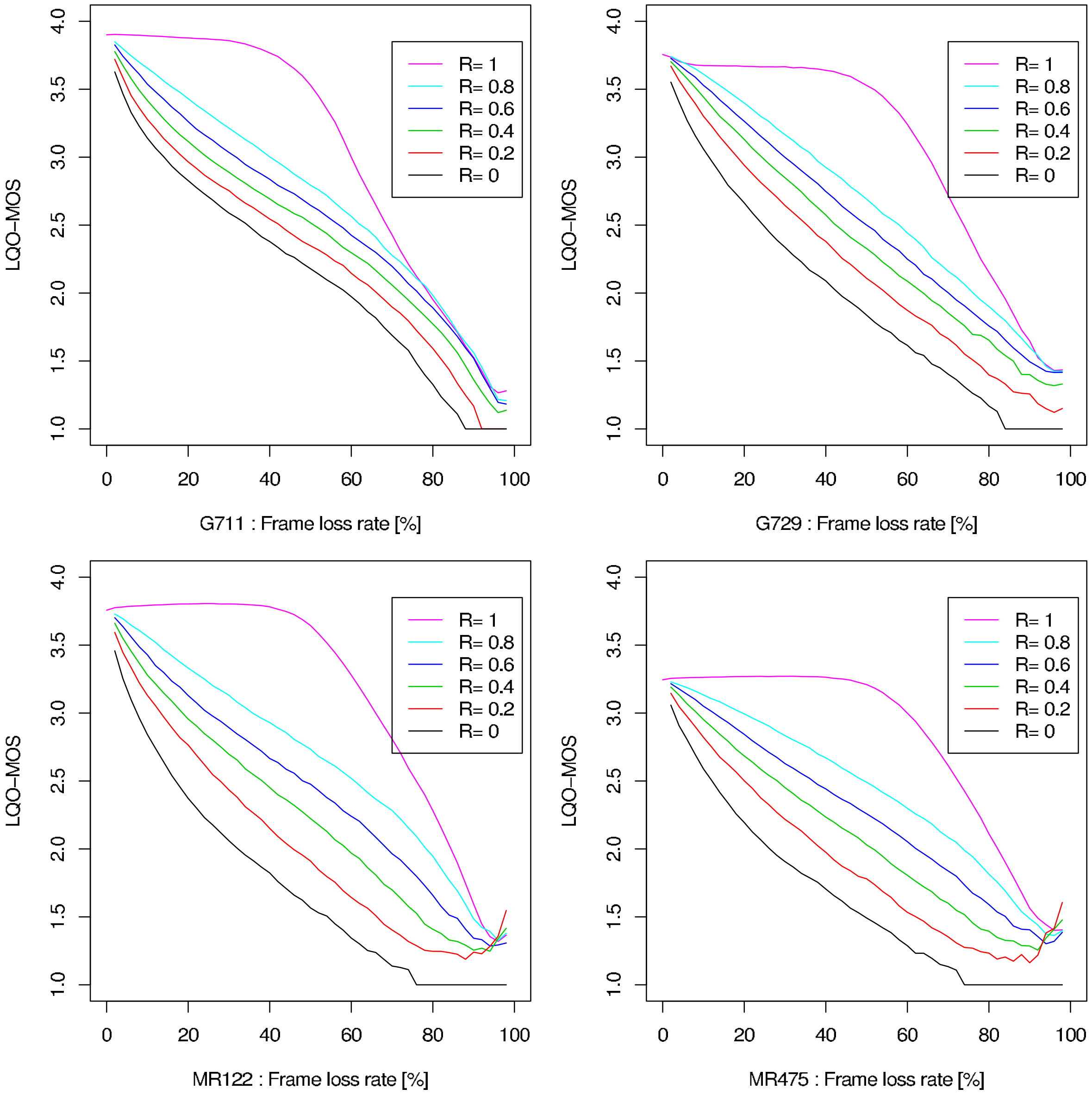


Figure 5.7.: Impact of prediction performance on the impact of the frame dropping strategy. The best (R=1) and the random (R=0) dropping strategies are displayed. Also dropping strategies are displayed, in which the frame classification is falsified by a Gaussian distributed error signal.

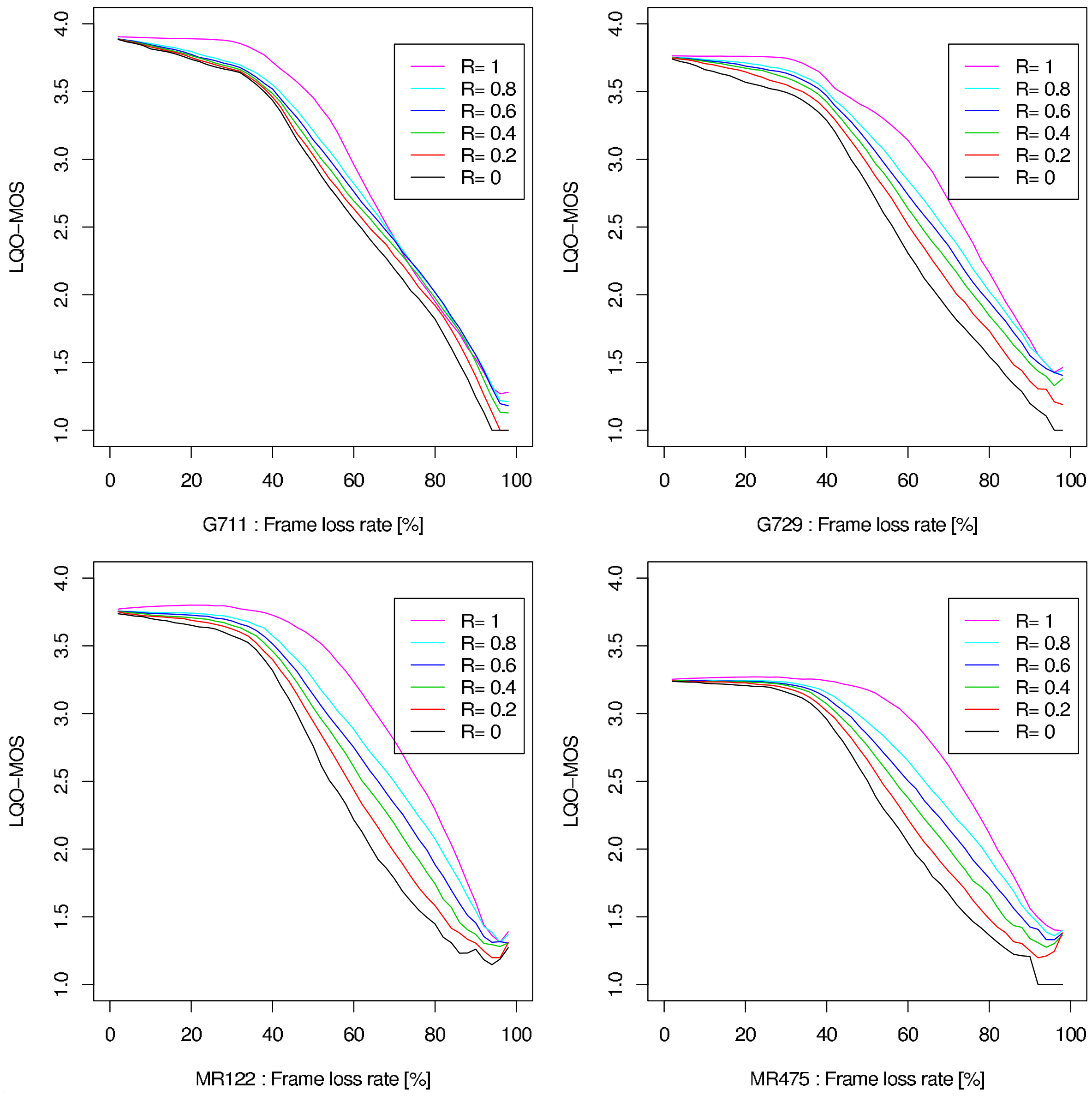


Figure 5.8.: Impact of prediction performance on the impact of the frame dropping strategy. As Figure 5.7 but this figure displays the best (R=1) compared to the DTX dropping strategy (R=0) .

6. Calculation of Speech Quality by Aggregating the Impacts of Individual Frame Losses

Losing VoIP packets or speech frames decreases the perceptual speech quality. The statistical relation between randomly lost speech frames and speech quality is well known [141]. In packet-based communication networks, such as the Internet, packet losses are a major source of quality degradation. One would expect that the impact of VoIP packet loss on speech quality is well understood. However, this is not the case as it is a highly interdisciplinary problem: Multiple "layers" have to be considered covering the loss process of IP-based networks, the behaviour of speech codecs and frame loss concealment, the psychoacoustics of the human hearing, and even the cognitive aspects of speech recognition.

State the of art algorithms look up speech quality scores in tables, depending on the measured loss rate and the speech coding [105]. Alternatively, these tables can be modelled as linear equations [198] or with neural networks [145]. However, the relation between mean packet loss rate and speech quality is only a statistical description as the deviation for specific loss patterns can be high and also depends on the content of the lost frames. Also, these relations are only valid for a specific loss pattern. For example, bursty losses (multiple consecutive VoIP packet or speech frame losses) can have a different impact [112, 114, 201] depending on the codec and the duration of the burst.

In Chapter 5, we describe a method that measures the *importance* of a single speech frame [89]. In this chapter we assume that we can use this method to determine the impact of a single frame loss. Then, the question arises how the impact of multiple losses can be determined using the importance of only single frame losses. The development of a novel metric or dimension of frame importance, which simply can be summed to obtain the overall impact of multiple frame losses, is presented in this chapter.

The ITU-T P.862 PESQ algorithm [16, 107, 176] can assess the impact of single or multiple frame losses but only works for audio files and not on a speech frame level. PESQ by itself cannot be directly applied to VoIP packets (refer to Chapter 9) and has a high computational delay and complexity, which limits its on-line and real-time use.

Thus, we remodel the internal behaviour of the PESQ algorithms using it for frame losses: We apply the algorithm that PESQ uses to aggregate signal distortions over time in order

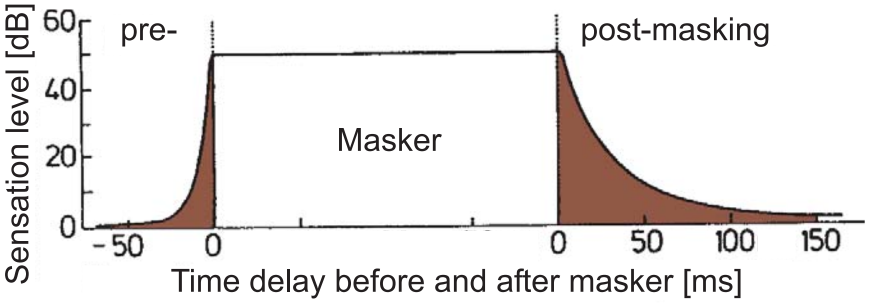


Figure 6.1.: Schematic drawing to illustrate and characterize the regions, in which pre- and post-masking occur (shaded areas) if a masker is present [227].

to accumulate frame loss distortions over time. This aggregation algorithm is also the basis for the novel importance metric, which allows adding the frames' importance linearly and thus has a low complexity. It shows a high prediction performance, if the losses are distant. If frame losses occur shortly one after the other, temporal auditory masking effects have to be considered. We develop a heuristic equation to model these effects. Overall, our approach shows a high correlation with instrumental speech quality measurements for many loss patterns.

The following section of this chapter first describes the required psychoacoustic background. Then, we present an approach of how to assess multiple speech frame losses. In the Section 6.3 we compare our approach with the PESQ's speech quality predictions. Finally, we summarize this chapter and give an outlook for further research.

6.1. Temporal masking

Zwicker and Fastl [227] describe temporal masking effects which characterize human hearing: The time-domain phenomena pre- and post-masking plays an important role (Figure 6.1). If faint sound follows shortly after a loud part of speech, the faint part is not noticeable because it is masked. Also, if the maskee precedes the masker, the maskee vanishes.

If the temporal masking effect is applied to distortion values, the distinction between masker and maskee on one side and between pre- and post masking on the other side is difficult: If a frame is lost it causes a distortion, resulting in a segment of speech which can be louder or fainter than the previous segment. If it is louder, the previous segment is pre-masked, if it is fainter, the loss is post-masked. Thus, if one considers only distortion, it is not possible to distinguish pre- and post-masking. Instead, the same amount of distortion can cause between pre- or post-masking or can be effected itself by pre- or post-masking, depending on the loudness of the resulting speech segment.

The masking effects had been considered as an addition to the PESQ algorithm. However, after implementing it, it did not result in any improvements of the prediction performance of PESQ. Thus, it was not included.

6.2. Additive metric

A metric that describes the importance of frames should fulfil the following requirements: First, it should be easily deployable to quantify the impact of frame losses. Consequently, the loss distortion should be measurable with off-the-shelf instrumental measurement methods like PESQ (or any other successor). For example, it should be able to calculate the metric with two speech quality measurements: with and without loss. Second, the metric shall be one-dimensional (in the meaning of unimodal). Of course, the distortions caused by frame loss can have many effects. However, it should be modeled as an one-dimensional quality scale as this would simplify the development of algorithms that utilize this metric. Last, it should be possible to give a statement like “frame A and frame B are as important as frame C” or “frame A is three times more important than frame B”. In a mathematical sense, this requirement is called the *Additive Property of Equality* [211]. It is of importance when frame loss impacts are to be applied in analytical contexts such as the rate-distortion multimedia streaming framework by Chou and Miao [41].

The development of such a metric is based on the idea to study the internal behavior of PESQ and to remodel it for multiple frame losses. PESQ predicts the impact of frame loss rather well but is far too complex to be applied on a frame basis. Thus, a simpler model is required that only contains the issues that are relevant. The proposed approach is based on the following three principles. First, we assume that the importance of frames is known. Second, if two or more frame losses have a distance of more than 320 ms, the importance values, as calculated by Equation 6.6, can simply be added. Last, if two frame losses occur shortly after each other, then Equation 6.9 is required to add the importance values. In the following it is described how we have developed this approach by remodelling PESQ’s behavior.

1. Asymmetric effect: PESQ judges the impact of distortion with two factors, the asymmetric and the normal distortion (Section 3.1.1). The asymmetric effect is mainly influence by the encoding distortion as the encoding reduces the transmission spectrum. Frame losses do not change the overall transmission spectrum because a frame loss is limited to its position of loss and does not change the rest of the sample. Therefore it is reasonable to neglect the difference between asymmetric and normal distortion and only consider their sum (see Figure 6.2).

2. Long-term aggregation: In general, the disturbances weighting over time is determined as in PESQ. In PESQ the syllable disturbances are summed up as described in Equation 3.3.

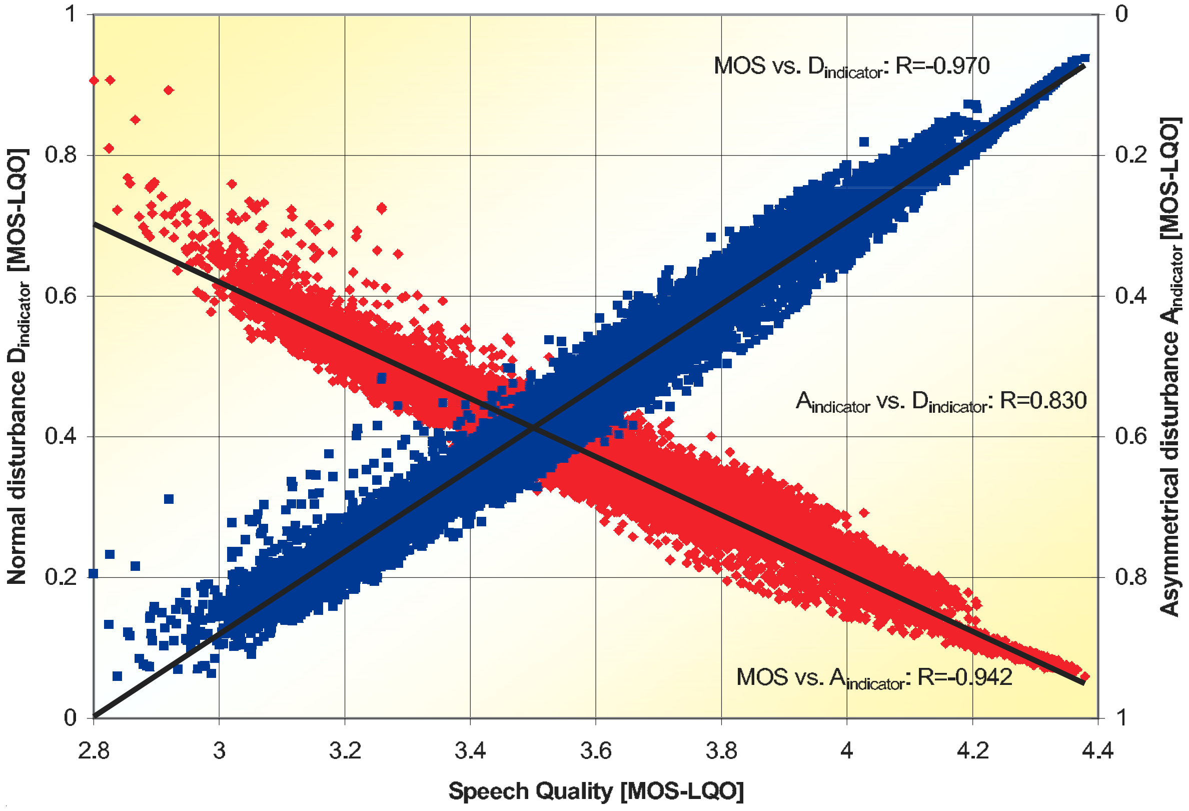


Figure 6.2.: Correlations between the PESQ MOS-LQO value, the asymmetrical distortion and the normal distortion (refer to Equation 3.1).
The observation values are plotted after conversion to the MOS scale.

Contrary, we consider disturbances of speech frames instead of syllables.

The disturbance consists of coding as well as loss distortion as shown in Equation 6.1. If frame losses are not present, the term $dist_{loss}[i]$ is zero.

$$sylablle_{indictor}[i] = dist_{coding}[i] + dist_{loss}[i]\,. \tag{6.1}$$

Combining (3.3) and (6.1) we can write

$$(AorD_{indicator})^2 = \frac{1}{N}\sum_{n=1}^{N}\left(dist_{coding}[n] + dist_{loss}[n]\right)^2 \tag{6.2}$$

and transform (6.2) to (6.3):

$$\begin{gathered} N \cdot (AorD_{indicator})^2 - \sum_{n=1}^{N} dist_{coding}[n]^2 \\ = \sum_{n=1}^{N}\left(dist_{loss}[n]^2 + 2 \cdot dist_{coding}[n] \cdot dist_{loss}[n]\right) \end{gathered} \tag{6.3}$$

As an approximation we combine both asymmetric and symmetric disturbances. Then, (3.1) can be simplified to:

$$MOS = 4.5 - AorD_{indicator} \tag{6.4}$$

with $AorD_{indicator} = 0.1 \cdot D_{indicator} - 0.0309 \cdot A_{indicator}$. Combining (6.3) and (6.4), we get:

$$\begin{aligned} &\left((4.5 - \mathrm{MOS}\,(s,c,e))^2 - (4.5 - \mathrm{MOS}\,(s,c))^2\right) N \\ &= \textstyle\sum_{n=1}^{N} \left(dist_{loss}\,[n]^2 + 2 \cdot dist_{coding}\,[n] \cdot dist_{loss}\,[n]\right) \end{aligned} \tag{6.5}$$

with MOS (s,c) being the speech quality due to coding loss and MOS (s,c,e) being the speech quality due to coding as well as frame loss. Equation 6.5 is the basis of our new importance metric. One can see that if a loss distortion does not overlap within one syllable, the distortions can simply be added.

We define Equation 6.6, which approximates a linear scale better than old metric given in Equation 5.1.

$$\begin{gathered} Imp\,(s,c,e) = (cl - c) \cdot t\,(s) \\ \text{with } cl = (4.5 - MOS\,(s,c,e))^2 \text{ and } c = (4.5 - MOS\,(s,c))^2 \end{gathered} \tag{6.6}$$

3. Short-term aggregation: For the frame losses occurring shortly one after the other, we first model the impact of two frame losses with two delta impulses at time t_a and t_b with heights of imp_a and imp_b representing the importance. If the distance $t_{width} = t_b - t_a$ is larger than 320 ms, adding of the importance values is done as described in the previous section. Otherwise, it is calculated as explained below.

First, we calculate the probability that both losses occur in the same syllable. We assume that syllables start at $0, 320, \ldots$ ms and have a length of $t_{syll} = 320$ as in PESQ. Because of the re-occurrence pattern of syllables, it is sufficient to consider only the period of $0 \leq t_a < t_{syll}$. The overlapping of syllables can also be neglected. The probability that the two losses are within one syllable is

$$\begin{aligned} P_{in.syll}\,(t_{width}) &= \tfrac{1}{t_{syll}} \textstyle\int_{t_a=0}^{t_{syll}} \left\{ \begin{array}{cc} 0 & \text{if } t_a + t_{width} \geq t_{syll} \\ 1 & \text{otherwise} \end{array} \right\} \mathrm{d}t_a \\ &= \left\{ \begin{array}{cc} 0 & \text{if } t_{width} \geq t_{syll} \\ 1 - \frac{t_{width}}{t_{syll}} & \text{otherwise} \end{array} \right. \end{aligned} \tag{6.7}$$

If two losses are within a syllable, PESQ adds them with an exponent of $p = 6$ (refer to Equation 3.2). Because it is not simple to remodel PESQ's algorithm, we introduce the following heuristic function (Equation 6.8 and Figure 6.3), which shows similar behavior as PESQ.[1]

[1] Actually, we also tested to add cubics of importance values to model the effect of $p = 6$ but this solution did

$$add_{shortterm}\left(imp_a\,,imp_b\right)=\sqrt{{imp_a}^2+{imp_b}^2} \tag{6.8}$$

4. Overall aggregation: For a loss distance longer than the length of a syllable, it simply sums up the importance values. If it is lower, the importance values are added but the sum is weighted with a factor $1-P_{in.syll}$. Also, if the distance is short, we use another addition, which sums up the square importance values. Again, this later addition is weighted by the probability of $P_{in.syll}$:

$$\begin{aligned}&add\left(imp_a\,,imp_b\,,t_{width}\right)\\&=\left(imp_a+imp_b\right)\cdot\left(1-P_{in.syll}\left(t_{width}\right)\right)+add_{shortterm}\left(imp_a\,,imp_b\right)\cdot P_{in.syll}\left(t_{width}\right)\\&=\begin{cases}imp_a+imp_b & \text{if } t_{width}>t_{syll}\\ \left(imp_a+imp_b\right)\frac{t_{width}}{t_{syll}}+\sqrt{{imp_a}^2+{imp_b}^2}\left(1-\frac{t_{width}}{t_{syll}}\right) & \text{otherwise}\end{cases}\end{aligned} \tag{6.9}$$

Equation 6.7 partially models the time-frequency masking effect, which causes a masking of minor distortions by nearby louder ones. However, PESQ models the temporal masking effect only in the statistical mean. PESQ's masking is stronger – or at least longer – than the pre- or postmasking effect. It can be seen if one compares Figure 6.1 with 6.3. This observation explains why it was not necessary to add time masking effects to PESQ: It is already included.

6.3. Validation of the new importance metric

6.3.1. Testing the impact of random losses

We consider a scenario in which speech frame losses occur randomly and we determine for a given frame loss rate the speech quality. The same scenario has been conducted in [89] with the old importance metric based on Equation 5.1. The experimental set-up is the same as described in Section 9.3. In short, we follow the recommendation [106], conduct many instrumental speech quality measurements, and vary the coding, the sample, and the loss patterns: A speech sample is encoded, frame losses are enforced depending on the experimental requirements (in this case random packet loss), the frames are decoded or concealed, and finally PESQ calculates the MOS value by comparing the original sample with the degraded version.

Figure 6.4a displays the relationship between the random frame loss rate and the speech quality for different codecs: the higher the loss rate, the worse PESQ's speech quality ratings. Next, we calculate the importance of frame losses (Figure 6.4b). At a loss rate of 0% the

not led to higher correlation coefficient.

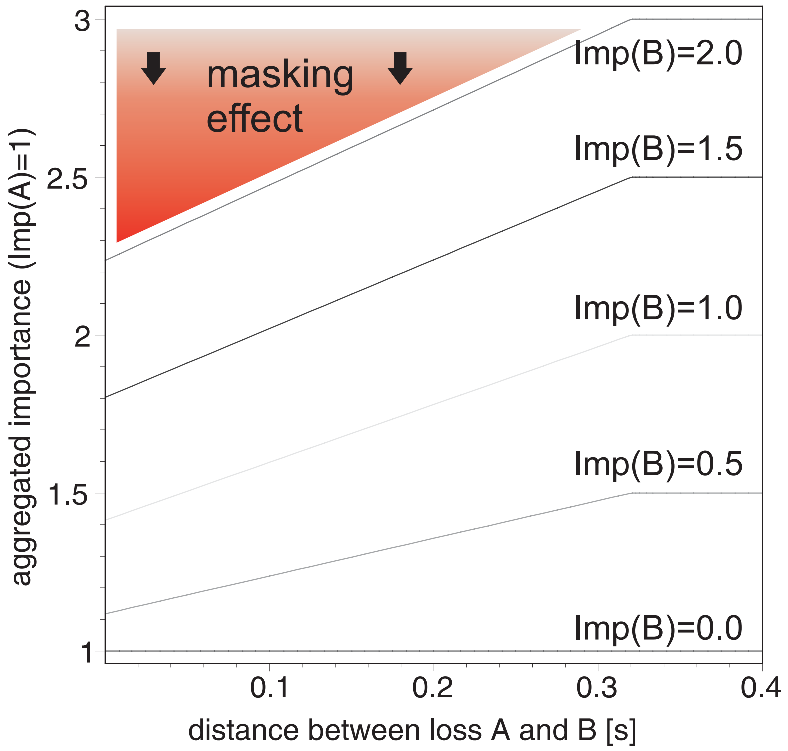


Figure 6.3.: Aggregated importance given by Equation 6.9 depending on the distance between Imp(A) and Imp(B) and the importance value of Imp(B).

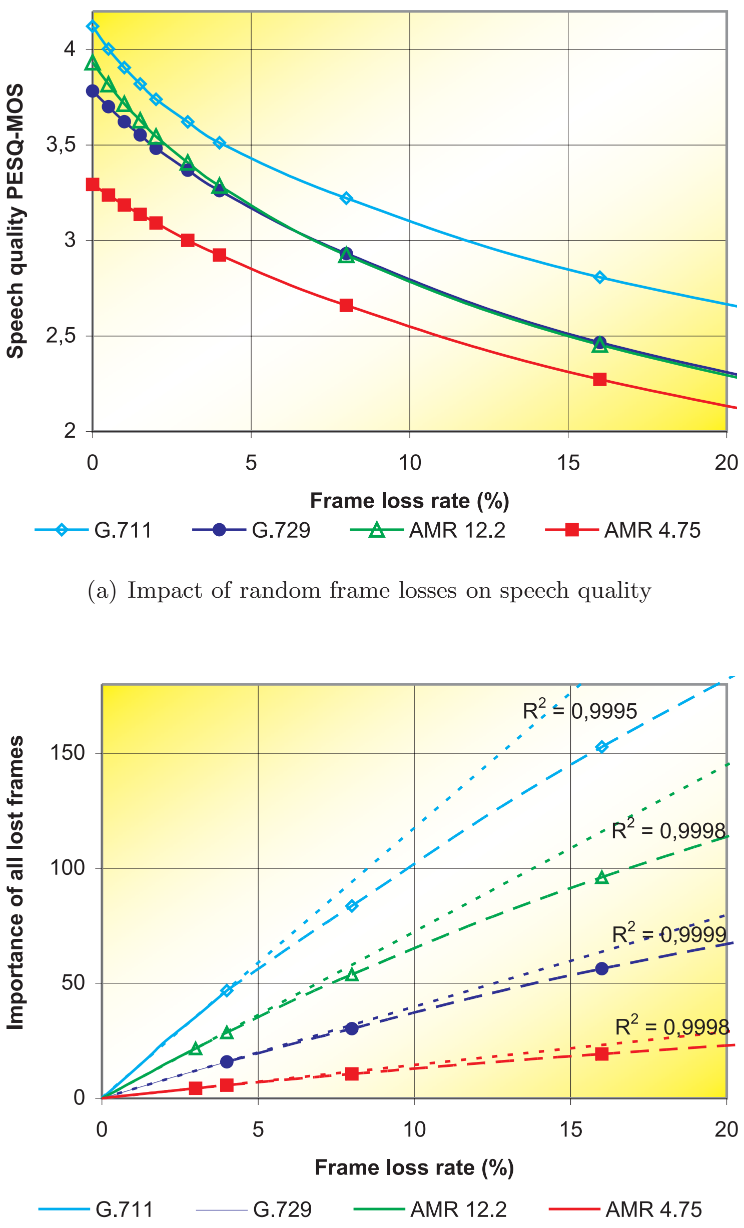
importance is 0. As long as the loss rate is low, the importance increases linearly with the loss rate.

If the impact of frame losses can be added, the following statement is valid: The overall importance of the loss of N frames can be calculated by multiplying the mean importance by N (Equation 6.10). In Figure 6.5, the mean importance as a function of the loss rate is displayed. For low loss rates the importance is slightly underestimated. In the case of loss rates over 8%, it is clearly underestimated. One should note that in this experiment the masking is not considered thus the loss impact at high loss rates is underestimated.

$$Imp\left(s, c, l_{mean}\right) \cdot N = Imp\left(s, c, \{l_1, \ldots, l_N\}\right) \tag{6.10}$$

6.3.2. The impact of a second frame loss if one loss is already present

The next experiment resembles the measurements of single packet losses described in Chapter 5, but this time we dropped two speech frames instead of one. Between the losses there is a lossless gap of 40, 80, 160, 320, or 640 ms. In Figure 6.6 we display the importance averaged over all single frame losses, vertically sorted according to the encoding scheme and marked



(a) Impact of random frame losses on speech quality

(b) Importance of many losses: measured (slashed) and predicted (dotted)

Figure 6.4.: Impact of random frame losses on the speech quality.
For loss rates (AMR <3%, ITU G.711/729 <4%) the correlation between measured and predicted importance is high.

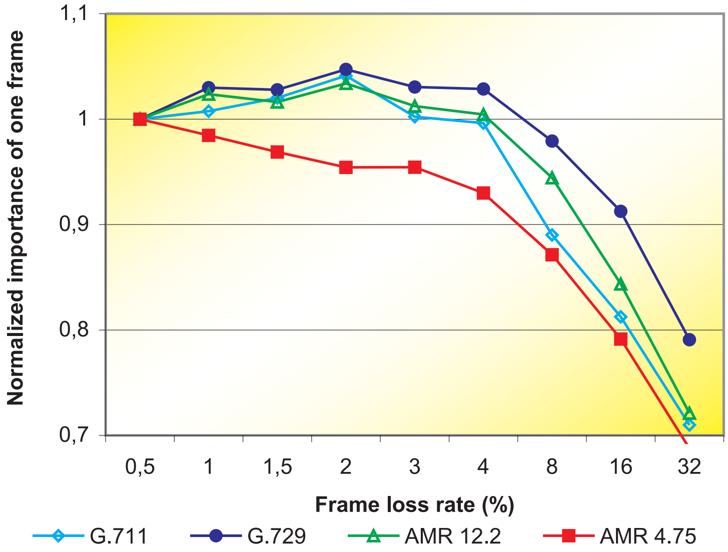


Figure 6.5.: Impact of random frame losses on the speech quality on the mean, normalized importance.

with "Single" on the horizontal axis. Also, we display the importance of the second frame l_2, if the first frame l_1 is already lost. The importance value is calculated using Equation 6.11.

$$Imp\left(s, c, \{l_2 \mid l_1\}\right) = \left(\left(4.5 - MOS\left(s_i, c, \{l_1, l_2\}\right)\right)^2 - \left(4.5 - MOS\left(s_i, c, \{l_1\}\right)\right)^2\right) \cdot t\left(s\right) \quad (6.11)$$

Considering the G.711 results, one can see that the nearer the frame losses are, the lower the importance of a frame becomes. This effect can be explained with the cognitive and auditory masking effect as described in Section 6.1.

In Figure 6.6, the mean importance for two AMR frame losses increases significantly, if the loss distance is 40 ms. We assume that this effect is due to a mismatch between the encoder's and decoder's internal state (refer to Section 7.2). The first loss results in a desynchronized decoder. The applied loss concealment leads to a wrong prediction of the frames' content. Since the de-synchronisation of the decoder can last for multiple following frames (up to 700 ms, refer to Figure 7.2), the mean impairment due to the concealment of the second loss can be significantly higher. This effect occurs only with the AMR codec, thus we will not consider it further in this work. However, a detailed analysis of AMR codec's behavior should be subject of further studies. Also, the packet loss concealment of AMR should be enhanced to be tolerant of frames losses that occur shortly one after the other separated by a small gap in between.

Figure 6.7 displays the prediction performance of losing two speech frames compared to the sum of losing two individual frames. The correlation coefficient between the importance of

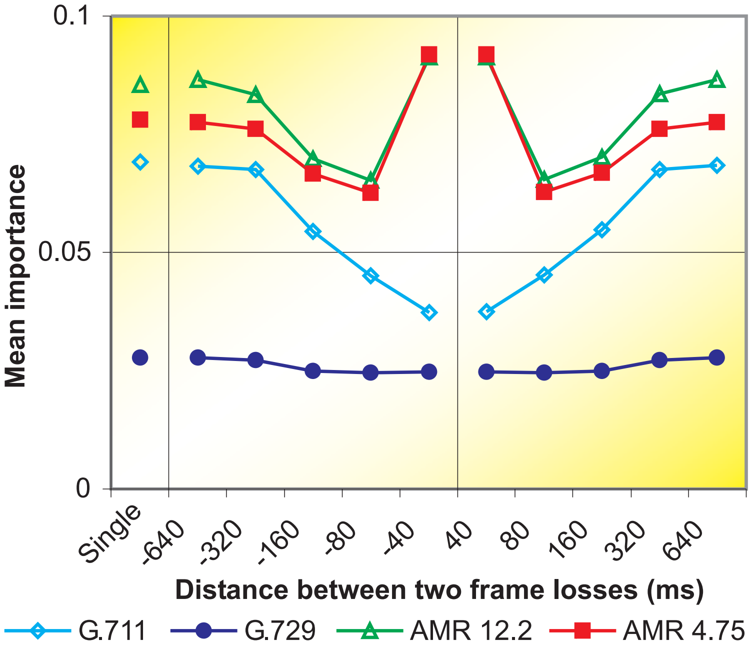


Figure 6.6.: The importance of the second lost frame is displayed in relation to its distance from the first lost frame.
The mean importance of a single lost frame marked with “Single”.

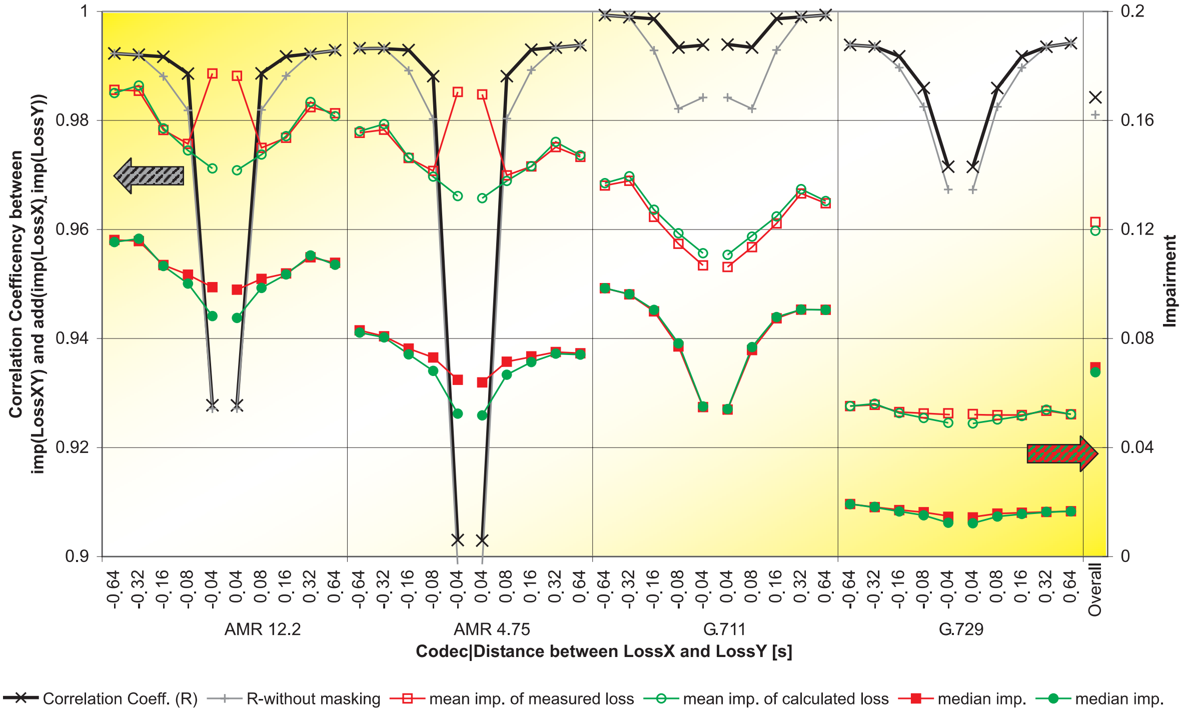


Figure 6.7.: Median and mean impairment due to two frames losses. Estimated with PESQ displaying $Imp\left(s, c, \{l_1, l_2\}\right)$ and with our model: $add\left(Imp\left(s, c, l_1\right),\ Imp\left(s, c, l_2\right),\ t_{width}\right)$. We also show the cross correlation between PESQ ratings and the rating of our model (R value). The R-without-masking compares PESQ with the long-term only aggregation function.

the double loss case and the sum of both single loss cases is calculated and displayed. The correlation for a distance of >320 ms is about R>0.98 and drops to a minimum of R=0.78 at a distance of 40 ms. This effect can be explained with concealment and error propagation effects that are not modeled in our model.

6.3.3. Studying bursty packet losses

In the next experiment we study the effect of bursty packet loss. We dropped one block of continuous speech frames within a sample length of 8 seconds. The duration of the complete block was between 10 to 80 ms (in Figure 6.8 the red lines marked with a square). Also, we used our model to add the importance of the corresponding single frame loss (the green lines marked with a circle). To calculate the importance of the burst loss we use Equation 6.12 with N being the number of continuously lost frames, pos the position of the first lost frame,

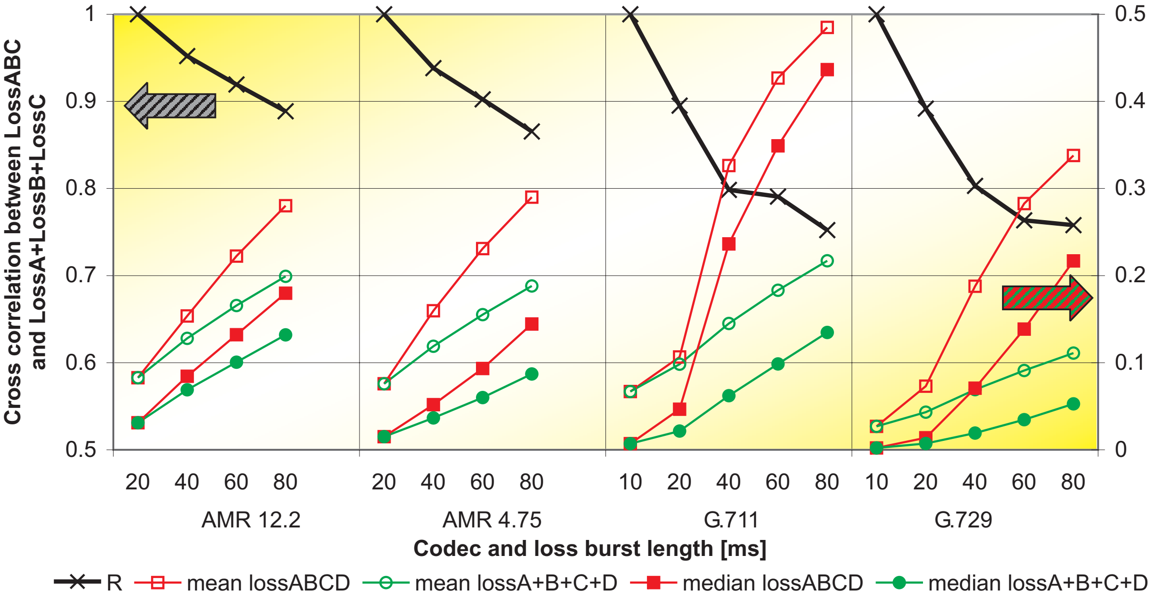


Figure 6.8.: Importance of a block of frame losses.
Red: PESQ, green: our approach, black lines: cross correlation R.

and $Imp^*_{pos}$the importance of a frame loss at position pos.

$$Imp^*(N, pos) = \begin{cases} Imp^*_{pos} & if\ N = 1 \\ add\left(Imp^*(N-1, pos), Imp^*_{pos+N-1}, 0\right) & if\ N > 1 \end{cases} \tag{6.12}$$

The correlation (R) between PESQ and our model is displayed with black lines. The longer the loss burst, the worse the cross correlation. Our model underestimates the impact of bursty losses compared to PESQ. This modelling is in line with the indications that PESQ displays an obvious sensitivity to bursty losses and judges them worse than humans [201]. Nevertheless, the effects of concealment performance after previous loss and error propagation tend to increase the impairment of bursty losses.

6.3.4. Discussion

The distortion due to frames loss increases linearly if the losses are distant. But which else effects have to be considered in cases of bursty loss as already mentioned in Section 6.2? Both, PESQ as well as our model take the psychoacoustic temporal masking effect into account. We do not know whether cognitive effects have to be included as such cognitive effects have not been published.

Another point that has not been discussed in this chapter is due to the phone variability in the temporal domain. The importance of a speech frame is only slightly correlated to the importance of the following frames as spoken language changes rather slow over time. Thus, the phone characteristics do not vary quickly. This effect has an impact on the variability of packet loss bursts. The importance of loss bursts tend to have a lower variability as it is highly likely that both important and non important speech frames occur in the same loss burst. However, before this effect can be fully understood, it is required to model the temporal progression of speech. Up to now, only a model with the on-off characteristic of speech has been developed [42]. A similar model for the temporal progression of frame importance, which would enable such investigations, is still lacking.

6.4. Conclusion

This chapter describes the impact of speech frames loss by considering their temporal relation. It is based on the concept of the *importance of speech frames* and models psychoacoustic aggregation behaviour over time. Thus, our model covers an important aspect in the relation between speech frame losses and speech quality. Our model shows a high prediction accuracy for many loss patterns when compared to PESQ. Additionally, our time aggregation function has a low complexity. The results contribute to research and standardization:

First, they enable researchers developing communication protocols to model the impact of frame loss with high accuracy. For example, it can be applied for algorithms that reduce the burstiness of frame losses.

Second, this work provides also feedback to the developers of PESQ or similar algorithms, as it explains why PESQ does not require temporal masking: It was already included.

Third, it identifies weaknesses of frame loss concealment algorithms (e.g. AMR). Last but not least, our work is directly intended for the standardization process of ITU-T P.VTQ, as it can been seen as an alternative or complementary algorithm to the ITU's E-Model, Telchemy's VQmon and Psytechnics' psyVOIP algorithms [105, 171, 203], which relate VoIP packet loss and delay to service quality.

6.5. Outlook

Before our approach can be fully applied, two issues have to be addressed in future research:

First, we compare the performance of our aggregation algorithm to the same PESQ algorithm, which we used to derive and remodel our algorithm. We achieve a high prediction performance. However, it is still an open issue how well our algorithm performs, if it is compared to subjective listening-only test results. The verification with databases containing subjective results is a subject of future studies.

Second, further studies are required to see how our metric scales at high loss rates. Definitely, the effects of concealment and error propagation play an important role if losses are frequent or bursty and need to be modelled.

7. Real-Time Classification of Speech Frames

Losing one speech frame impairs the speech quality of the transmitted signal even if the receiver tries to conceal the loss. This impairment is due to the imperfect packet loss concealment (PLC) and also due to error propagation (EP), originating from desynchronisation of the decoder's internal state. We developed a measurement method to determine the effect of the imperfect PLC and the temporal progression of the error propagation. The results show the trade-off between algorithmic delay and the accuracy of frame importance determination.

Knowing the importance of speech frames can increase the transmission performance of telephone calls. Speech intelligibility is maintained during wireless telephone calls experiencing high loss rates if only negligible active and silent frames are lost. But this gain can only be achieved if the packets' importance is determined ahead of time at high accuracy: An algorithm needs to look ahead 20-40 ms in order to calculate a frame's importance precisely.

We also benchmarked related work in this field of research including published speech frame classification methods requiring low algorithmic delay and complexity. Finally, a real-time algorithm is presented that outperforms the previously presented approaches in terms of prediction accuracy but at the cost of higher computational complexity.

7.1. Introduction

Packet loss significantly decreases the quality of voice communications. If a speech frame is lost, the receiver tries to extrapolate the last successful received frame to limit the impact of the lost frame. Such algorithms are known as packet loss concealment. Nowadays, they are often standardized and part of the decoder[1]. A lost frame causes the current speech period to become distorted as the receiver's PLC cannot fully reconstruct the lost frame. Thus, the concealed frame differs from the sent frame and hence introduces a *loss distortion.*

Low-rate speech coders that transmit only signal differences suffer from an additional effect: If a frame is lost, the decoder becomes desynchronized [178]. If the internal state of the decoder does not match the encoder's state, the decoding of the following frames is affected and an additional distortion is introduced. We refer to this effect as *error propagation.* This effect is well known from digital, compressed TV and video transmissions. A transmission error

[1] In this thesis we assume the presence of such standardized algorithms. In reality, some implementations lack PLC algorithms or implement algorithms that even outperform the standard recommendations (e.g. [209]). Then, our studies would have to be repeated with the implementation of the particular PLC algorithm. However, for the sake of generality we do not consider proprietary solutions.

causes the video signal to be distorted for a long period that can even last multiple video frames.

In Chapter 5 we presented a method of how to determine the impact of an individual frame's loss – called the *importance of a frame*. Here we extend this method to quantify the impact of the loss distortion and temporal progression of error propagation by studying common narrow-band speech codecs. Our results show that the frame following the loss contains the largest amount of the error propagation.

In published literature, four algorithms have been presented which classify VoIP packets in real-time:

- The first is the voice activity detection (VAD), which identifies silence periods. We will refer to it as *discontinuous transmission* (DTX).
- The second approach uses the *voicing* differentiation. Voiced sounds have usually a higher energy than unvoiced sound.
- The third approach has been introduce by De Martin and is called Source-Driven Packet Marking [48].
- The fourth that has been presented by Sanneck is referred to as SPB-DiffMark [184].

Using our measurement procedure, we evaluate and compare these real-time packet classification algorithms and show that their prediction performance is rather low.

To overcome the deficits of the published algorithms, we developed a novel real-time packet classification algorithm that is based on the metric importance. We reduce its computational complexity by limited the sample length considering only a short period before the lost segment. Also, we use only one MOS measurement (instead of two in the off-line approach). Finally, we have developed a low complexity version of PESQ called *PESQlight*, which considers the special properties of the G.711 codec to lower its computational complexity. The predictions of this algorithm show a higher correlation to the results of the off-line approach, indicating a superior prediction performance.

This chapter is structured as follows: In Section 7.2 we analyze why packet loss distortions occur. Next, we present our measurements on quantifying the amount of error propagation. Then, we assess the previously published real-time frame classification algorithms. In Section 7.5 we present a real-time classification algorithm that is based on the off-line measurement procedure described in Chapter 5. Finally, we conclude and give an outlook for further research issues.

7.2. Real-time packet classification

To control the transmission of speech frames, their importance should be known at transmission time. For example, in addition to the encoding of speech the sender could calculate

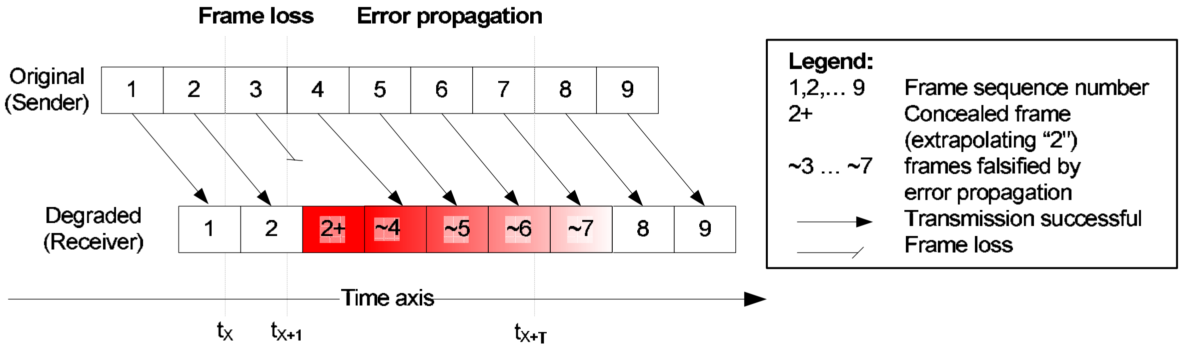


Figure 7.1.: Consequences of losing a frame.

the importance of each speech frame. This leads to the question, is it possible to predict the importance of speech frames at transmission time?

In general, the consequences of packet loss can be split into two effects (Figure 7.1):

- First, the lost frame is concealed at the receiver, which causes a distortion if the concealment does not perfectly predict the frames content. In the illustration this refers to frame 3 (transmitted) and frame 2+ (concealed). The encoder knows the original and degraded speech segment. It can also predict the behaviour of the decoder in case of loss, as the decoder's concealment algorithm is known (since it is standardized). In principle, the encoder can therefore calculate the impact of imperfect concealment.

- The second effect of packet loss is due to error propagation. After a frame loss the internal state of the concealment algorithm is desynchronized. In Figure 7.2, we display how long it takes until synchronisation of the decoder is achieved. We measure desynchronisation lengths for the ITU G.729 coding, which last up to 650 ms. The impact of error propagation cannot be known at the time of transmission because the length of error propagation depends on the following speech content. In case of interactive telephony the following speech has not yet been spoken. Thus, predicting the importance of a speech frame at run-time will always be falsified by the effect of error propagation.

To demonstrate the impact of imperfect packet loss concealment and error propagation we plotted the speech signals of a sample segment in Figures 7.3 and 7.4 for different encoding schemes. Beside the original sample, the figures also contain the encoded/decoded (=degraded) signal, the encoded/lost/decoded/concealed signal, and the difference between those signals. Also, the figures contain the PESQ MOS values to quantify the perceptual impact of coding and concealment degradation.

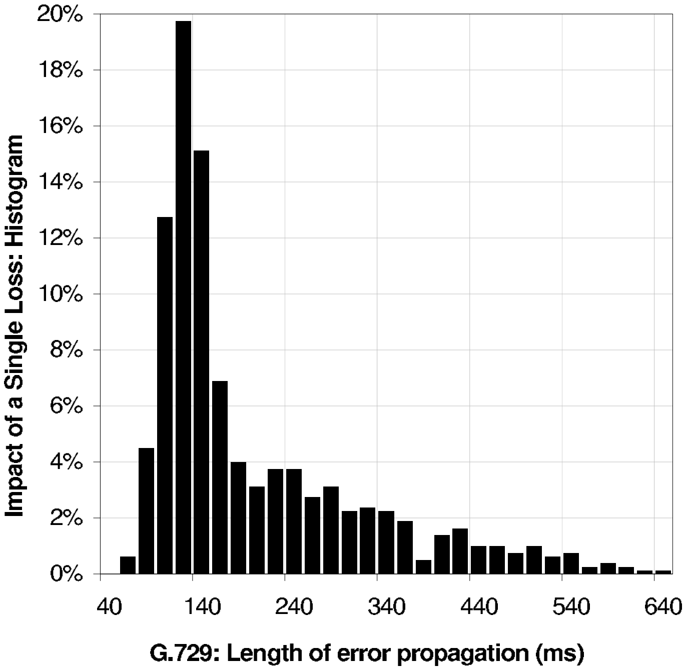


Figure 7.2.: Histogram of error propagation lengths in case of loss of one G.729 frame. We measure the time until the internal state of the G.729 decoder matches the non-loss state again. The decoders' post-filter is ignored as it does not synchronise again.

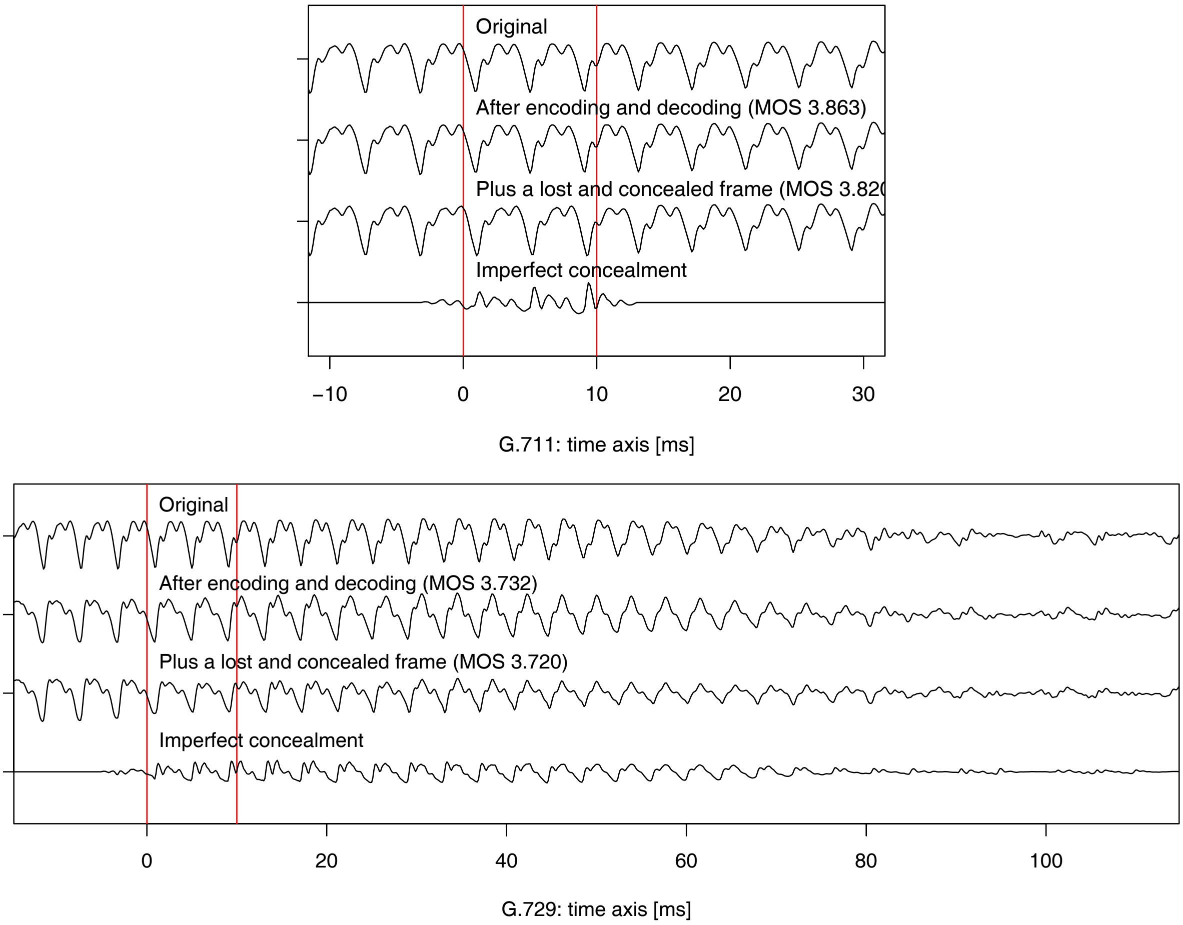


Figure 7.3.: Speech signals before and after decoding, after loss concealment, and the difference between the decoded and concealed signals.
The two vertical lines define the length of the frame loss.

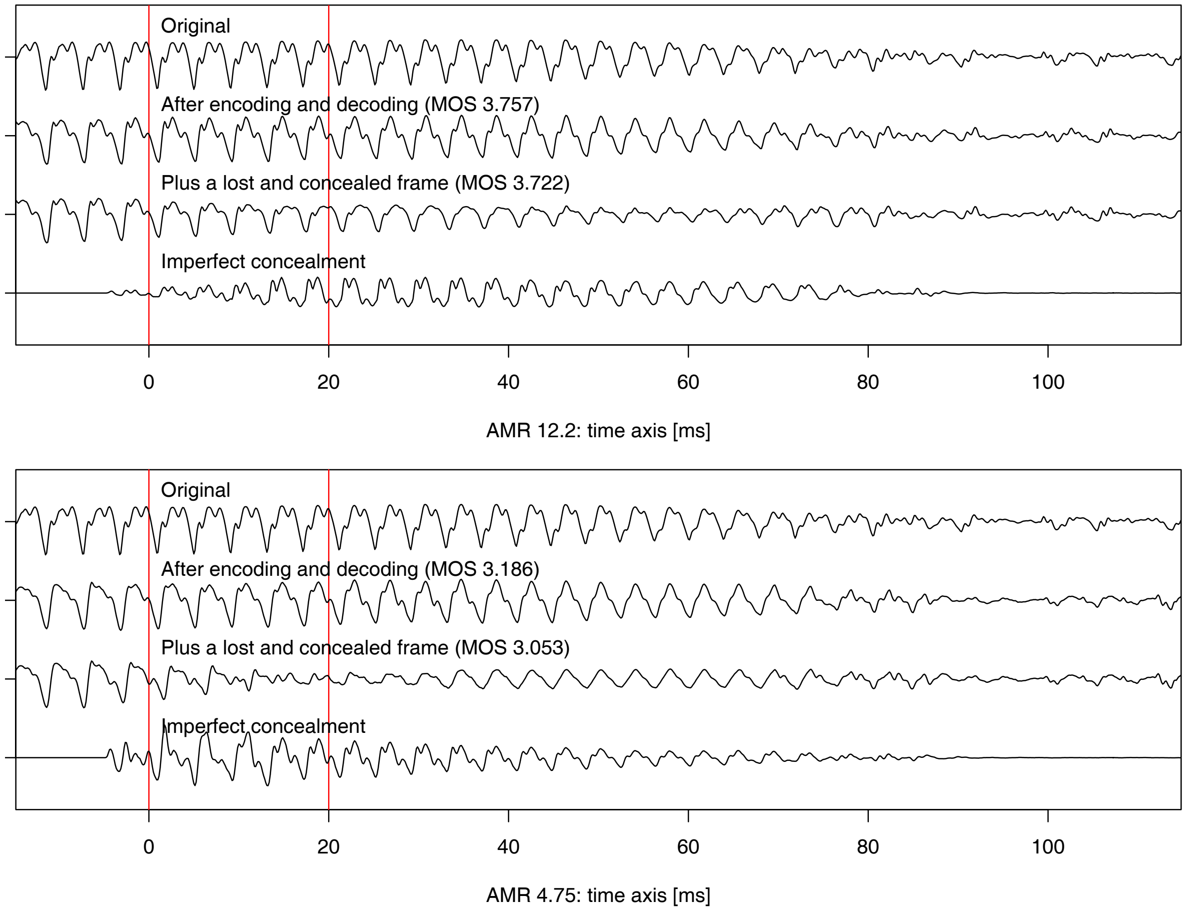


Figure 7.4.: As Figure 7.3 but displaying the AMR codec.

7.3. Quantifying error propagation

The aim of this section is to quantify the imperfect concealment and error propagation caused by a single frame loss. The question arises how should the effects be measured? The speech sample could be split into two parts. The first part contains the content until the end of the concealed frame (e.g. frame 1, 2, 2+). The next part contains the remaining content (e.g. frame ˜4 to 8). The position of the split is exactly after concealing and decoding the lost frames. Thus, the effect of concealment and the effect of error propagation are separated into two samples. For both samples the degradation can be measured with PESQ and compared with to the corresponding samples that do not contain any frame loss. This method is problematic due to two reasons.

- First, PESQ judges the speech quality largely different if the sample content differs. Thus, splitting the sample and thus changing the sample's length introduces a source of error. Instead, the sample content must not be changed.
- Second, a hard split between two samples introduces an additional clicking sound, which falsifies the results.

Therefore, we developed the following measurement procedure (see Figure 7.5). We generate two samples containing first the degraded sample without loss and second, the degraded sample with one frame loss. Then, we mix both samples to produce new samples: We crossfade just after the lost frame (right vertical line in Figure 7.5). The crossfading function is a cosine curve. Then, two new samples are produced. The first called "left" contains the concealment frame and the second called "right" contains the error propagation. The speech quality of those samples is then measured with PESQ.

This algorithm leads to another question: How long should this crossfading period be? For one frame loss we conducted measurements with varying crossfading lengths (see Figure 7.6). The black lines represent the speech quality considering imperfect concealment and error propagation. If the crossfading is done in less than 4 ms, it introduces an addition distortion that lowers the speech quality. However, if the crossfading is too slow, the short effect of a single frame loss is smeared over the left and right samples. Thus, we will use a crossfading length of 4 ms in the following work.

The impact of frame loss has been measured several million times for different frame contents and codecs as described in Chapter 5 and [89,93]. Additionally, we split not only exactly after the lost frame (position 0 ms) but also at positions 10 to 320 ms after the lost frame. Thus, we can observe the temporal progression of the error propagation.

For each sample we calculate the importance values. In Figure 7.7, we display the importance value of the left and right parts of the loss distortion. Actually, we choose to display the 75% percentile as it is close to the median importance of all active speech frames. In addition,

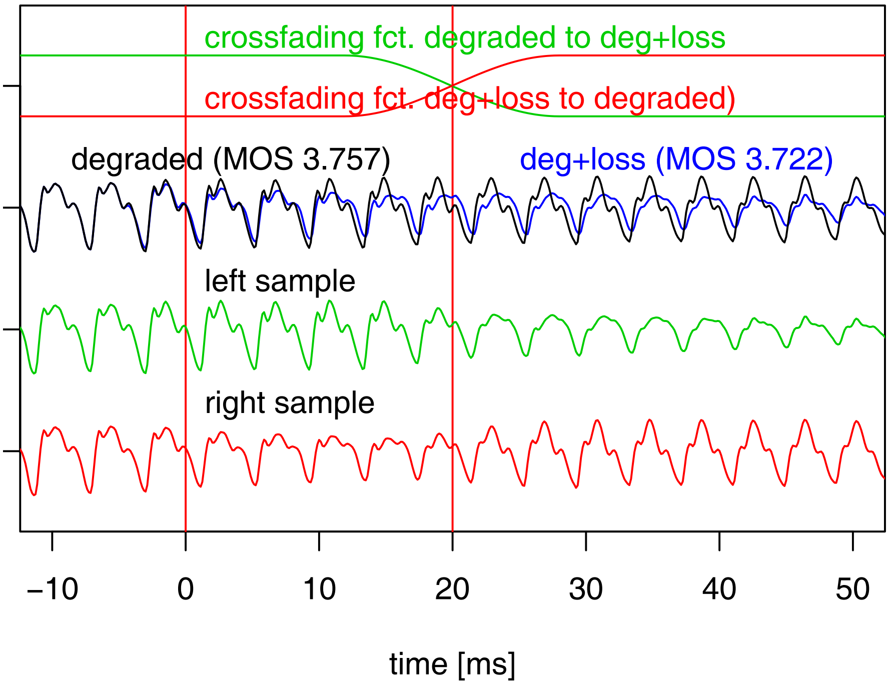


Figure 7.5.: Splitting the imperfect concealment and error propagation into two different speech samples.
The position of the lost frame is marked with two vertical, red lines. Two degraded samples are generated, with loss (blue) and with-out loss (black). Then, to get the impact of PLC we crossfade from the loss to the no-loss sample to produce a new speech sample called “left”. Similar, to get the impact of EP we crossfade from the no-loss to loss sample to produce a new speech sample called “right”.

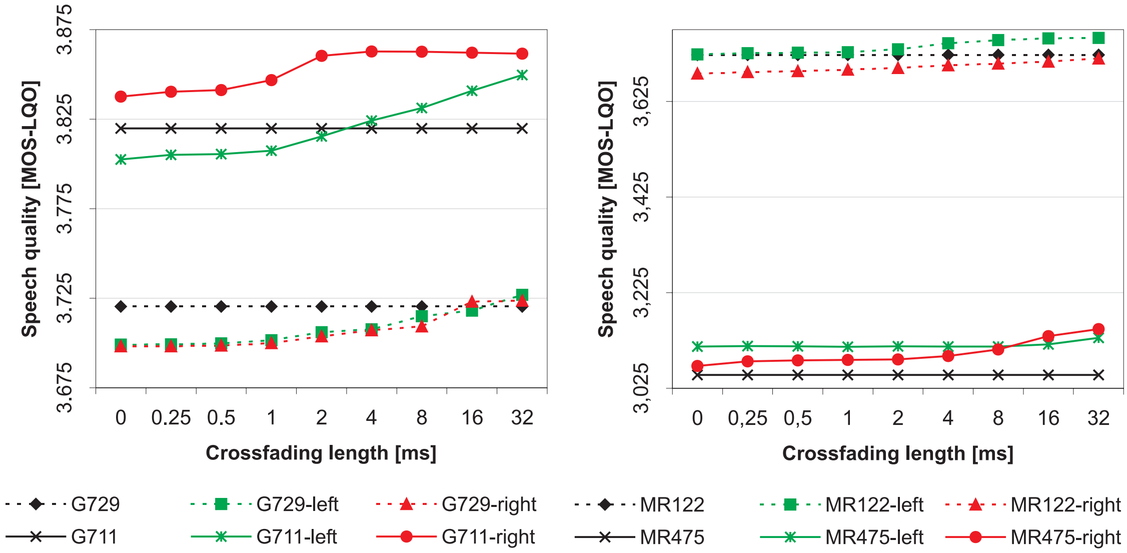


Figure 7.6.: Impact of crossfading length on speech quality.

the sum of the left and right importance values are displayed since the importance metric is to some extent additive (refer to Chapter 6). The first graph containing G.729 values shows that this codec has a high amount of error propagation and it takes approximately 80 ms until this effect disappears. The next graph using G.711 is to demonstrate the quality of our measurement procedure as in case of G.711 the error propagation is fixed to a length of at most 3.75 ms (refer to Section 3.2). It shows that our measurement procedure does not split perfectly both distortion effects, but has an inaccuracy of 0–10 ms. Last, the values for AMR coding are shown. The amount of error propagation is small and disappears after 20–40 ms.

Coming back to the main question of this chapter: How well can the importance of a speech frame can be predicted in real-time? As a performance metric we will calculate the Pearson's cross correlation coefficient of the offline, reference importance values and the left, right, and both (left+right) importance values.

The cross correlation coefficient is often used to compare speech quality scores of human and instrumental predictions (e.g. PESQ). A value of R=1 would a perfect match between both score sets, whereas R=0 means to no correlation at all. A positive behaviour of cross correlation is it is not influenced by linear scaling or adding an offset: Any linear regression applied to sets of measurement data does not change the value of R at all.

In Figure 7.8 we display the cross correlation to compare the importance value sets of the reference (offline), left, right, and both measurements. If only the imperfect concealment is considered to calculate importance values, the performance for G.729 coding is R=0.72, for G.711: R=0.91, and for AMR: R=0.73. If in addition the next frame after the concealed frame is considered, the performance increases to G.729: R=0.92, G.711: R=1.00, and AMR:

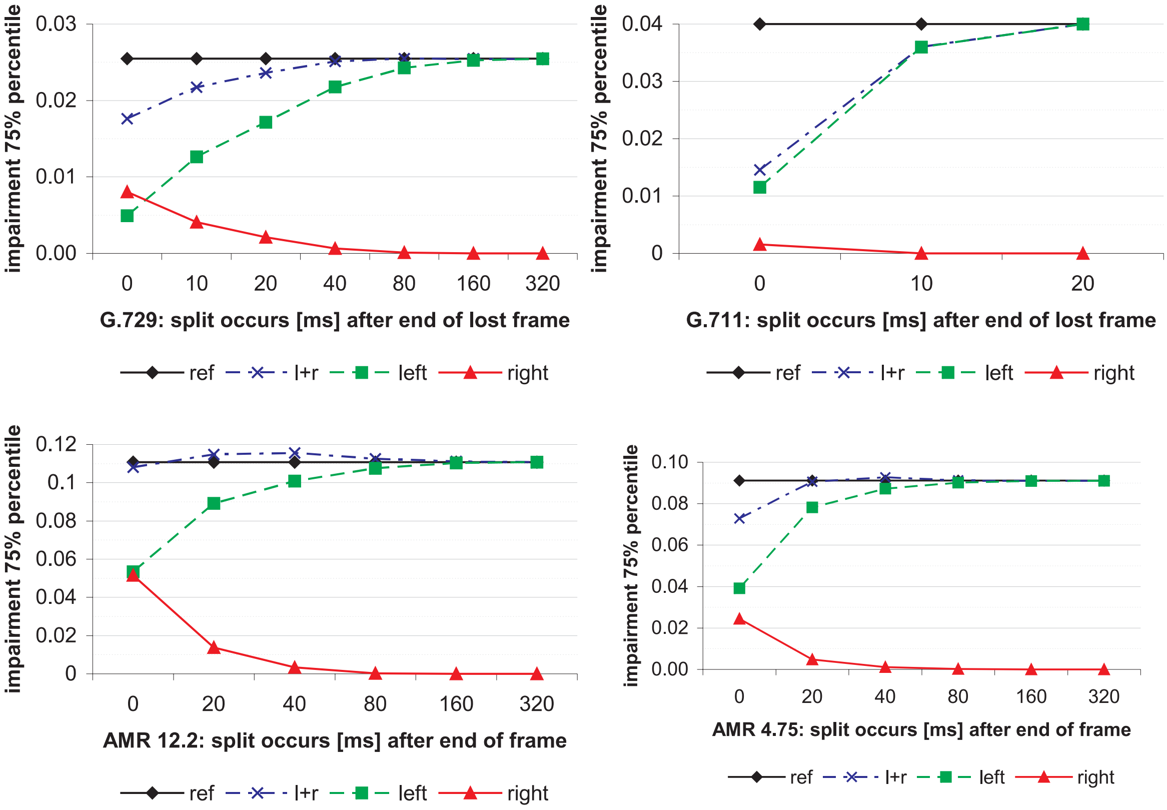


Figure 7.7.: Temporal impact of error propagation on the importance of frame losses.

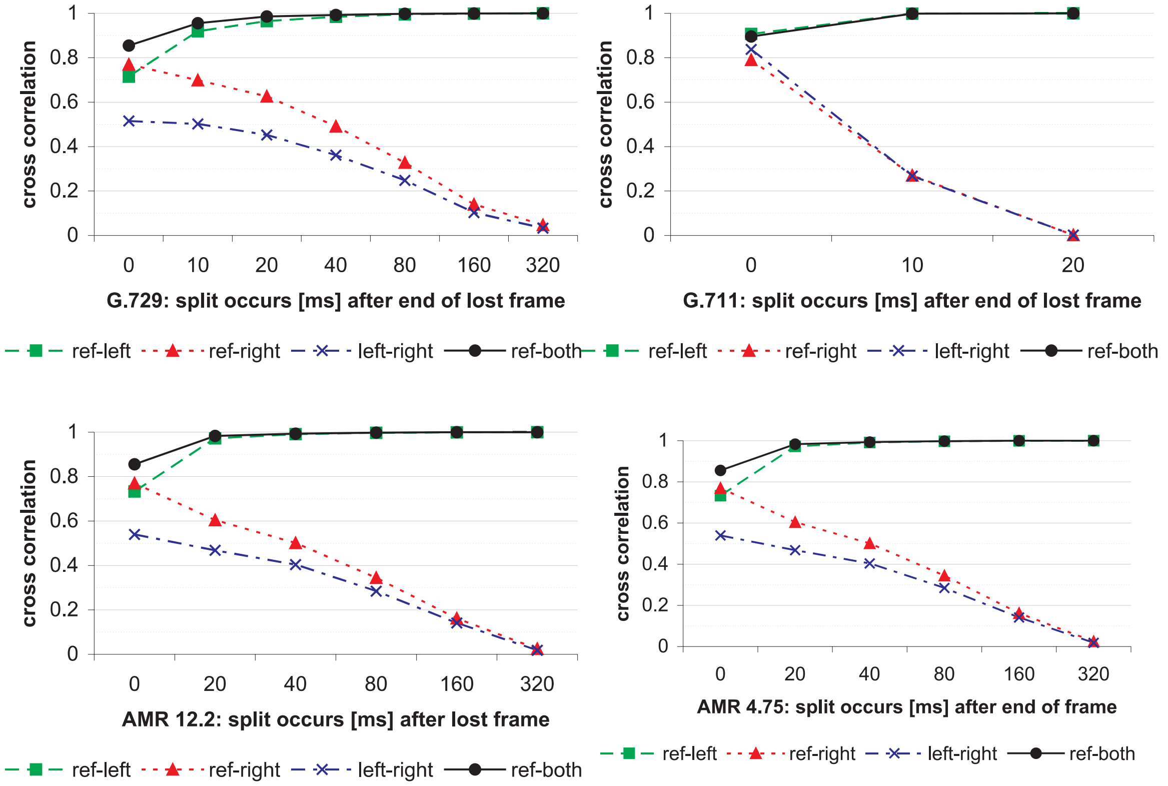


Figure 7.8.: How well can the frame importance be predicted if some X milliseconds of the following speech is considered in addition to concealment effect? (This result is displayed in the "ref-left" line.)

R=0.97. Given the precision of our measurement procedure (R=0.94, refer to Section 4.2), the later results are almost perfect.

7.4. Benchmarking real-time classification algorithms

In this section, we benchmark algorithms that classify speech frames in real time for dropping purposes. We use our measurement method as a reference to benchmark the packet classification algorithms DTX, voicing, SPB-DiffMark, and Source-Driven Packet Marking.

For each classification scheme, we calculate the frame importance distribution for each packet marking, calculate the mean importance for each marking class, and the correlation between the packet classification scheme and the reference values (displayed in Figure 7.9). To allow calculating of the correlation coefficient, we describe the packet classification by a mapping function: For example, in case of the DTX algorithm, silent (off) frames receive a marking of 0 whereas active (on) frames receive a 1.

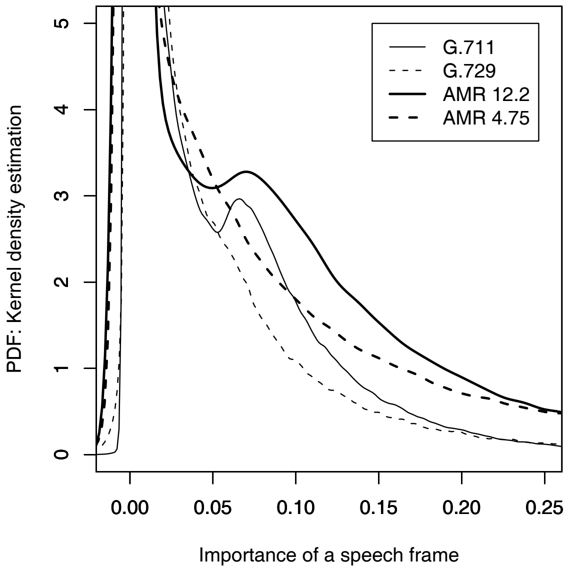


Figure 7.9.: Distribution of the importance of frames.
A kernel density estimation is similar to a histogram. The higher, the more values are expected to a given value.

7.4.1. Discontinuous transmission (DTX)

If frames are lost during silence, the impairment is hardly audible (Figure 7.10 and 7.11. In Table 7.1, the importance of mean active and silent frames are listed. In general, silent frames are fifty times less important than active ones. Thus, the DTX algorithm performs well and is a good indicator of unimportant frames.

Table 7.1.: Prediction performance of DTX algorithms.

Algorithm	**Mean importance (active frames)**	**Mean importance (silent frames)**	**Percentage of active frames**	**Correlation coefficient (R)**
G.711 with G.729 DTX	0.04425791	1.050064e-03	66.8%	0.3841000
G.729 DTX	0.03897468	1.143842e-03	66.8%	0.3023400
AMR 12.2 DTX	0.11788374	2.819836e-03	68.4%	0.4069918
AMR 4.75 DTX	0.10825452	2.109359e-03	68.4%	0.3625117

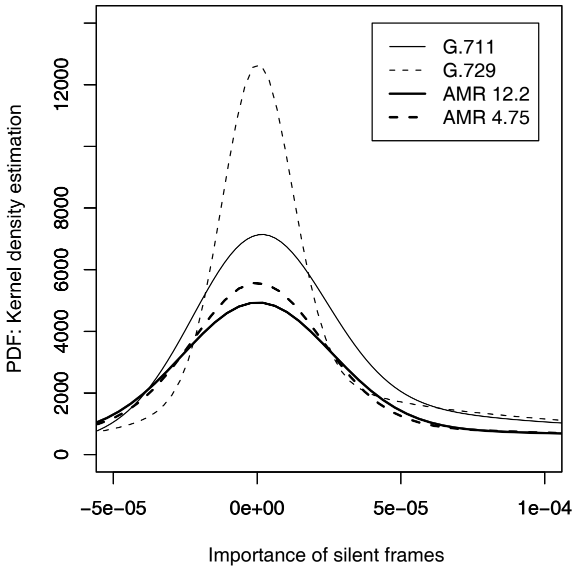
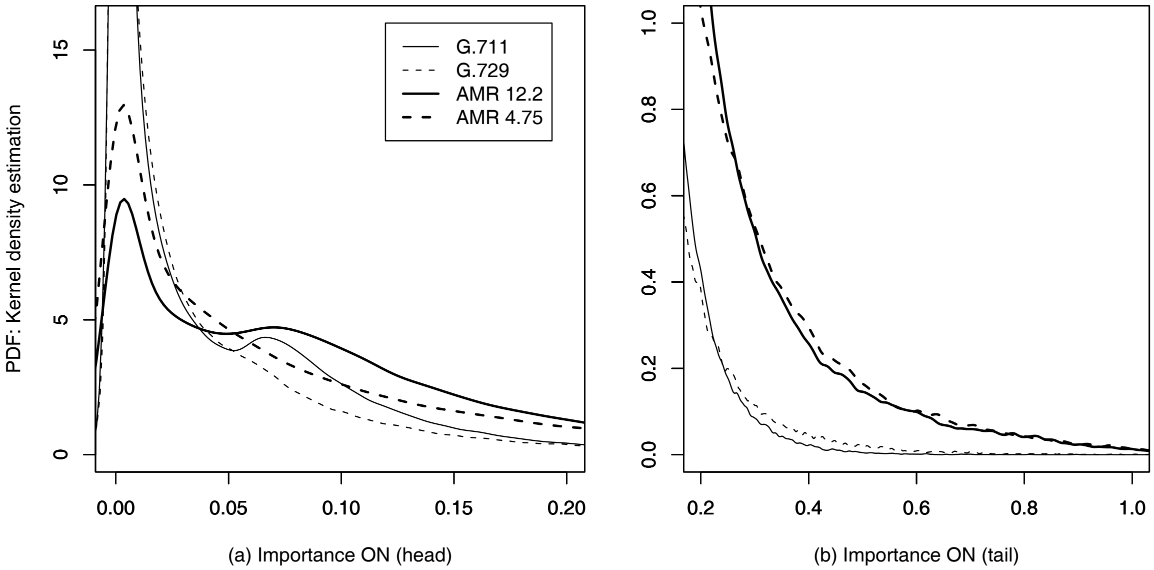


Figure 7.10.: Distribution of the importance of silent frames.



(a) Importance ON (head)

(b) Importance ON (tail)

Figure 7.11.: Distribution of the importance of active frames.

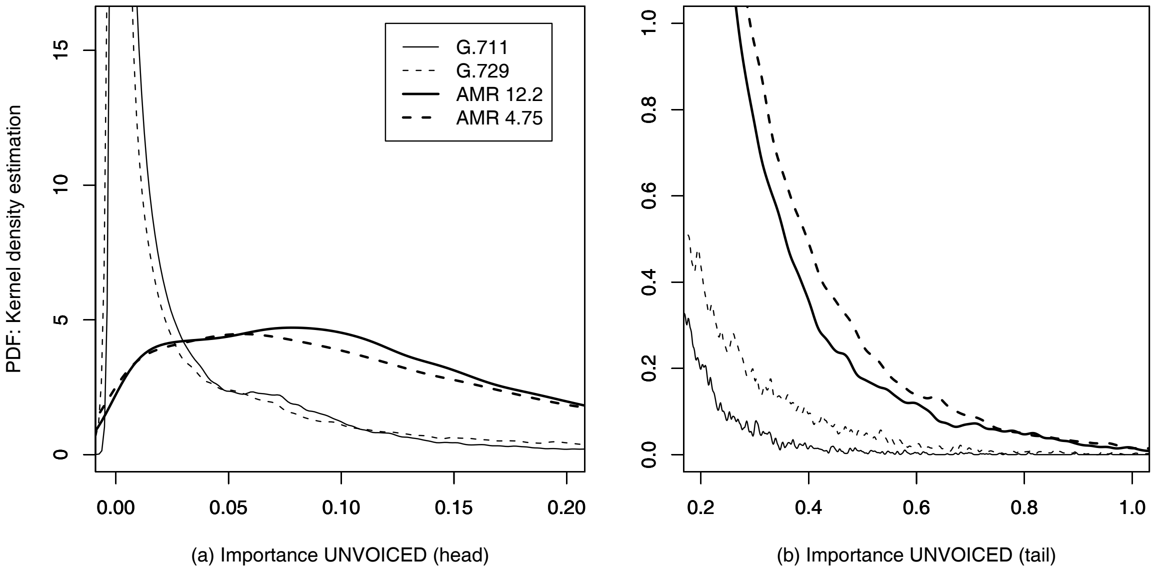


Figure 7.12.: Distribution of the importance of unvoiced frames.

Table 7.2.: Prediction performance of voicing algorithms (based on 1.327.276 values).

Algorithm	Mean importance (voiced)	Mean importance (unvoiced)	Percentage of voiced frames	Correlation: importance vs. voicing
G.729-voicing on G.711, 10ms	0.05057883	0.02513223	75.16%	0.1843085311
G.729-voicing on G.729, 10ms	0.03899152	0.03892372	75.16%	0.0004285957
AMR-voicing on AMR 12.2, 20ms	0.15190373	0.09593398	39.22%	0.1898486948
AMR-voicing on AMR 4.75, 20ms	0.16456056	0.07192577	39.22%	0.2967894446

7.4.2. Voiced and unvoiced speech frames

In this section, we use the voicing information gathered from the decoders and look, how it influences the importance of a frame. Figures 7.12 and 7.13 display the distribution of the importance for both unvoiced and voiced sound (all results are displayed for active frames only). Table 7.2 contains the mean importance for unvoiced and voiced sounds. In general, voiced frames are more or at least equally important than the unvoiced frames. Interestingly, in case of the G.729 codec, voiced frames are as important as unvoiced ones.

7.4.3. Source-Driven Packet Marking

We reimplemented De Martin's algorithm based on his publication [48], as the original implementation is not available anymore. One should note that our algorithm [89] might differ from De Martin's implementation. However, the author has confirmed that our implementa-

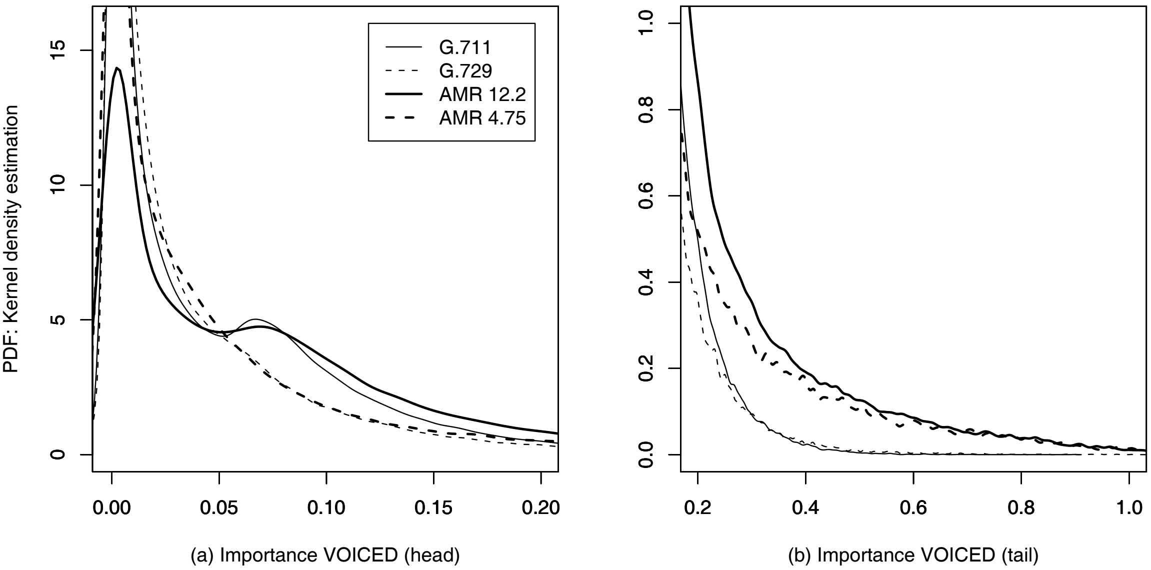


Figure 7.13.: Distribution of the importance of voiced frames.

tion complies with his algorithms, but does not seem to mark all the low-energy packets as best-effort.

The Source-Driven Packet Marking applies four criteria to mark packets as premium. In Table 7.3, the importance, frequency and the correlation of the particular marking criteria are listed. The prediction performance of De Martin's algorithm is high for unvoiced packets. For voiced packets, considering the difference in the codebook indices and gains is less beneficial, but the spectral distance is a good predictor for voiced frames, too. If the selection criteria for important packets are combined, the Source-Driven Packet Marking algorithm predicts the importance of speech frame with an accuracy of R=0.19. As described in the paper, about 20% of all frames are marked.

One drawback of De Martins algorithm is that only just the quality impairment of the current frame is analyzed. Any error propagation is not taken into account.

7.4.4. SPB-DiffMark

The implementation of SPB-DiffMark is publicly available. It uses a modified decoder to obtain the information whether a frame is voiced or unvoiced. The original implementation does not treat packets differently if they are silent. Thus, some packets are identified as voiced, even though the voicing decision is based on a frame that has a very low energy.

In Figure 7.14 we use a function which equals 1 for the last unvoiced frame before a following voiced frame. All other packets are marked as 0. We calculate the cross correlation between this function and the frame importance. One can see that the frames shortly before an unvoiced and voiced transition are more important (except for the G.711 coding). Similar

Table 7.3.: De Martin: Importance of marked frames, listed for each selection criterion. Importance of unmarked and marked speech frames.

Frames	Correlation (R)		
	on	voiced	unvoiced
all	1.00000000	1.00000000	1.00000000
marked	0.19476290	0.09727415	0.46871403
Fixed cookbook gain	0.34117405	0.26135819	0.43980751
Index difference	0.09628635	0.09242958	0.03449131
Adaptive cookbook gain	0.07022292	0.04134183	0.06553747
Spectral density	0.19380020	0.16127852	0.22975768

behaviour can be shown for voiced-unvoiced transitions.

Next, we analyse the marking performance of the SPB-DiffMark algorithm. In Figure 7.15 the mean importance of the marked packets is depicted, depending on the values of N. Figure 7.16 shows the correlation between the marking function and the importance. We display this function for all and for active frames. Sanneck did not describe how to mark silence packets. This function has a maximum at N=3 with a correlation of R=0.10. In our opinion, the drawback of Sanneck's marking algorithm is that the unvoiced/voiced transition is detected too late. Also, marking packets with -1 has no beneficial value.

7.5. Real-time classification of μ-law frames

In this section we present an algorithm, which classifies speech frames in real-time. We simulate the impact of packet loss already at the sender side in an approach that is called analysis-by-synthesis [46] and is similar to the algorithms of De Martin [48]. Then, the packet can be marked with its importance value for further processing steps in the access or backbone network. The approach is based on the offline measurement algorithm for frame importances as given in Chapter 5. To reduce the complexity, the following three ideas are the applied.

1. The off-line classification presented in Chapter 5 uses two PESQ MOS calculations as it compare the original with the degraded to calculate $MOS(s,c)$ and the original with the degraded plus concealed to get $MOS(s,c,l)$. In our algorithm, we just compare the degraded with the degraded plus concealed samples (Figure 7.17).

2. We reduce the context information, including the pre-leading and following segments of speech, as long as the prediction performance of our approach is not disrupted.

3. Because we know the behaviour of the codec and the packet loss concealment, we can simplify the PESQ algorithm and remove unnecessary parts of its processing.

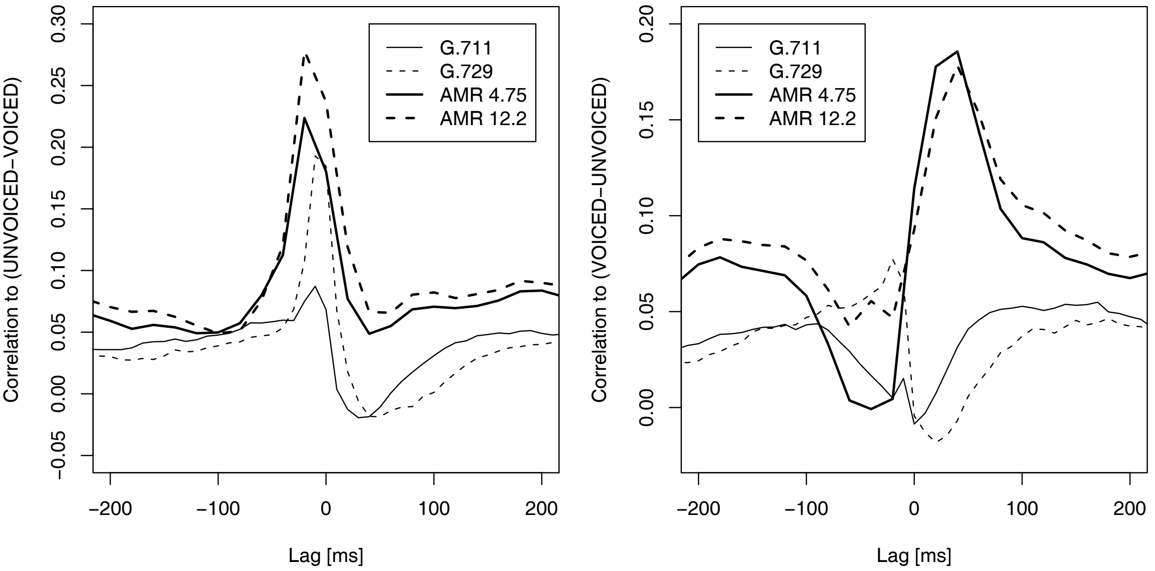


Figure 7.14.: Impact of the unvoiced-voiced transition (left) and voiced-unvoiced transition. Shortly before an unvoiced-voiced transition frames are important and show a high correlation coefficient (left figure). Shortly after a voiced-unvoiced transition frames are important, too (right figure).

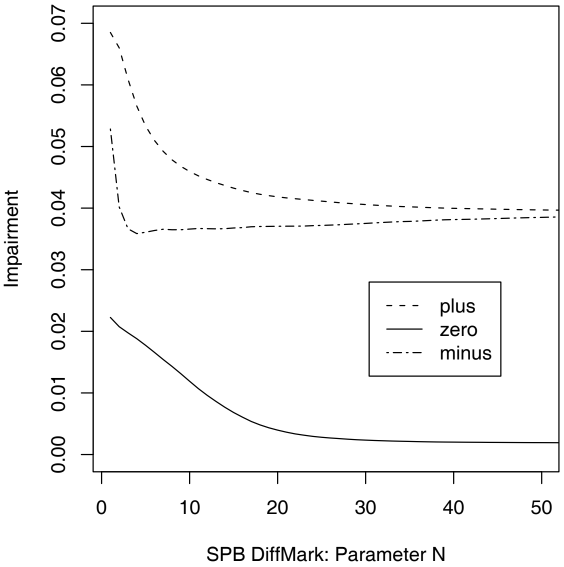


Figure 7.15.: Importance of speech frames marked with –1, 0 or +1.
The results depend on the parameter N, which controls the marking of frame following the unvoiced/voiced transition.

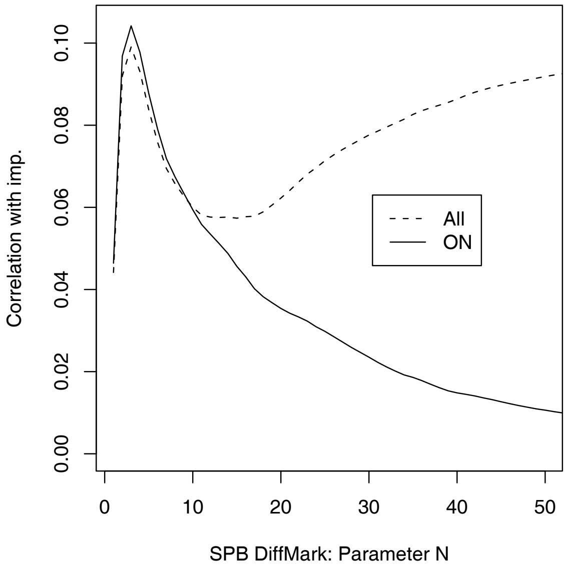


Figure 7.16.: Prediction performance for all or active (ON) frames, depending on the parameter N.

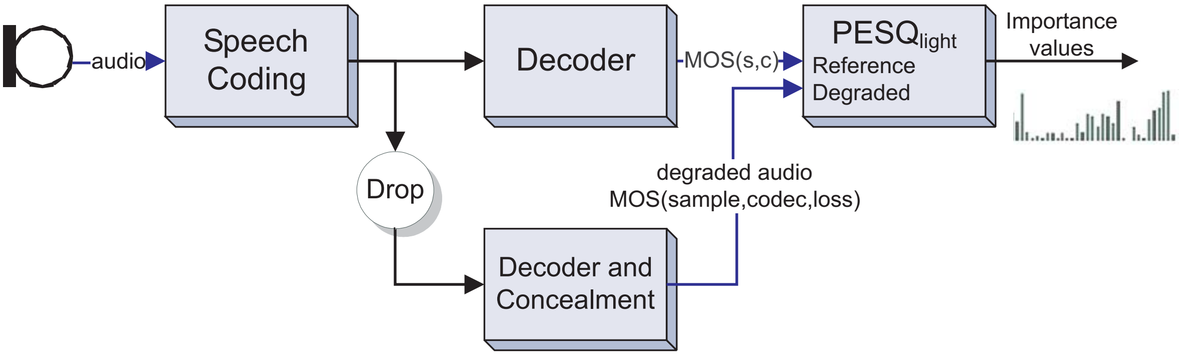


Figure 7.17.: Schematic of a sender-side calculation of the importance of a VoIP packet.

7.5.1. Behaviour of the concealment algorithm

The algorithm that calculates the importance of speech frames uses a sample rate of 8 kHz, the coding μ-law following the ITU-T recommendation G.711 [98], and the packet loss concealment algorithms [104] given in G.711 Appendix I. The concealment work with a frame size of 10 ms and to calculate the importance we also drop a frame of 10 ms. Based on the description of the G.711 Appendix I packet loss concealment (refer to Section 3.2) we state the following important insights to simplify the implementation of PESQlight:

- The speech signal at the decoder is delayed for 3.75ms (30 samples).
  Before calculating the importance, this delay must be removed to synchronize the original and degraded signals.
- No loudness change.
  The power of the degraded speech signal does not differ to the original speech signal.
- The disturbance after the lost frame is limited to one quarter of the pitch period, which is less than 3.75 ms.
- Within this work, we study only single frame losses, thus for a sender-side calculation of importance values, it is not required to consider the effect of the multiple frame losses.

7.5.2. Description of PESQlight

A large challenge of this work was to understand the behaviour of the reference code of PESQ given in the ITU-T standard. By a detailed study and continual comparison with the textual description of the algorithm, the parts of the source code have been assigned to the mathematical model description. Afterwards, the following features have been identified, which can be simplified (or even omitted).

Constant length of the test samples: The original PESQ does not assume constant length of the samples. In addition it takes into consideration that original and transferred sample may differ in length. Both are irrelevant in our case, since the both lengths of the original and degraded sample are fixed and equal.

No delay: Because the delay caused by the encoding and decoding is known and removed, PESQlight does not have to align the degraded sample in time.

Known loudness: With the unmodified PESQ algorithm a signal power adaptation takes place to match the levels of original and degraded versions. In addition, both signals are raised or lowered to a normalized power level. To achieve this, the mean loudness of both signals has to be calculated.

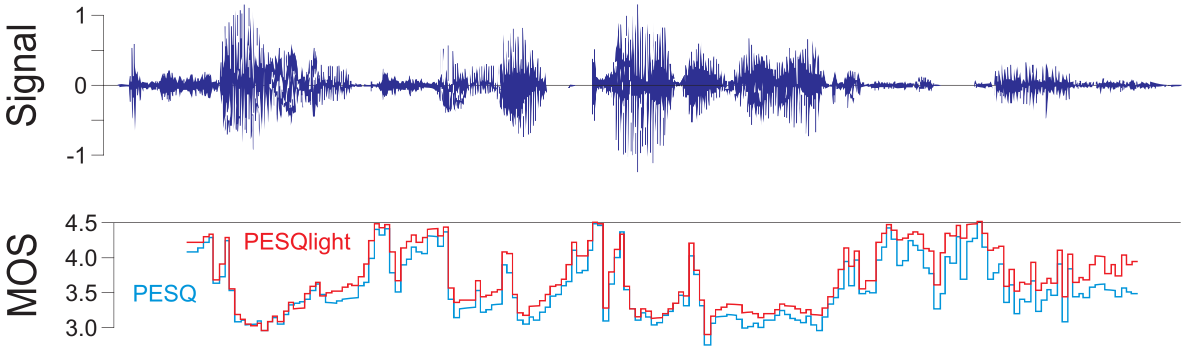


Figure 7.18.: Visualizing PESQ and PESQlight.

In this case PESQlight knows that both versions have the same loudness. Also, if one assumes that the telephone has an adaptive gain control, which optimizes the loudness of the input signal, the determination of the loudness is not required within PESQlight. Instead, the sample loudness can be multiplied by a constant factor.

The above statement holds only, if the sample length is much greater than the length of the dropped frame. Otherwise, the loudness of the concealed versions could change.

Irrelevant asymmetric distortion: The unmodified PESQ determines the overall frequency distortion present in the degraded version. Then, an asymmetric signal derived from the original is calculated, which does not include the lacking frequency components. The frequency distortion is caused by the codec. It the codec is known its individual frequency filtering can be determined in advance. For example, the μ-law coding does not introduce any asymmetric distortion. Thus, we do not have to calculate it.

In Table 7.4 we describe the components which have been optimized and simplified. A detailed description of the simplification can be found in [218].

7.5.3. Validation of PESQlight

The correctness of the PESQlight implementation has been verified by comparing it with PESQ. For example, Figure 7.18 displays (red) both MOS like ratings of PESQ and PESQlight over time. It also displays the original sound signal (blue) and the difference between PESQ and PESQlight (green). For course, PESQlight is not perfectly predicting the importance but it is faster.

To judge the computational complexity of an algorithm, we measure the execution time on a notebook with a Pentium M Centrino 1500 MHz CPU running Linux OS. Before starting and after ending the algorithm, we took timestamps and calculated their difference. The timestamps measure the number of CPU cycles with the read timestamp counter (RDTSC)

Table 7.4.: Functionality removed from the PESQ algorithm.

Function	**Description**	**Function**	**File**
Time alignment	The time alignment can be removed, because between original and disturbed signal no variable delays are introduced.	input_filter, calc_VAD, crude_align, utterance_locate	pesqmain.c
Voice Activity Detection	The Voice Activity Detection (VAD) occurs naturally already before calculation of the importance values. For example, a VAD is included in the encoder, the adaptive gain control or the echo compensation.		
Power reference	The values are not required for the real PESQ functionality.	pow_of	pesqmod.c
Utterances	The subdivision into several utterances is not necessary, because only speech segments no larger than one second are considered.	short_term_fft	pesqmod.c
Frequency responses compensation	No constant frequency distortion is to be expected because of the given codec.	total_audible, time_avg_ audible_of, freq_resp_ compensation	
Constant loudness		fix_power_level	pesqmain.c
Skip silent samples at start			pesqmod.c

Table 7.5.: Prediction quality versus segment length.

Segment length [ms]	**One MOS calculation with PESQ [ms]**	**One MOS calculation with PESQlight [ms]**	**Performance gain**
1000	63.23	22.43	2.82
750	47.83	20.23	2.36
500	31.21	10.95	2.85
250	26.45	8.69	3.04

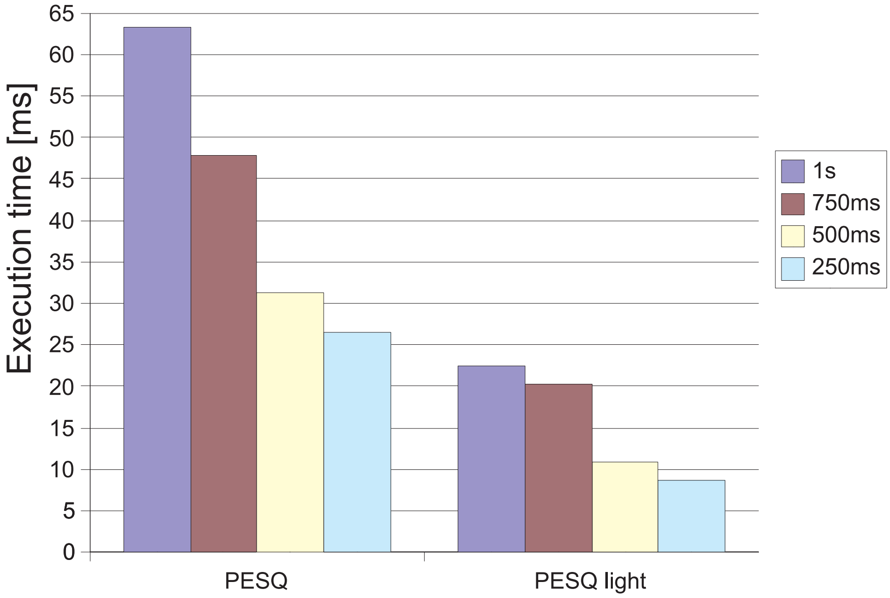


Figure 7.19.: Segment length versus execution time.

instruction [97]. This type of time measurement has low dependency on the utilization of the system when compared with timer based measurements.

7.5.4. Performance of PESQlight depending on the segment length

The first experiments were conducted to determine the execution time in relation to the length of the speech segment L. As an upper limit a segment length of one second was chosen. The lower limit is 250 ms, as it is the limit given in the PESQ algorithm. Each performance observation is based on 10000 MOS value calculations. Table 7.5 displays that the execution time increases with increasing segment length. In Figure 7.19 this expected result is displayed graphically.

Table 7.6.: Performance (given as cross correlation R) of PESQlight versus G.711 off-line packet classification.

Segment length [s]:	**0.25**	**0.25**	**0.5**	**0.5**	**1.0**	**1.0**
Position of lost segment:	**N-1**	**N**	**N-1**	**N**	**N-1**	**N**
considering only on frames	0.600	0.318	0.629	0.283	0.583	0.346
considering only off frames	0.034	0.014	0.050	0.017	0.078	0.047
considering all frames	0.404	0.165	0.466	0.141	0.494	0.242

7.5.5. Benchmarking

Similar as in Section 7.4 we rate the prediction performance of our approach. Our approach cannot classify the difference of silent frames; however it works well for all and for only the active speech frames. The prediction performance is significant better, if the lost frame is the next to last (Table 7.6).

7.5.6. Outlook on further enhancements

In this work PESQ was optimized for the use with μ-law. Some of the optimizations were only possible due to the codec's properties. Therefore, if the codec is changed, also the packet classification algorithm has to be modified, which might change the execution complexity. If one wants to change PESQlight to be used with other codecs the following properties have to be modified:

- Different frame lengths.
- Compensation of the delay introduced by the concealment.
- Inclusion of distortions that are caused by coding. For example, the loudness levelling or the asymmetric frequency distortion.

At this point we should explain the possibilities of further enhancements of the PESQ algorithm with regard to its execution time. The following modifications would be possible:

- The minimum segment length of 250 ms is limited by the window dimensions of the FFTs. If this limit is reduced, the execution time is also reduced.
- Another gain in execution time could be achieved by the optimization of the algorithms FFT. In [94] a speed increase is achieved by use of a 4 cousin FFT. Speed profits from to 50% could be proved by use of this more efficient FFT implementation – for example – using the MMX instruction set.

- During work on this topic, we look also on the possibility to save intermediate results so that the calculation of the next importance value can utilise them. This would reduce the complexity significantly.

7.6. Conclusions

Given the knowledge of packet importance, we showed that significant performance gains can be achieved if only packets are transmitted with priority that are important. However, the importance of speech frames has to be known precisely, otherwise this performance gains are lost (refer to Section 5.7).

The importance of a packet can be measured both off-line and in real-time. A measurement procedure that identifies the impact of a single frame loss offline has already been developed and has been verified with formal listening-only tests. In this chapter we studied how the importance can be measured in real-time. This is difficult as the importance values partly depend on the amount of error propagation, which is not known at the time of transmission. Waiting for the next frame before calculating the importance value significantly increases the accuracy of the importance predictions. The enhancement comes at the cost of an increased algorithmic delay. A good compromise is a look ahead of 20 to 40 ms to minimize error propagation effects.

We used our off-line measurement procedure as a reference method to benchmark the existing real-time frame classification algorithms DTX, Source-Driven Packet Marking, and SPB-DiffMark. Furthermore, we presented a novel algorithm that predicts frame importance more precisely than the aforementioned algorithms as it shows a higher correlation with the reference measurement values (Table 7.7). Our classification algorithm calculates of the importance of a μ-law coded VoIP packets. The run-time and complexity of the algorithm is reduced while maintaining its prediction accuracy. A modified PESQ algorithm called PESQlight is developed and it is fast enough to calculate the importance at real time.

Further enhancements can be expected if speech frame importance is calculated using the inherent features of the encoding process. These optimizations have not been addressed in this thesis and we suggest further research with the aim of reducing the computational complexity and increasing the prediction accuracy.

Table 7.7.: Overview on packet classification algorithms.

Classification	**Codec**	**Performance for all frames (R)**	**Performance considering only active frames (R)**
VAD G.729	G.711	0.38	-
VAD G.729	G.729	0.30	-
VAD AMR	AMR 12.2	0.41	-
VAD AMR	AMR 4.75	0.36	-
G.729 voicing	G.711	-	0.18
AMR voicing	AMR 12.2	-	0.19
AMR voicing	AMR 4.75	-	0.30
De Martin, complete	G.729	-	0.19
De Martin, Fixed codebook gain	G.729	-	0.34
SPB-DiffMark, N=3	G.729	0.10	0.10
Our approach, 0.25s, N-1	G.711	0.40	0.60
Our approach, 0.25s, N	G.711	0.17	0.32
Our approach, 0.5s, N-1	G.711	0.47	**0.63**
Our approach, 1s, N-1	G.711	**0.49**	0.58

Part II.

Application and Data-Link Enhancements

8. State of the Art

8.1. Wireless LAN

The Institute of Electrical and Electronics Engineers (IEEE) ratified the original 802.11 specification as the standard for wireless LANs in 1997. The first version of the standard described transmission modes with 1 Mbps and 2 Mbps data rates at 2.4 GHz (Figure 8.1). In 1999, the 802.11b standard has been ratified, which provides data rates up to 11 Mbps also at 2.4 GHz. Latest additions include 802.11a for 54 Mbps in the 5 GHz frequency band and 802.11g also up to 54 Mbps at 2.4 GHz.

IEEE 802.11 defines two possible operational modes: infrastructure mode and ad-hoc mode. In the infrastructure mode the wireless network consists of at least one access point (AP) which connects one or more mobile stations to a wired network or backbone. The ad hoc mode consists of several wireless stations, which communicate directly with each other.

8.2. IEEE 802.11 medium access protocols

The IEEE 802.11 standard specifies the Medium Access Control (MAC). The MAC layer is a set of protocols which is responsible for maintaining order in the use of the shared medium. The original standard supports two MAC mechanisms, the Distributed Coordination Function (DCF) and the Point Coordination Function (PCF). While the DCF is responsible for asynchronous data services, the PCF offers time-bounded services.

The medium access of IEEE 802.11 is partitioned into continuously repeating *superframes* (Figure 8.2). Superframes are separated by periodic management frames, the so-called Beacon frames. Then, a contention-free period (CFP) follows, which is controlled by the Point Coordination Function (PCF). Afterwards, the contention period offers random access controlled by the Distributed Coordination Function (DCF).

IEEE 802.11 uses the three inter-frame spaces (SIFS, PIFS, and DIFS) to control the medium access, i.e. to give stations in specific cases higher or lower priority (see Figure 8.3).

8.2.1. Distributed Coordination Function (DCF)

IEEE 802.11 implements a Carrier Sense Multiple Access with Collision Avoidance (CSMA/CA) MAC in order to avoid collisions. In this protocol, when a node wants to transmit a packet,

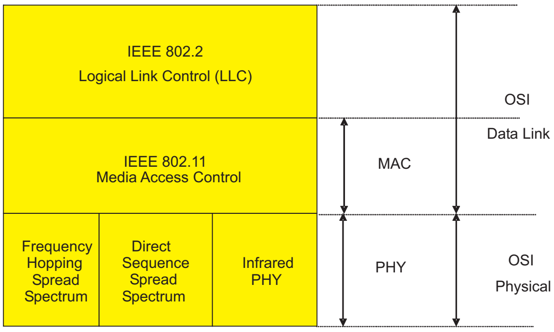

IEEE 802.2
Logical Link Control (LLC)
IEEE 802.11
Media Access Control
Frequency Hopping Spread Spectrum
Direct Sequence Spread Spectrum
Infrared PHY
MAC
PHY
OSI
Data Link
OSI
Physical

Figure 8.1.: The IEEE 802.11 standard covering the media access control (MAC) and the physical (PHY) layer.
The Physical Layer (PHY) of the original IEEE 802.11 standard included either the direct sequence spread spectrum (DSSS), frequency hopping spread spectrum (FHSS) or infrared (IR) pulse modulation.

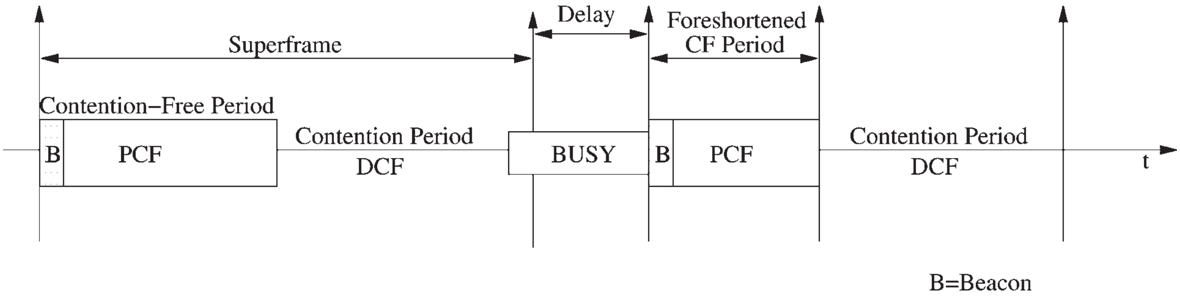


Figure 8.2.: Superframe structure in IEEE 802.11 [134].

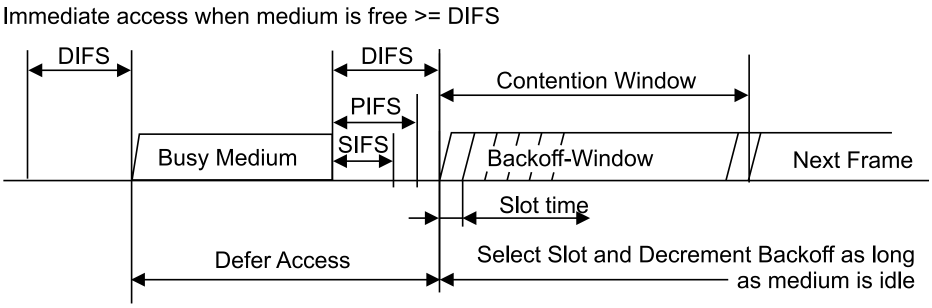

Immediate access when medium is free >= DIFS
DIFS
DIFS
PIFS
SIFS
Contention Window
Busy Medium
Backoff-Window
Next Frame
Slot time
Defer Access
Select Slot and Decrement Backoff as long as medium is idle

Figure 8.3.: Interframe space relationships in IEEE 802.11.

it first listens to the medium to ensure that no other stations are transmitting. If the channel is free, the node transmits the packet; otherwise, it waits for the medium to become free, and then waits a random amount of time before attempting to access the medium. Since the probability that two stations will choose the same waiting time, collisions between packet transmissions are rare.

On the wireless medium, collisions cannot be detected, because when a node is transmitting, it cannot hear any transmissions of other nodes. To detect collisions, each successfully received unicast data packets is immediately (more precise, after a SIFS period) acknowledged by its the receiving node. If the acknowledgement (ACK) is received without errors, the transmitter of the data packet knows that no collisions or transmission errors occurred.

In DCF, the sending stations compete for the channel. A station has to sense the channel to become idle before sending a frame. After the medium became idle, the stations wait for the DIFS before starting the random back-off procedure. The random back-off procedure is a slotted, exponential back-off procedure similar to slotted Aloha.

Each station that wants to send calculates a random *backofftime*= $slotTime \cdot Random$, where *Random* is a pseudo-random integer value from a uniformly distributed contention window $[0, CW]$. The upper bound CW is initially $CW = CW_{min} = 7$. However, if an acknowledgement has not been received and a collision has occurs, the sender assumes a collision and doubles CW, until CW_{max} reaches 255.

During the back-off procedure, each station decreases its *backofftime* as long as the medium is idle. If the *backofftime* equals zero, the station starts to the send its packet. Then, the medium is not idle anymore and all other stations stop the back-off procedure. After a station has successfully transmitted its packet, it sets its CW value back to CW_{min} again.

8.2.2. The Point Coordination Function (PCF)

The PCF can only be used in infrastructure-based networks as it requires an access point (AP). Usually the Point Coordinator (PC) runs on the AP. The PC manages the access to the medium in the CFP by polling stations sequentially. The PCF comes with a higher complexity than DCF. It is implemented in most commercial APs but is rarely used due to the lack of an optimized scheduling and polling algorithm.

8.2.3. IEEE 802.11e MAC

The DCF and PCF MAC modes cannot achieve a good quality of service, if a high background load is present. Thus, many QoS enhancements have been studied and evaluated. For example, the QoS enhanced MAC protocol IEEE 802.11e is currently standardized.

For achieving QoS, 802.11e uses multiple priority queues for the prioritized and separate handling of different traffic categories (TCs). In addition, 802.11e introduces the Enhanced Distributed Coordination Function (EDCA) and the Hybrid Coordination Function (HCCA).

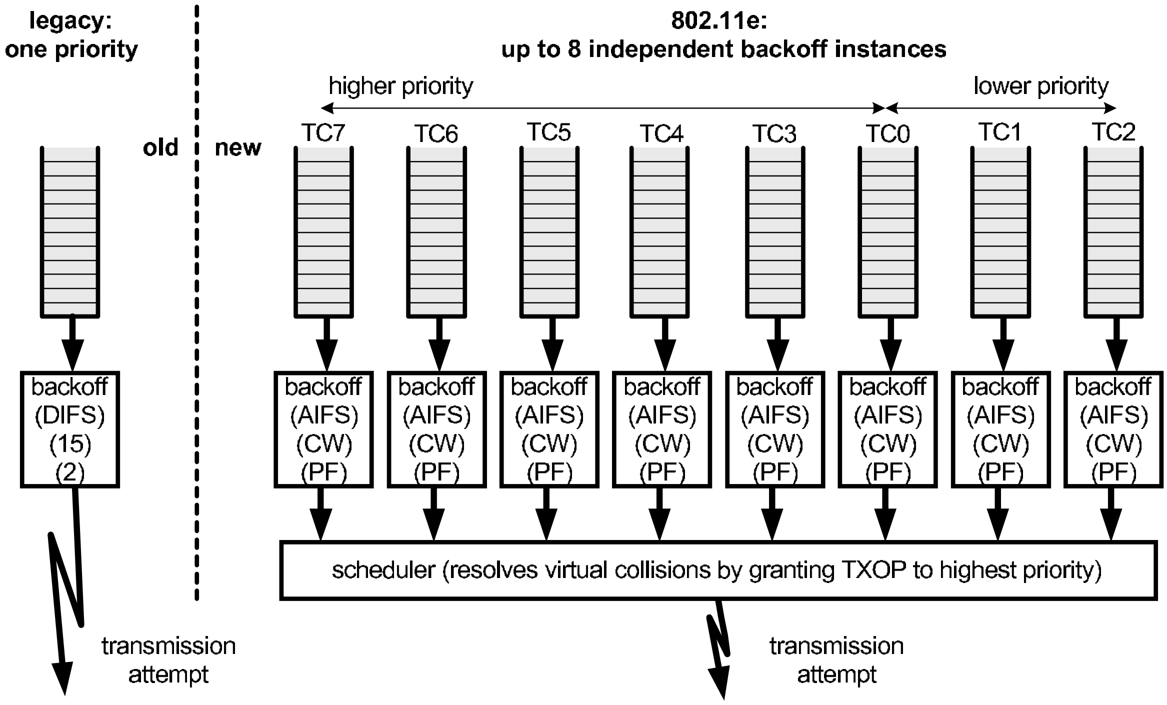


Figure 8.4.: EDCA supports multiple back-off instances in parallel.

The EDCA manages the medium access in the CP while the HCCA is responsible for the CFP and the CP. Both functions are described below.

8.2.3.1. The Enhanced Distributed Coordination Function (EDCA)

The EDCA enhances the 802.11 DCF by introducing its own back-off instances with separate back-off parameter sets for each priority queue. It can be seen as multiple DCF MAC running in parallel (Figure 8.4). Each back-off instance is called *Traffic Category* (TC). Each TC has its own set of parameters to prioritise its data flow. In an old draft, 8 TC were supported [59]. Now only 4 different TC and transmission queues are supported [96].

DCF contends for the transmission of one packet. In EDCA, each TC on a station contends for a transmission opportunity (TXOP), which can contain multiple packet transmissions. A TXOP is defined in [96, 140] as *"an interval of time when a station has the right to initiate transmissions, defined by a starting time and the maximal duration"*.

The parameters for the prioritization of TCs are the arbitration inter-frame spaces (AIFS), the minimum size of the CW ($CW_{min}[TC]$) and the $TXOP_{limit}[TC]$ (see Figure 8.5):

- The AIFS describes the duration of time that the medium must wait after the last transmission before a station can access the channel or decrement a TC's back-off counter.

- To ensure prioritized access with respect to legacy 802.11 stations, the EDCA should

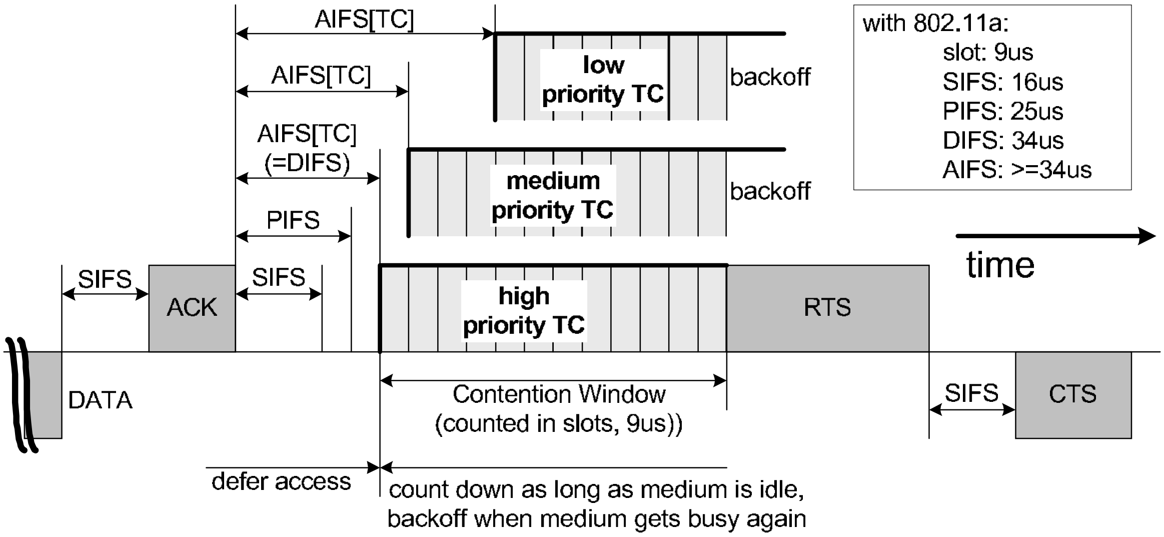


Figure 8.5.: IFS relationships in 802.11e (source [140]).

use smaller CW_{min} values for high-priority data flows, which is exponentially increased after an unsuccessful transmission.

- The $TXOP_{limit}[TC]$ specifies the maximal duration of one TXOP.

Two or multiple TCs can start to transmit at the same time. To avoid such potential collisions, a virtual scheduler grants access to the TC with the highest priority and starts a back-off for the lower TC (after increasing $CW_{min}[TC]$).

8.2.3.2. The Hybrid Coordination Function (HCCA)

The HCCA controls both the CFP and the CP. It uses a polling scheme to control the medium access. To grant and administer polled-TXOP requests a scheduling management called Hybrid Coordinator (HC) is included in the PC. In this thesis, we do not study the HCCA because we assume that EDCA is sufficient to support QoS and because EDCA has a lower complexity.

8.2.4. The Contention Free Bursting (CFB)

Tourrilhes [205] proposed the idea of Contention Free Bursting (CFB) to improve the performance of small packets (of time-bounded services) in Wireless LANs. CFB decreases the overhead and delay and increases the throughput. CFB sends multiple small packets as a burst without intermediate contention as soon as the station gains access to the medium (see Figure 8.6).

It is possible to send packets to different destinations in one burst frame. Between an ACK and the following packet a time interval of SIFS is only required and the contention period

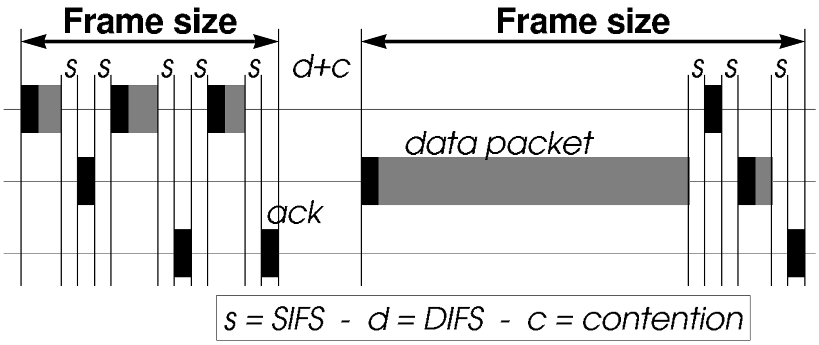


Figure 8.6.: Principle of CFB [205].

is omitted. The station maintains control over the medium during the whole burst. Sending multiple small packets in a burst avoids contention for each single packet and increases the efficiency. However, the medium access time might be increased because packet bursts occupy the medium for a longer period.

8.3. VoIP over WLAN

One primary design goal of the IEEE 802.11 wireless LAN standard has been to define how to connect wireless computers to local area networks. The high volume of traffic in a LAN consists of TCP like WWW or email. But does WLAN also allow to transmit interactive speech [12, 21, 210] in a good quality?

The DCF and PCF modes provide inadequate performance [208] and various performance improvements have been proposed and evaluated (overview in [134]). In the following we concentrate on papers that include WLAN as well as QoS to enhance the quality of telephone calls.

Veeraraghavan et al. [206] have analyzed how many voice flows can be transmitted simultaneously in an IEEE 802.11 network if the PCF polling mode is applied. D. Chen et al. [39] studied the capacity of IEEE 802.11b's PCF mode to transmit variable bit rate (VBR) VoIP calls. Their results how that the capacity is up to 17 at 11 Mbps and 10 voice calls at 2 Mbps.

In [124], Köpsel et al. simulated whether the DCF and PCF MAC mechanisms can transmit real-time traffic. In the DCF mode stringent delay requirements are fulfilled only in low load scenarios. In a high load scenario or in a scenario with a high number of nodes, DCF fails to provide low delay and jitter. Therefore, the authors suggest to switch from DCF to the PCF mode in those cases. In [124], the audio flows are transmitted over a 2 Mbps wireless channel. In the case of an audio stream with 64 kbps coding rate and 20 ms packetisation, the capacity is 12 stations in the DCF mode and 15 in the PCF mode. As a minimal quality level, the authors have chosen a maximal transmission delay of 250 ms and maximal 5% packet loss.

The usage of PCF, however, decreases the overall throughput due to unsuccessful polling attempts.

In a follow-up publication [125], Köpsel studies the benefit of higher data rates. Increasing the data rate (up to 54 Mbps) leads only to limited quality improvements. This effect can be explained due to of the packet overhead of the IEEE 802.11 PHY and MAC protocols, containing large protocol headers at a low rate, immediate acknowledgements (ACK) and large gaps between the packet transmissions (inter-frame spaces). Instead, to improve delay and jitter the authors suggest to use a transmit queue that supports two priorities. The high priority is reserved for interactive voice flows whereas the low priority is intended for the best effort traffic. If a high priority queue is present, the author does not see an immediate need for an extended DCF mode.

In [60], S. Garg et al. experimentally studied the capacity of IEEE 802.11b to determine the maximal number of VoIP calls. The maximal number depends on the packetisation of VoIP (reciprocal of the packet frequency), the geographic distribution of the wireless clients, and the distance between the wireless clients and the base station. The authors determined the quality of VoIP calls by measuring packet delay, jitter and loss rate. Using G.711 and 10 ms packetisation six simultaneous calls were possible. Starting the seventh, only the wired-to-wireless streams failed. The authors concluded that lowering the packet frequency is the most efficient solution to increase the number of VoIP calls in a WLAN cell.

P. Garg et al. simulated the ability of IEEE 802.11e's EDCA and HCCA coordination function to support an effective QoS and higher channel efficiency [59]. They transmitted various flow types (VoIP, video and ftp) over a basic service set and measured the delay distribution and bandwidth. Their simulation model is an extended version of Atheros Communication's 802.11e model for ns-2. Their findings lead to the conclusion that both the coordination functions are highly sensitive to the chosen parameters. However, they can reach the desired QoS requirements but HCCA has a higher bandwidth efficiency than EDCA.

Choi et al. [40] compared IEEE 802.11 DCF with IEEE 802.11e's EDCA and CFB according to throughput, dropped data rate and delay in an IEEE 802.11b PHY. In their scenarios, they used a combination of unidirectional voice, video and data traffic. They reported a large decrease in the number of dropped packets and delay as well as a more constant throughput for voice and video transmission in the EDCA simulations.

Kawata specified a detailed protocol behaviour of PCF in 802.11 in order to support voice traffic efficiently [121].

Casetti and Chiasserini proposed enhancements to the 802.11 EDCA MAC mode to support Voice traffic fair regardless of the transmission direction [33].

8.4. Prioritisation of speech frames

Discarding of speech frames or segments, depending on their priority, has been discussed for many years [7, 20, 163, 190, 194]. The aim is to increase the capacity of multiple flows are transmitted over a common channel. Depending on the activity of the voice source, a different priority is assigned to packets. Then, if the link is temporally congested, low priority packets are dropped first.

Sanneck proposed to use a modified Random-Early-Dropping (RED) strategy at packet forwarding nodes. If a node is congested, the probability of packet dropping should depend on the packets' marking. Sanneck proposed to mark packets with +1 (foreground), 0 (best-effort), and –1 (background traffic) depending of their speech properties using SPB-DiffMark. Packets with a +1 will be dropped at a low probability, and packets marked with –1 with the highest probability. The number of fore- and background packets should balance over the long term due to fairness requirements.

This algorithm has been evaluated under different loss patterns using objective speech quality evaluation algorithms (MNB and EMBSD). The authors showed that the SPB-DiffMark algorithm increases the perceptual quality of VoIP, compared to alternative algorithms, such as the alternating or random marking of packets.

De Martin [48] has proposed an approach called Source-Driven Packet Marking, which controls the priority marking of speech packets in a DiffServ [23] network. If packets are assumed to be perceptually critical, they are transmitted in a premium traffic class. All other packets are sent using the best-effort traffic class.

Petracca et al. [164] presented an algorithm to transmit AMR coded speech over 802.11 WLANs. About 10 percent of the most important packets are marked as critical. On the wireless link, premium packets are protected by packet repetition and on wireline links with a modified RED queue management.

8.5. Rate-distortion media streaming

In the last years, several rate-distortion optimized multimedia streaming algorithms have been presented, which utilize the temporal variance within a multimedia flow. These approaches are all based on the fact that not every *media frame* in a multimedia flow is of equal *importance*. The losses of some frames are hardly noticeable, whereas other frames can cause significant degradation in the perceptual service quality. Thus, the communication network should concentrate on preventing the loss of more important frames rather than those lesser importance.

Recently, it has been investigated how media streams can be transmitted over a lossy channel in a rate-distortion optimized way. A packet scheduler could calculate the expected distortion and rate of any particular packet transmission schedule and then choose the optimal one. However, as noted in one of the first publication on this topic by Podolsky and McCanne [166], this approach is complex.

Chou et al. introduced a framework that utilizes the different importance of multimedia frames. It decides for each given transmission opportunity, which packet – if any – should be sent in order to optimize the distortion for a given rate [41]. The authors presented an iterative algorithm using dynamic programming, which reduces the complexity of the search for the optimal schedule. At each discrete time step, these schedules are update to take into account the latest error-control feedback. Chou shows that performance gains of up to 8 dB can be achieved for audio and video flows.

However, to limit the complexity, the authors assume a simple error concealment and error propagation model. This assumption inhibits the use of many, if not most, coding schemes. A better error model is introduced in [35], which also extends the framework to networks with multiple, diverse paths.

Chou and Miao considered one fixed transmission deadline. In practise, due to playout rescheduling the existence of multiple deadlines can be assumed. Kalman et al. applied the R-D framework to this problem in [119] and described the tradeoff between buffering latency and reconstruction quality in a R-D framework. Kalman has shown via simulations and analysis that an adaptive media playout can enhanced the streaming of media over lossy links supporting ARQ [120].

RaDiO framework has been extended to consider the effects of congestion [189].

A low-complexity version of RaDiO uses precomputed importance values, which are transmitted in parallel and piggy backed to the actual media data [34].

Röder has proven that RaDiO streaming is NP-hard and developed a branch-and-bound algorithm to calculate the Lagrangian for a single data unit [177, 182].

8.6. WLAN data link characteristics and the impact of motion

The performance of WLAN radio modems has been often measured, but the impact of motion is rather seldom covered:

Nguyen and Katz published results on WLAN measurements in 1995 [152]. They studied the loss behaviour of AT&T WaveLans. They applied a trace-based approach and showed that WaveLan transmissions experience an average IP-packet error rate of 2 to 3 percent. The authors developed an error model to control simulation models. The simulation models showed a high correlation with experimental results of TCP throughput on WLAN.

Willig et al. [217] conducted bit error measurements taken with IEEE 802.11b-compliant MAC-less radio modems in an industrial environment. Some measurement traces included machinery, which moved. The author's results allowed some conclusions about the error characteristic of wireless links to be drawn: A general observation is that mean bit error rates are time-variable over several orders of magnitude. Sometimes, long lasting consecutive packet losses occur. Such link outages should be taken into account by the higher layers.

The first published work on WLAN link quality with moving nodes is by Chen and Forement in 1995 [38]. They studied radio communication between vehicles. Two cars were driven through outdoor environments. Both cars communicated via a 900 MHz, 2 MBit/s WaveLan radio. The authors were primary interested in the transmission range, if the nodes are out-of-sight, if additional cars were in between and if the two nodes were moving. They did not notice an increase in the packet error rate if the cars were moving.

Recently, there have been many efforts to study the use of WLAN for vehicle communications. In [157] Ott and Kutscher used WLAN for car to stationary AP communication and from car to car communication. At high speeds (80-180 km/h) the communication quality (throughput and delay) decreases with speed. This fact is partly explained by the shorter connection time between car and AP.

Zorzi achieved analytical results on the performance of packet transmission over mobile radio channels. In his early work Zorzi *et al.* [225] analyzed the accuracy of a two state Markov model, when it is applied as an approximation of block transmissions over a slowly fading wireless channel using a Rayleigh fading model [111]. The Markov model describes a wireless channel with two states, good and bad. The states have exponentially distributed holding times. The hold times decrease if the Doppler Frequency increases. Zorzi states that the Markov model is a good approximation; he also used it in his following publications.

For example, in [226] the authors analysed the lateness probability of an ARQ scheme on a two-state Markov channel. One result states that the probability of a packet being too late even after queuing and multiple retransmissions depends on the length of the error burst. For an average error burst length somewhat smaller than the packet length, the lateness probability is minimal. For longer and shorter burst lengths it increases.

Even though Zorzi's analytical models can be widely applied, we did not find a study on the effects of slow user movements on IEEE 802.11b wireless links.

8.7. WLAN based location sensing techniques

Approaches for in and outdoor location sensing techniques have been presented [78]. In this thesis we focus on locating techniques which use the intrinsic features of WIFI based wireless access. The RADAR system [9] has been one of the first approaches presenting an indoor positing system based on WLAN components – others have followed [64, 69, 116, 117, 126, 147, 223]. An essential part of location sensing algorithms is a method to determine the distance between two wireless nodes. In general, three methods have been considered:

1. The information, of which nodes are within transmission range, is used to estimate the distance. This approach benefits from densely populated networks such as sensor

networks [75].

2. The received signal strength indication (RSSI) of data packets is considered as it decays with distance. As RSSI decreases sharply in a non-linear fashion with distance, signal strength maps have to be gathered to relate the RSSI values with positions. Generating these maps is time-consuming and it has to be redone if the environment changes.

3. The propagation time of radio signals can be used as it linearly increases with the distance in free air. Such an approach is usually considered to be impossible without the assistance of special signal processing hardware [207].

The classic approach to the latter method of position location estimates the time of arrival (TOA) of pure radio signals (instead of WLAN packets). This is conducted by applying signal processing algorithms based on cross-correlation techniques [131]. The received signal resembles the initial transmitted signal delayed by the propagation delay. The autocorrelation function for the transmitted signal shows its maximal peak at a certain shift in time. TOA based time measurements require synchronised clocks. Although TOA as a ranging metric is considered to be the most promising technique for accurate indoor positioning [6], the method suffers from multi-path conditions. The difficulty is to determine the autocorrelation peak referring to the signal travelling along the direct line of sight (DLOS). The problem can be encountered with a wider frequency band, e.g. ultra-wide band.

TOA measurement is being employed both outdoors for GPS-positioning [53] and indoors to find objects marked by a tag [212]. In the latter paper, the author gives the achievable accuracy when measuring the round trip TOA within the 2.44 GHz and 5.78 GHz bands. For a signal bandwidth of 40 MHz an accuracy of 3.8 meters can be the achievable resolution limit unless further signal processing techniques are applied. Those might enhance the resolution up to 1 meter.

An early paper focusing on measuring pure packet propagation delays is [130]. The objective is to determine the speed of light using the averaged measured round trip propagation delay of many ping packets and the known distance between the sender and receiver. The measurements were conducted in a wired Ethernet infrastructure. Estimating the propagation delay which ranges below the clock resolution was achieved by employing the concept of noise-assisted sub-threshold signal detection. The aim of this work is teach students the effect of stochastic resonance [58] and to explain how to enhance the resolution. For measurements in an IEEE 802.11b wireless environment, the round trip times were too variable and noisy to be used.

9. Adaptive VoIP Applications

Recent studies show that a significant number of Internet backbone links do not provide *toll quality* — the lowest quality of classic PSTN based telephone calls — when used for Voice over IP (VoIP) applications [141]. To overcome this shortcoming, *application level control* of the transmission of voice calls is a promising approach. Application level control can be used to complement or substitute QoS mechanisms like over-provisioning [57], DiffServ, overlay networks, or semantic data-link protocols. Voice over IP applications can adapt the *VoIP configuration* to the current state of the network. In recent years, several algorithms have been proposed that dynamically tune the configuration to current packet delays and losses. These algorithms change the size of the playout buffer, the coding rate and the amount of forward error correction in order to maximize the VoIP quality. How can an adaptive VoIP application assess the quality of its transmission and then predict the impact of its adaptation actions?

It is obvious that the quality of telephone calls should be measured by the users: Humans should evaluate the *perceived* Quality of Service (QoS). Of course, such subjective measurement campaigns are time consuming and costly if statistically meaningful results are to be obtained (ITU P.800, [100]; they are also evidently not applicable for on-line adaptation. On the other hand, VoIP applications measure only directly observable, network or transport-layer metrics like packet loss rates, round trip times, and packet delay distributions. These metrics of *networking QoS* [219], however, do not reflect the perceived service quality precisely.

An efficient way to correlate the perceived QoS and the networking QoS are *quality models* that simulate human rating behaviour. In Section 3.1 and Chapter 4 we described quality models that calculate a perceptual quality rating using input parameters. The main drawbacks of these three quality models are the following: The PESQ algorithm is not able to predict the speech quality at run-time nor does it take into account end-to-end delays. As well as being computationally complex it is also patented. The E-Model, on the other hand, considers operational parameters which are not known or are not relevant to the application (see Section 9.1). It does not consider the impairment due to dynamic adaptations. Furthermore, it assumes tandem coding (transcoding) conditions [103] and as a result leads to an imprecise correlation between the loss rate and speech quality. Our new model is not suitable and has similar drawbacks as it is based on PESQ and E-Model. Thus, none of the quality models are suitable for adaptive VoIP applications as they work under different operational conditions and lack particular features demanded by adaptive VoIP applications.

In this chapter, we present a perceptual quality model that is primarily intended for adaptive VoIP applications. It is similar to the quality model that is presented in Chapter 4. However, in order to cope with the high computational complexity that PESQ introduces, we pre-calculate the impact of impairments that occur in a VoIP system. The pre-calculation is done with PESQ and stored in a database. Our main contributions are the following:

1. We measured the coding distortion of the commonly used codecs with PESQ for different loss rates and loss patterns without considering tandem coding.
2. We measured the impairment of speech quality, when the packet playout schedule is adjusted, and determined the detrimental effect caused by switching between different coding rates. Contrary to the generally accepted view, switching coding modes does noticeably harm speech quality.
3. Our quality model is open-source and available on the Internet [82]. The model can be used in several circumstances. In particular, its on-line nature enables its use within applications to judge the actual or potential benefits of modifying protocol parameters.
4. To demonstrate the potential of our model, we apply it to select the ideal coding and packet rate in bandwidth-limited environments.
5. Furthermore, we decide, based on model predictions, whether to delay the playout of speech frames after *delay spikes* (refer to Section 4.3).

We show a considerable improvement in perceptual speech quality if our model is applied to control VoIP transmissions. In addition, these results give important insides on how to design VoIP optimised data-link protocols regarding the requirements on packet delay and loss.

This chapter is structured as follows: In Section 9.1, we describe the requirements for an application layer quality model. Next, we describe our quality model. In Section 9.3, we present measurement results on the coding performance of common codecs and parameterize our quality model. We include two application examples for the quality model in Sections 9.4 and 9.5. Finally, the results of this chapter are given and rules-of-thumb are presented.

9.1. Requirements

If a perceptual quality model should be applied to the adaption of VoIP parameters in the application layer, certain requirements have to be met. In general, application layer control can be divided into two parts, an acoustic part and a transmission control part. Although the acoustic processing is highly important, we shall not discuss it in this thesis. The transmission control is briefly surveyed in Section 2.2. In Table 9.1 we give references to show the diversity of adaptive VoIP algorithms that a quality model should deal with. Many tradeoff decisions have to be make to find the optimal parametrization (see Figure 9.1).

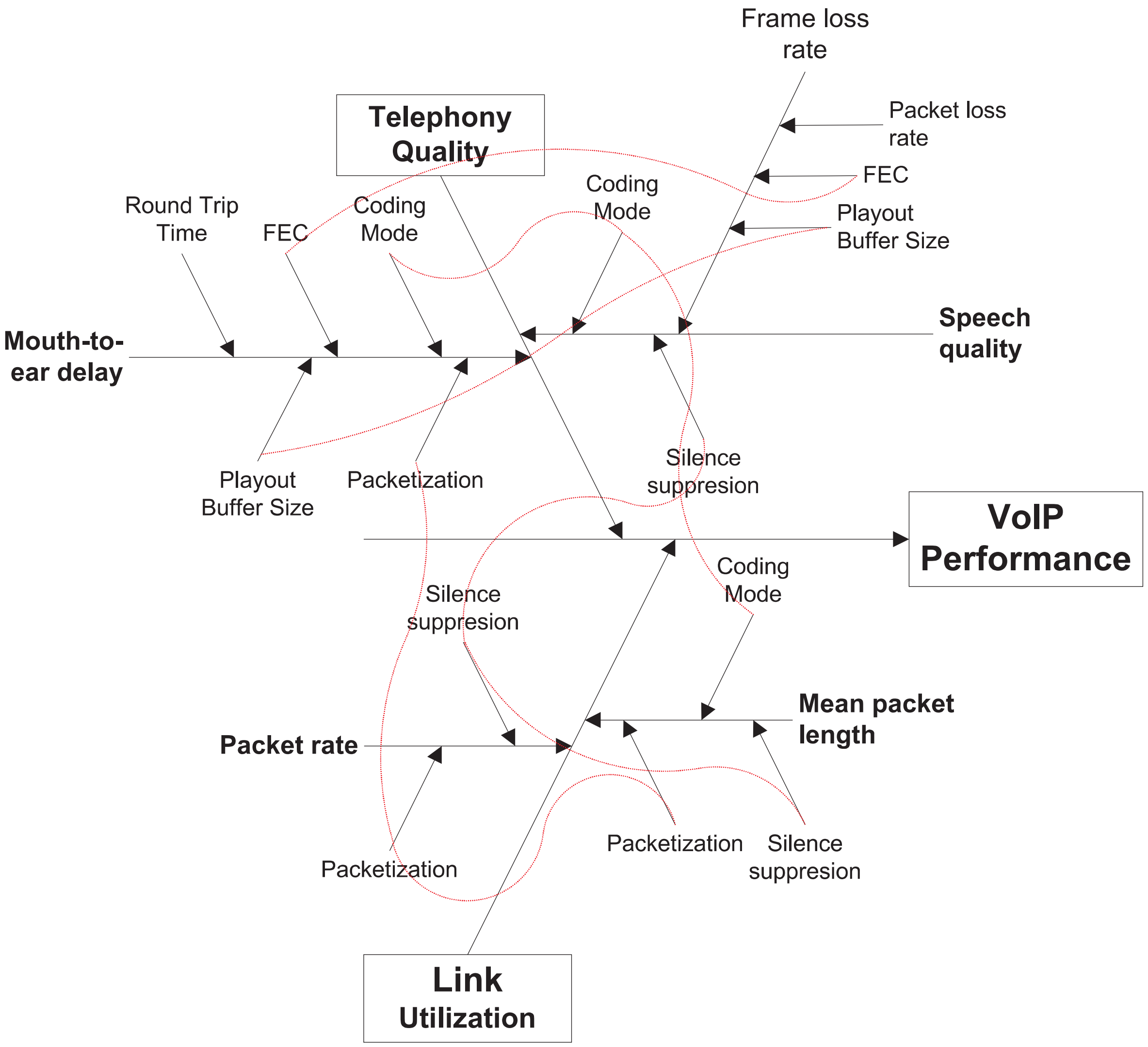


Figure 9.1.: Application layer adaptation optimises the VoIP configuration as it controls the coding mode, FEC, packetisation, and silence suppression.
The VoIP configuration influences many different parameters, like the delay, the speech quality, the mean packet length, and the packet rate. These parameters have an effect on the link utilization and the telephony quality, which both define the efficiency and performance of a VoIP transmission.

Table 9.1.: Overview on application-layer adaptation algorithms.

Parameter	Studied by	Effect
Playout scheduling	[129,132,135,146,165, 173,193]	The later the playout deadline, the higher the delay but the lower the frame loss rate.
Coding	[10,19,79,138,188]	The codec influences the speech quality and has a certain algorithmic delay.
Coding rate	[10,188,222]	The higher the coding rate, the better the speech quality, and the larger the packet length.
Silence suppression	NA	If silence suppression is used, the speech quality slightly degrades but less speech frames need to be transmitted.
Forward error correction	[25, 27, 30, 161, 167, 179]	FEC decreases the frame loss rate but comes at the cost of higher delay and larger or extra packets.
Packetisation	[206]	If the packetisation is increased, which means that more speech frames are transmitted in one packet, the delay is increased also but the number of packets decreases.

To predict the telephony quality, our quality model has to cope with the static and variable impairments due to coding distortion, packet loss and delay. These are the coding distortion due to the selected codec and the impact of switching the coding modes, the impact of mean and single packet losses, the absolute transmission delay and the impact of variable playout delays. The quality model is required to have low computational complexity and delay as it needs to be used in real-time conditions. Last not least, if used in an open-source telephony application, it further needs to be free of license fees and patents. These requirements are not met by PESQ, E-Model and other published quality models. An overview is given in Table 9.2.

9.2. An application layer quality model

As published quality models do not fulfil all requirements, we introduce a new quality model. It takes into account coding distortion, packet loss and delay to predict the perceptual quality but it assumes an optimal acoustic processing. We split the quality model into *source* and *sink* sides. The source controls the transmission of voice, based on a periodic, but delayed feedback of mean packet delays and loss rates from the receiver. The other side, the receiver, has to react to incoming packets immediately. For example, the playout time may have to be adjusted to accommodate a late packet. Our quality model has to take into account both these time scales.

Table 9.2.: Properties and features of quality models.

Features of quality model and which kind of impairments they consider	PESQ	E-Model	Combined Model (Chapter 4)	Application Model (Section 9.2)
coding distortion	yes	yes	yes	yes
mean packet loss rate	yes	yes	yes	yes
absolute delay	no	yes	yes	yes
delay variations	yes	no	yes	yes
single packet loss	yes	no	yes	yes
switching the coding mode	yes	no	yes	yes
computational complexity	high	low	high	low
works at real time	no	yes	no	yes
license fee	no	yes	no	yes
acoustic impairments	many	many	-	-

9.2.1. Source side

In the following, only the parameters available at the source are considered. Equation 9.1 is based on the E-Model. If the acoustic processing is optimal we can simplify the E-Model to fewer parameters with c describing the codec, dtx the DTX mode, cr the coding rate, lr the mean packet loss rate, $pack$ the packetisation time, and t the end-to-end delay. The computation of R is then given by:

$$R = \mathrm{MOStoR}\left(\mathrm{MOS}\left(c, dtx, cr, lr, pack\right)\right) - I_{dd}\left(t\right) \tag{9.1}$$

The term I_{dd} is given in Equation 4.2. In Section 9.3, we derive $\mathrm{MOS}\left(\mathrm{c}, \mathrm{dtx}, \mathrm{cr}, \mathrm{lr}, \mathrm{pack}\right)$ values from PESQ measurements. In a real implementation, the values would typically be stored in a table for efficiency reasons. If the table does not contain a parameter but only higher and lower values, the MOS value is calculated by linear interpolation of available values.

9.2.2. Sink side

At the receiver, we introduce a novel view of quality: The quality is degraded by a continuous flow of *impairment events* that relate directly to a single psychophysical stimulus. An impairment event decreases the quality of the transmission temporally. It starts at some point in time t_{start} and lasts until t_{end}, when it is not noticeable anymore. In a VoIP system, three different events can cause impairments.

First, if one or multiple consecutive frames are lost, the quality decreases as the receiver-side concealment algorithm cannot extrapolate the acoustic signal. Second, if the playout scheduler changes the playout time, the speech may be impaired (Figure 9.7). Last, switching the coding mode or coding rate can cause “clicking” sounds (Figure 9.6).

Impairment events can be measured by the duration and the strength of their distortions. We use the same metric that has been applied in Chapter 5.4 to measure the impact of a frame loss [89]. This time we extend it to measure not only frame losses, but also to impairment events in general. Then, if impairment events occur, the resulting quality is described by MOS$(s, c, e_1, e_2, \ldots)$. The values of e_x describe the impairment events at position x. *The impairment of an event is defined as the difference between the quality due to coding loss and the quality due to coding loss and the change of VoIP configuration, multiplied by the length of the sample:*

$$\mathrm{Imp}\,(s, c, ev) = (\mathrm{MOS}\,(s, c) - \mathrm{MOS}\,(s, c, e)) \cdot t\,(s) \tag{9.2}$$

In Section 9.5, we will show how our new quality model, the measure of impairment, can be used to optimise packet loss bursts against playout adjustments.

9.3. Tuning the quality model

In the previous section, we introduced the abstract notion of our quality model. Still, the absolute parameters and variables have to be defined. For example, we introduce the function MOS$(\cdot)$, which defines MOS values for various operating conditions. We also introduce the notion of an impairment event. The objective of the following speech quality measurements is to determine the concrete curve and values of these functions so that the quality model developed here can use these values. To limit the length of this work we will confine ourselves to a single codec, the Adaptive-Multi-Rate coding (AMR), which is the default codec for third generation WCDMA systems [1].

9.3.1. Measurement setup

We followed the recommendation in [106], which describes how to derive the equipment impairment factor I_e from listening-only tests, but we used fewer test cases and instrumental assessment tools. Each single measurement consists of five steps and is repeated several times with different configurations (see Figure 9.2).

- First, a speech recording is selected from a database. We used the ITU P.suppl 23 data base [102] that contains 832 samples from four different languages, speakers and sentences. Each sample has a duration of 8 s. Additional background noise is not present.

- Second, the ITU reference implementation of AMR compresses the sample. AMR generates speech frames. Each frame contains 20 ms of speech and can be encoded with a coding rate of 4.75, 5.15, 5.75, 6.7, 7.2, 7.95, 10.2 or 12.2 kbps.

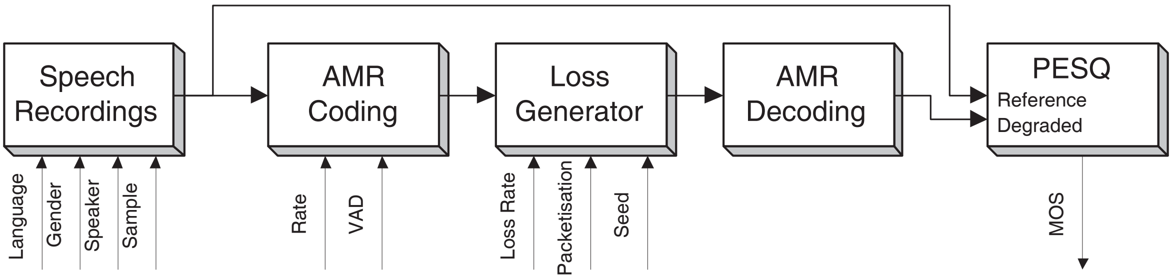


Figure 9.2.: Measurement set-up.

- Third, a loss generator simulates the packet losses depending on the loss probability, packetisation and a random seed.

- Next, the AMR decoder generates a degraded version of the speech sample and conceals lost frames.

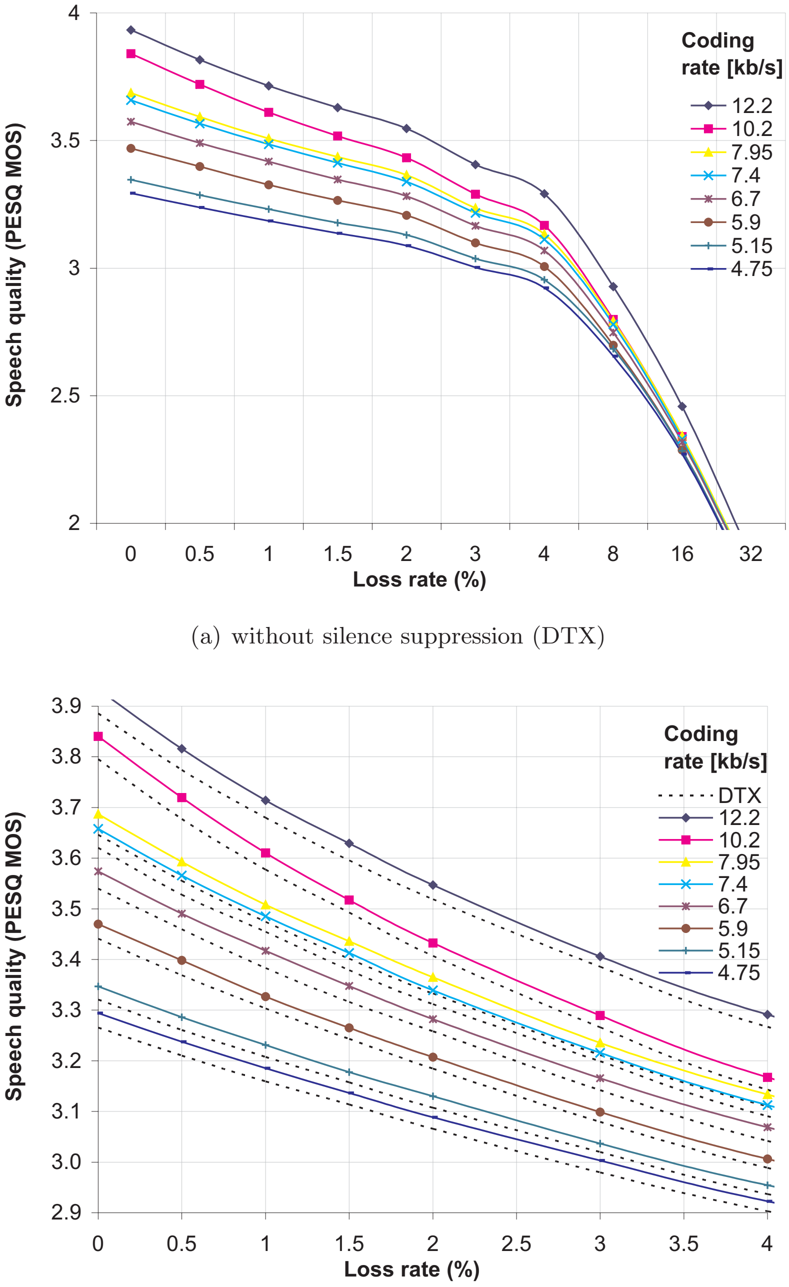
- Finally, the ITU reference implementation of the PESQ algorithm compares the degraded speech sample with the reference sample to calculate the MOS value.

9.3.2. Results

We study the impact of single random losses on the objective speech quality. Figure 9.3a shows the impact of loss and coding rate on the speech quality with a packetisation of one frame per packet. A lower coding rate and a high loss rate decrease the speech quality. Figure 9.3b displays the distortion due to silence compression (DTX), which is present but low. Figure 9.4 shows that a higher packetisation (=lower packet rate) does not change the speech quality to any large extent.

We move on and show the distortion caused by frequent switching of the coding rate (Figure 9.6) versus the mean coding rate. During the encoding of a sample, we switch the coding rate several times to different rates (Figure 9.5). For example, if the switching period $t_{switching}$ is 80 ms, the lower and higher coding rate alternating at a speed of 80 ms: The lower coding rate is selected from $t_{low} \in \{20, 40, 60\}$ and the higher coding rate is selected from $t_{high} \in \{60, 40, 20\}$ respectively. In Figure 9.6a we display the resulting speech quality for an average coding rate *mean.rate*. In Figure 9.6b contains the resulting impairment values. We also display the cases without any mode switching, which have an impairment of zero.

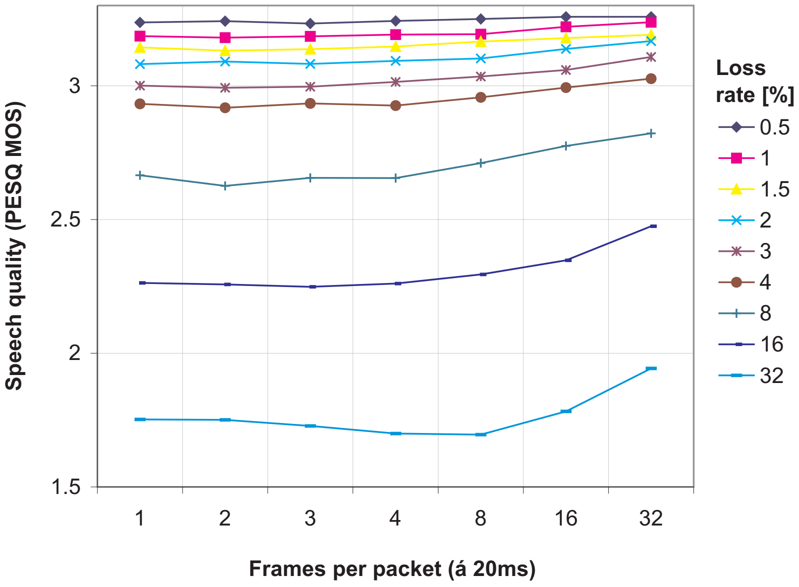
Because playout schedulers adjust the playout time of speech frames, we also measure these impairments. We consider one adjustment within a 8 s sample and distinguish between adjustments during silence (Figure 9.7b) and during voice activity (Figure 9.7c). A *positive* adjustment extends the degraded sample. The resulting gap is concealed by the decoder's concealment algorithm. A *negative* adjustment shortens the degraded sample (the lines are



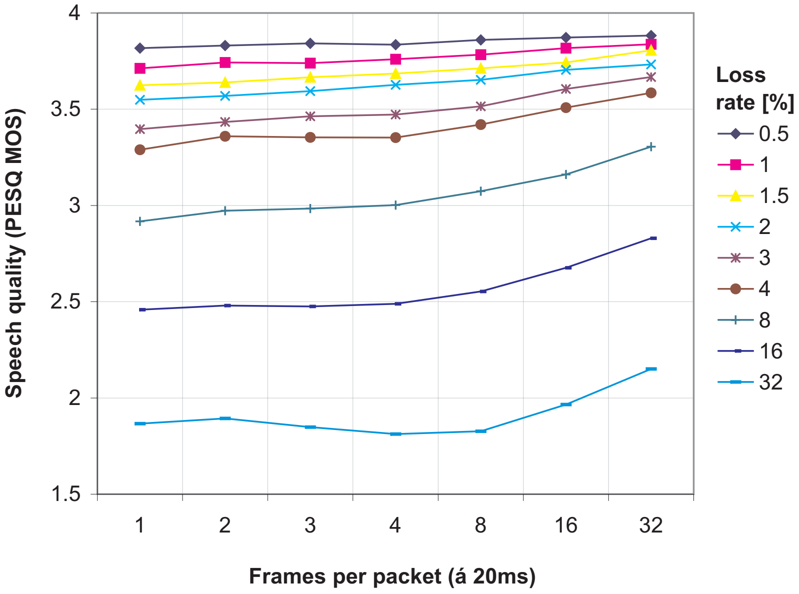
(a) without silence suppression (DTX)

(b) without DTX (straight line) and – slightly lower – with DTX (dotted line)

Figure 9.3.: Impact of coding rate and loss rate.

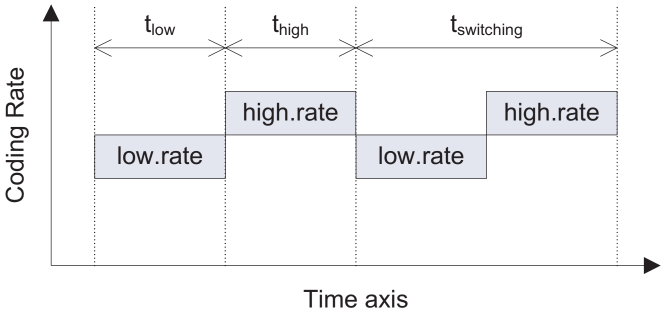


(a) AMR 4.75 kbit/s



(b) AMR 12.2 kbit/s

Figure 9.4.: Impact of packetisation (frames per packet) vs. packet loss rate.



$$mean.rate = \frac{low.rate \cdot t_{low} + high.rate \cdot t_{high}}{t_{period}} \quad \text{and } t_{period} = t_{low} + t_{high}$$

Figure 9.5.: Measuring the impact of switching the coding mode.

named “Adapt”). As a comparison, we also measured the impairment caused by a *loss burst* that has the same length as the positive adjustment’s gap (lines “Loss” in Figure 9.7). During silence, PESQ does not consider adjustments up to one second as harmful. Adjustments during voice activity decrease the speech quality and increase the impairment.

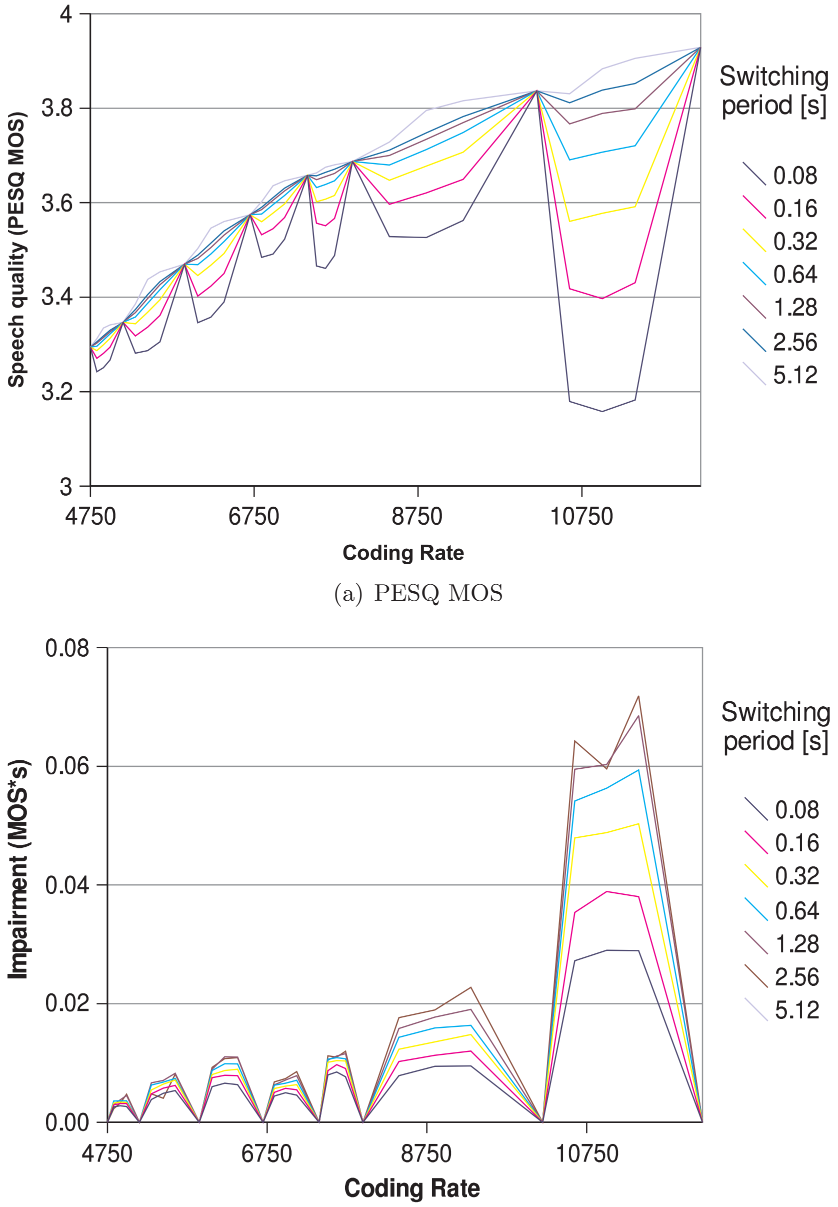
9.4. Adapting coding and packet rate to limited bandwidth

In this example, we apply our quality model to the problem of adapting a VoIP flow to a channel with limited available bandwidth that is described by a maximal data rate. To our best knowledge the problem of how to adapt both coding rate and packet rate to limited bandwidth has never been studied in published literature. Our parameterized quality model allows us to analyse this question. We assume that the capacity of a connection remains constant and is known. The transmission delay of a packet is given and remains constant for each packet. The question to answer is how to choose the optimal coding rate and packetisation under such circumstances. We discuss this issue for a circuit-switched link and a packet-switched, Ethernet-like link, first.

9.4.1. Circuit switched link

Let us assume a channel that has a limited bandwidth and carries one stream of AMR coded frames. If the coding rate exceeds the bandwidth of the channel frames are dropped. The loss rate L depends on the bandwidth of the channel B_c and the bandwidth of the flow B_f, which is equal to the coding rate B_s (see Equation 9.3).

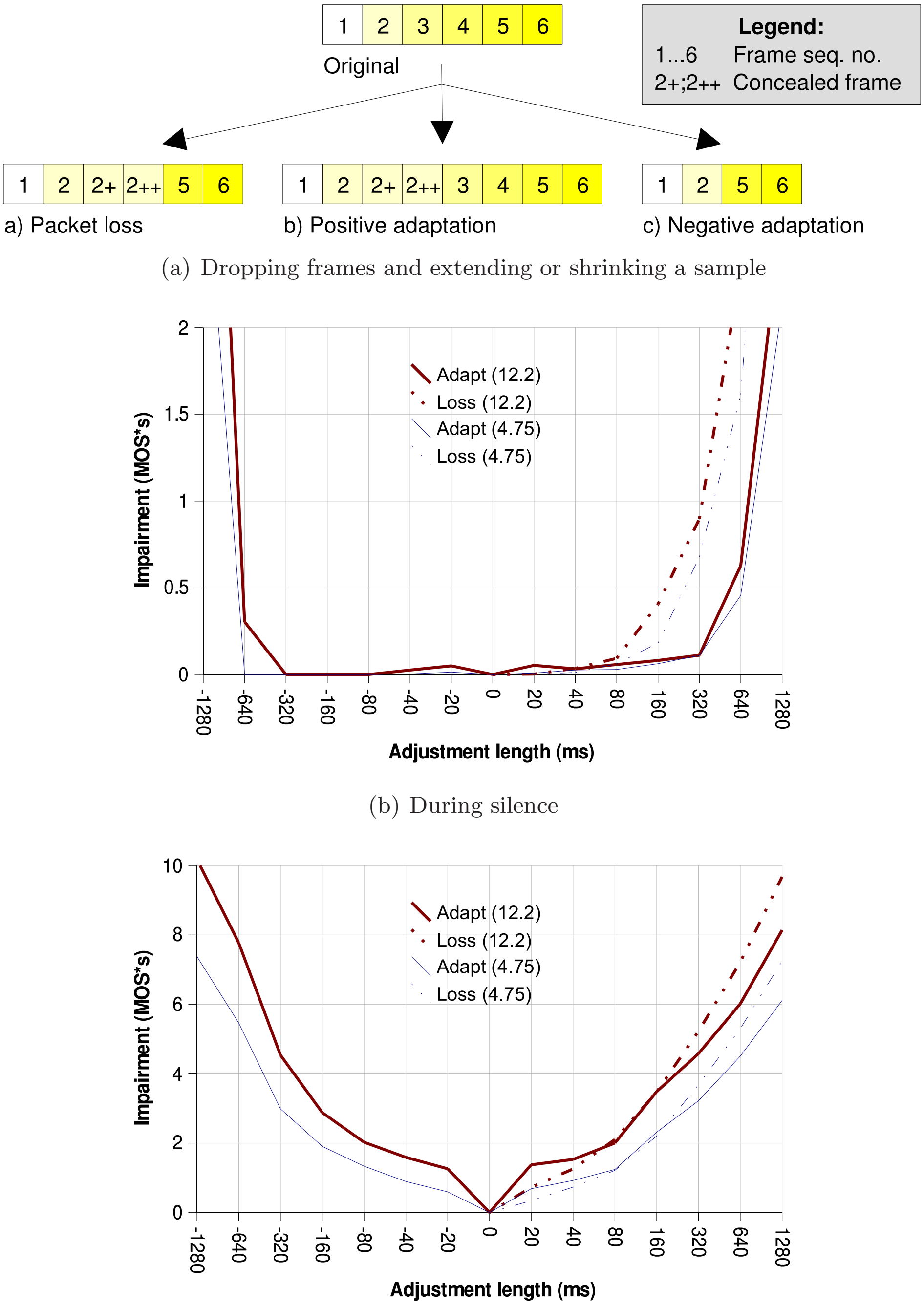
$$L = \begin{cases} B_c > B_f : & 0 \\ B_c \leq B_f : & 1 - \frac{B_c}{B_f} \end{cases} \tag{9.3}$$



(a) PESQ MOS

(b) Impairment MOS*s

Figure 9.6.: Impact due to switching the coding mode (at different frequencies).



(a) Dropping frames and extending or shrinking a sample

(b) During silence

(c) During voice activity

Figure 9.7.: Impact of playout re-scheduling on the importance.

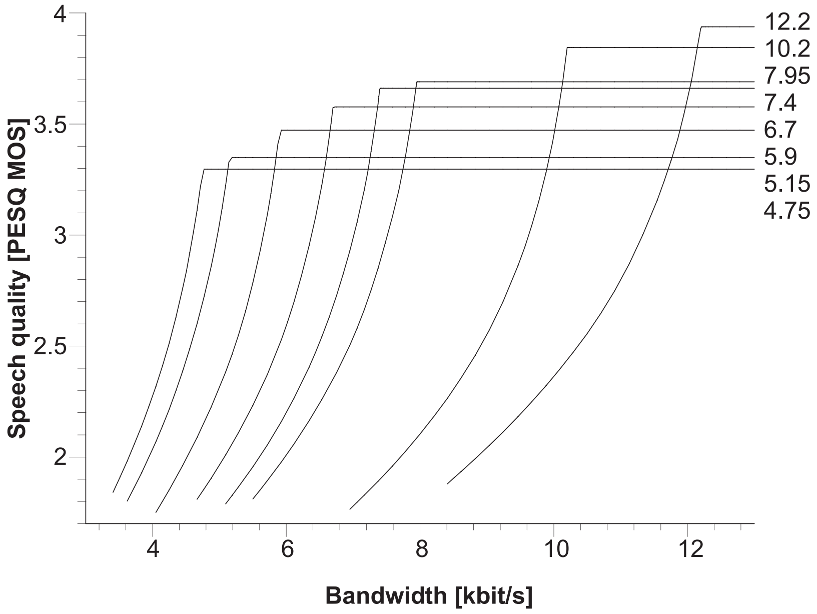


Figure 9.8.: MOS vs. channel bandwidth and coding rate.

Clearly, there is a tradeoff between coding rate and loss because both decreasing coding rate and increasing loss rate will lower the speech quality. In Figure 9.8 the tradeoff in MOS between available the channel bandwidth, the loss rate and the coding mode is displayed taking into account Equation 9.3 and the measurement data of Figure 9.3.

If the loss rate exceeds a value of about 0.5 % (i.e., the available bandwidth B_c is less than coding rate B_s), a better speech quality is obtained by lowering the coding rate. The drop in the MOS value is very sharp if the coding rate exceeds the available bandwidth. As expected, voice flows are highly sensitive to losses and packet losses should be avoided by switching to a lower coding rate.

9.4.2. Full-duplex Ethernet link

Next we assume a full-duplex, switched Ethernet link, which bypasses the CSMA/CD medium access protocol and has a capacity of B_c. Speech frames are generated at a rate of r. A packet consists of f speech frames. In addition, VoIP packets contain protocol headers: The Ethernet header is 26 bytes long (8 bytes preamble, 14 bytes header and 4 bytes CRC), IP (8 bytes), UDP (20 bytes) and RTP (12 bytes). A short header is added (6 bits) in front of each speech frame [192]. The size of a packet p is rounded to the next byte, if its size is a fraction of a byte:

$$p = 8 \left\lceil \frac{628 + (B_s/r + 6)\, f}{8} \right\rceil \tag{9.4}$$

We can calculate the flow bandwidth B_f using the packet size p, the number of frames per packet f and the frame rate r.

$$B_f = \frac{p \cdot r}{f} \tag{9.5}$$

The loss rate depends on the bandwidth of the flow B_f and of the channel B_c as described in Equation 9.3. In addition to the impairment caused by loss, multiple frames in a packet introduce an additional packetisation delay which we have to consider. Thus, we apply Equation 9.1 to take into account both the loss and delay and obtain the following equation. The system delay t_{sys} is the end-to-end transmission delay without any packetisation delay.

$$\begin{aligned} R = \; & \mathrm{MOS_2R}\left(\mathrm{MOS}\left(\mathrm{c}, \mathrm{dtx}, \mathrm{cr}, \mathrm{lr}, \mathrm{pack}\right)\right) \\ & - I_{dd}\left(f/r + t_{\mathrm{sys}}\right) \end{aligned} \tag{9.6}$$

In Figure 9.9, we show the optimum VoIP configuration (as rated by the *R factor*) if both the packet and coding rate are ideally chosen under limited bandwidth. We use the AMR codec (50 packets per second) and 150 and 400 ms as the system delays. The figures show that the packetisation increases if the available bandwidth drops. Only at a very low bandwidth the optimal coding rate decreases, too. In the figure, we do not plot the optimal packet loss rate as it is zero nearly all the time.

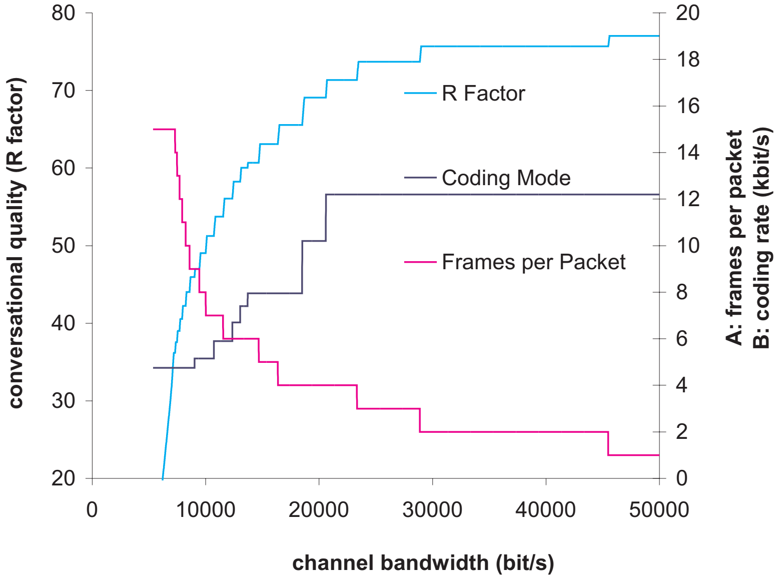
9.4.3. IEEE 802.11b WLAN link

An IEEE 802.11b WLAN link has an even larger packet switching overhead as the Ethernet link. The overhead for each packet has been calculated and validated in [8]. Due to compatibility reasons, the IEEE 802.11b MAC protocol seems to have the highest packet switching overhead of all common MAC protocols. In Table 9.3 we list the packet switching overhead for various transmission modes assuming a long physical preamble and no RTS/CTS. This is only a lower approximation and the packet overhead increases, if the numbers of stations, collisions, or packet errors increase. However, in this chapter these issues shall not be discussed as we are only interested in the tradeoff between coding and packet rate.

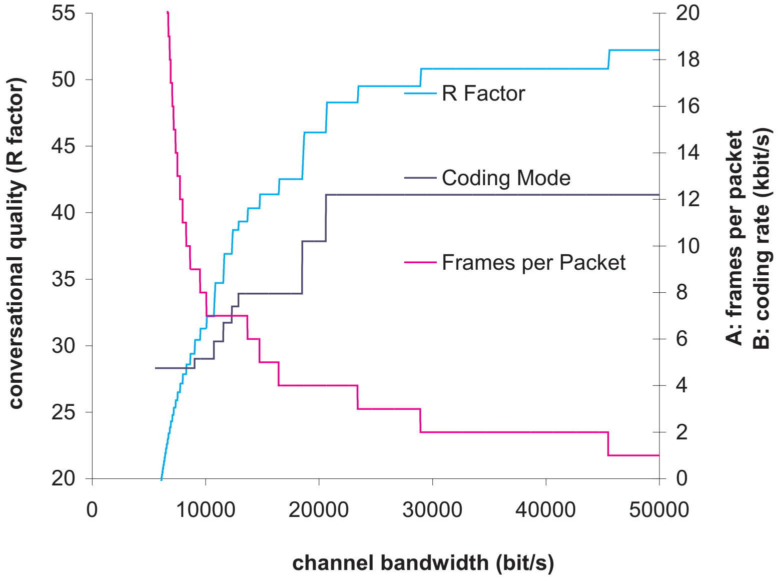
We assume that the WLAN link has a capacity of B_c. Again, speech frames are generated at a rate of r. A packet consists of f speech frames. In addition, WLAN packets have an overhead of approximately 1500 bytes.

$$p \approx 1200 + f \cdot \frac{B_s}{r} \tag{9.7}$$

The following calculations are done as described in previous section but using Equation 9.7 instead of Equation 9.4. In Figure 9.9, we show the optimal VoIP configuration for an IEEE 802.11b wireless link.

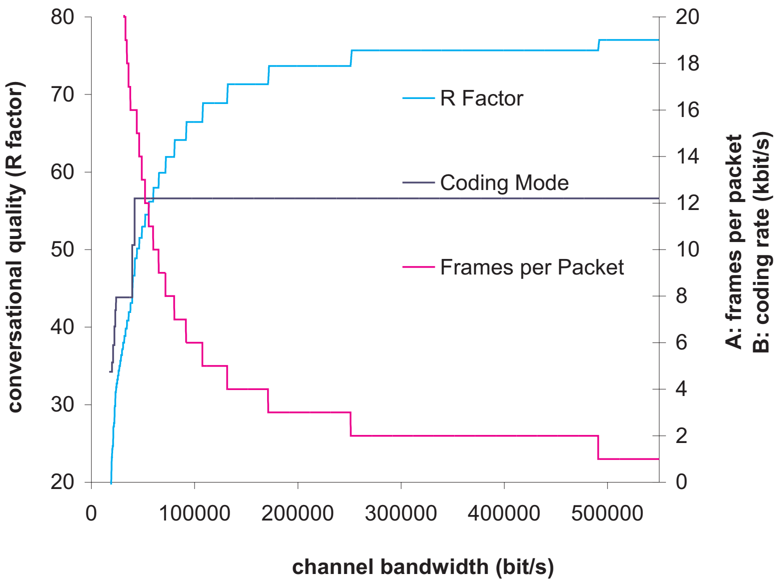


(a) System delay: 150 ms

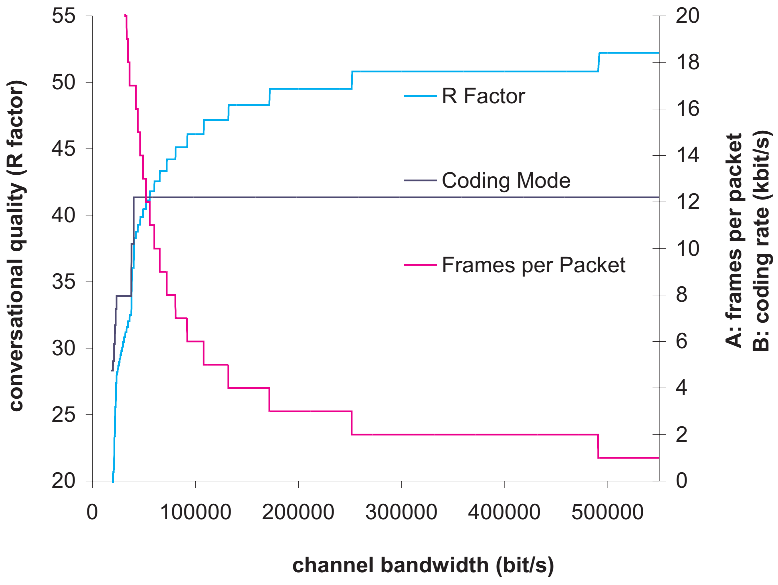


(b) System delay: 400 ms

Figure 9.9.: Choosing optimal coding rate and packetisation for a packet-switched link.



(a) System delay: 150 ms



(b) System delay: 400 ms

Figure 9.10.: Choosing optimal coding rate and packetisation on a WLAN link.

Table 9.3.: Overhead in microsecond and bytes per UDP/IP datagram for the IEEE 802.11 platform as given in [8].

Mode [Mbps]	Packet headers, 82b [μs]	ACK overhead, 8b [μs]	Physical overhead [μs]	Overall [μs]	Overall [b]
11	59.6	56	754	869.6	1196
5.5	119.3	56	754	737.3	639
2	328	56	754	946.0	284
1	656	112	754	1522.0	190
Consists of	RTP[12b], UDP/IP[28b], SNAP[8b], MPDU[30b], FCS[4b]=82b	ACK[8b]	2*PLCP[192μs], DIFS[50μs], SIFS[10μs], Backoff[310μs]	Packet headers, ACK, Physical overhead	Packet headers, ACK, Physical overhead

9.4.4. Discussion

In cases of limited bandwidth, the question of which parameter to adapt depends on the underlying networking technology. In case of a circuit-switched link without any packet overhead, the coding rate should be lowered. With a link technology that has a large packet switching overhead, the packet rate should be lowered but the coding rate should remain the constant.

This leads to the conclusion of how to support congestion control for low-rate VoIP. If the bandwidth is too low for even narrow-band telephony, instead of changing the coding rate, the packet rate has to be reduced. We have conducted further studies on these issues in [137, 138] to support an adaptive coding/packet rate VoIP application, which behaves TCP-friendly if congestion is present.

9.5. Reacting to delay spikes?

As an example of how to use our quality model on the sink side, we consider an open issue in the design of adaptive playout algorithms (compare to Section 4.3). The size of a playout buffer should be chosen in a way that both the number of late frames and the additional delay imposed by the buffer are low. Common playout buffer algorithms adapt the size of the playout buffer to the transmission history in order to find an optimal trade-off between loss and delay. However, analysis of Internet traces shows that packet delays can show sharp, spike-like increases in delay that cannot be predicted in advance [142, 173]. After a spike, packets are received at a high frequency. Soon afterwards, the jitter process returns to normal (Figure 4.9). We consider the question of whether it is advantageous to delay the playout of speech frame to include such spikes or to drop late packets (Figure 9.7). We concentrate on

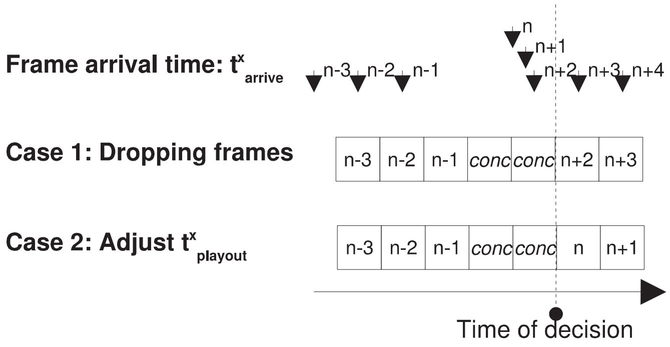

Frame arrival time: t^x_arrive
n-3
n-2
n-1
n
n+1
n+2
n+3
n+4
Case 1: Dropping frames
n-3 n-2 n-1 conc conc n+2 n+3
Case 2: Adjust t^x_playout
n-3 n-2 n-1 conc conc n n+1
Time of decision

Figure 9.11.: Playout scheduling strategies in reaction to delay spikes. Drop frames or delay the playout?

the non-trivial case of delay spikes during voice activity and use the quality model introduced in this chapter.

Frame F_n arrives too late to be played out on time. No consecutive frames F_i with $i > n$ have been received so far. The scheduled playout time of frame F_n is $t^n_{playout}$, but the frame has arrived at $t^n_{arrive} > t^n_{playout}$. At the arrival time, the decoder has already concealed all frames F_i with $t^i_{playout} < t^n_{arrive}$ as they have been considered as lost. Should the playout times be increased by $t_{gap} = t^n_{arrive} - t^n_{playout}$ temporarily so that the late frames are still played out?

Since adjustments have a different impact according to the current speech property, it is important to know whether the F_i frames $(i > n)$ contain either silence or voice. The voice activity of frame F_n is known as it has already been received. Thus, we know the speech quality impairment of the adjustment, which delays the playout.

But when to re-adjust the playout to its previous value again? Clearly as soon as the voice falls silent the playout should be changed as during silence the adjustment is not audible. But how long will the talker speak? The speech properties of the consecutive frames are not known, since they have not yet been received.

But there is hope in statistics: Brady discovered that both the talk-spurt and silence periods of digitised voice can be approximated by an exponential distribution [31]. A commonly accepted and standardized [99] model for artificial voices is a continuous-time, discrete state Markov chain with two states referring to talk spurt (ON) and silence (OFF) periods. The holding time in each state is exponentially distributed with mean $1/\lambda$ and $1/\mu$. Hence, the transitional rates from the ON to OFF state is λ and μ, respectively. We apply this model to predict when a negative adjustment can be made. (For simplicity reasons, we assume in the following that the next silence frames will be in exactly 1.004s.).

To calculate the quality rating of a delay spike without an adjustment of the playout buffer time, we apply the ITU E-Model. The R factor is calculated from the speech quality mea-

State	Mean Duration
talk-spurt (ON)	$t_{\text{on}} = \mu^{-1} = 1.004\text{s}$
silence (OFF)	$t_{\text{off}} = \lambda^{-1} = 1.587\text{s}$

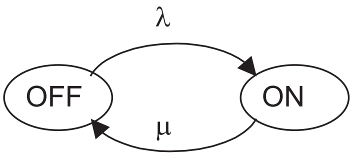


Figure 9.12.: Voice model: two-state Markov model.

surements (MOS$_{loss}$ from Figure 9.7c) and the mouth-to-ear delay (t_{m2e} is usually estimated or measured):

$$R_{loss} = \text{MOStoR}\left(\text{MOS}_{loss}\left(t_{gap}\right)\right) - I_d\left(t_{m2e}\right) \tag{9.8}$$

To calculate the quality rating of the adjustment, we use the MOS_{adapt} results from Figure 9.7c, which refer to samples with a length of $t_{sample} = 8\,s$. To calculate delay impairment, we sum and weight the quality of the adjusted period and the normal period. The quality impairment of the adjustment during silence is not considered as it is virtually zero:

$$R_{adjust} = \text{MOStoR}\left(\text{MOS}_{adapt}\left(t_{gap}\right) - I_d^{mean}\right) \tag{9.9}$$

with

$$I_d^{mean} = I_d\left(t_{m2e}\right)\frac{t_{sample} - t_{on}}{t_{sample}} + I_d\left(t_{m2e} + t_{gap}\right)\frac{t_{on}}{t_{sample}}$$

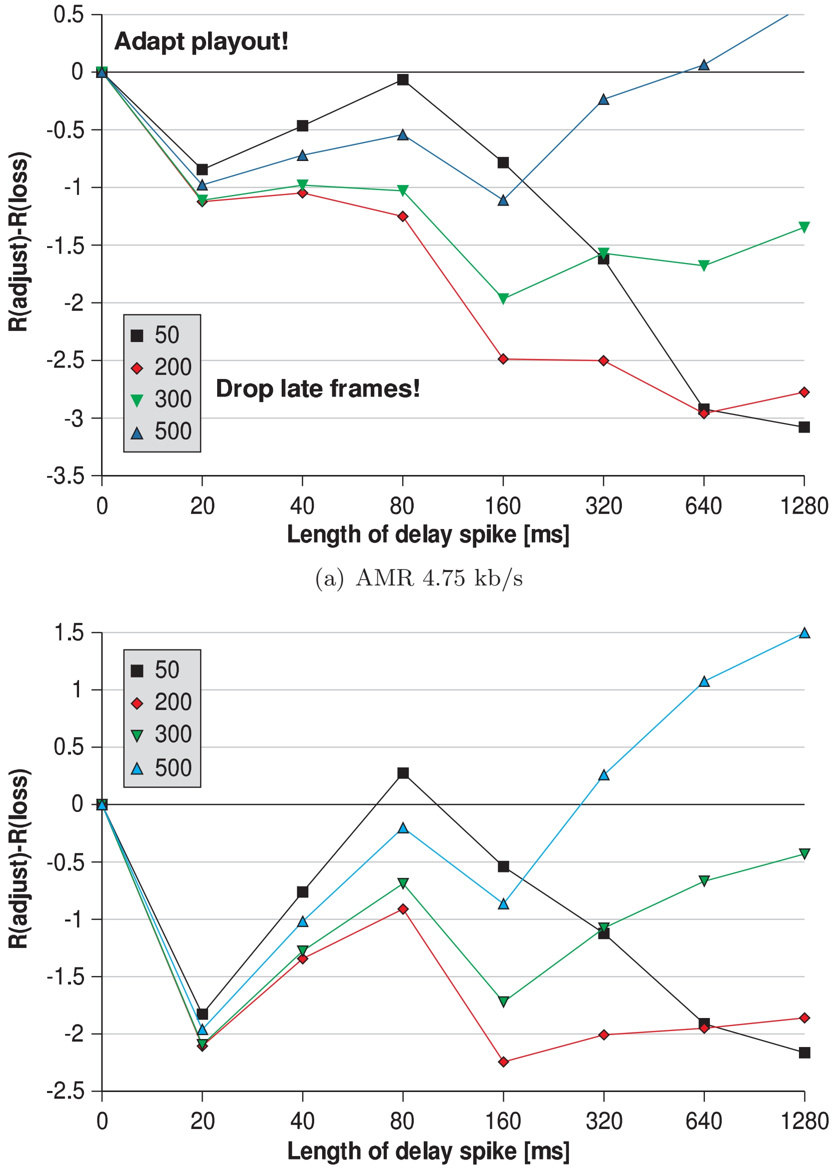
In Figure 9.7c we have shown that the rescheduling of speech frames harms the speech quality less than losing the frames as long as the gap is larger than 80 ms. In Figure 9.13 we also consider the impact of transmission delay and calculated $R_{adjust} - R_{loss}$ for different gap lengths and mouth-to-ear delays.

It can be seen that frames delayed by delay spikes in general should be dropped and an adaptation is not required as most R value differences are below zero. However, one should consider that this result is valid only for single delay spikes. Often a delay spike is only a first indication of a up coming period of further high transmission delays [142]. Further studies are required to identify the impact of multiple delay spikes that occur shortly one after the other.

9.6. Summary

If a quality model is being developed, the area and context of its application is highly important and has a large impact on design decisions: We presented a new quality model for telephony. Its main purpose is to parameterize adaptive VoIP applications and algorithms so that they can achieve high perceptual quality ratings.

One of the conclusions of this study is that the coding mode should not be switched too



(a) AMR 4.75 kb/s

(b) AMR 12.2 kb/s

Figure 9.13.: Whether to adjust the playout to late packets or drop late packets.

often as it deteriorates speech quality. Consequently, media-dependent FEC is not advisable. Media-dependent FEC tries to improve speech quality by switching to another coding mode. However, switching the coding mode reduces speech quality as it introduces clicking sounds.

We demonstrated that as soon as bandwidth is limited it is more efficient to change the packet rate rather than the coding rate. Previous approaches to rate-adaptive voice only considered the coding rate. This has interesting consequences for the semantic data-link: It shows that frame dropping strategies might be more important than changing the coding mode (refer to the discussion in Section 5.6).

Our results also indicate that a playout buffer should not adjust its playout to delay spikes if they occur singular. Thus, a data-link might not need to retransmit a late packet until it has been received error free but can be dropped after a certain period.

It is a common, yet false, opinion that a larger packetisation (longer packets) is worse than short and many VoIP packets in cases of packet loss. Our results indicate that losing one large or multiple short VoIP packets has about the same impact on the impairment of speech quality (refer to Section 6.3.3).

One should consider that the measurement results of our work are based on an objective perceptual model which only approximates the real rating behaviour of human beings. Thus, subjective tests are required to verify and to enhance the accuracy of these results. Partly, such tests have been conducted in Chapter 4. However, further, more focused formal conversational call quality tests are required.

10. The Impact of Slow User Motion

IEEE 802.11b compliant WLAN technology is increasingly being used for cordless telephone services. Often, a WLAN phone is moved during a call. In this chapter we explore to what extent slow user motion influences the wireless link quality. We conducted extensive measurements with speech over commercial IEEE 802.11b equipment using an experimental environment enforcing controlled motion. Between two nodes, a base station and a mobile node, bidirectional VoIP flows are generated. We measure the packets' transmission success and delay with our extended device driver. We alter the location, speed, and direction of movement of the mobile node. Different channel modulation types (automatic, 1 and 11 Mbps) and different numbers of maximal MAC layer retransmissions are considered. The measurements were conducted in both an office and a large gymnasium.

The results of this work are intended to parameterize the error model in wireless network simulations. In addition our experiments show that – in contrast to the common conception – an increase in the motion speed can result in a better link quality: The packet loss rate and its variance, if they are measured after link-layer retransmissions, decreases.

The main contributions of this work is a systematic approach to how to conduct experimental measurements, is the development of WLAN measurement software, and is the implementation of a device driver supporting semantic data-flows. Last not least, the experimental results are used in Chapter 12 for location sensing applications.

This chapter is structured in the following way. Section 10.1 describes the measurement software and hardware setup. In Section 10.2 we present and discuss the measurement results. Finally, we summarize this chapter and draw conclusions.

10.1. Measurement setup

When the link characteristics for Internet Telephony are being studied, it is important to construct usage scenarios, which are similar to how humans conduct telephone calls with cordless phones. The usage of a cordless phone is often limited to a small area, e.g. a room or a building. We limited our study by not including handovers and are primarily interested in the effect of human movements, which are rather slow (e.g. 1 m/s).

One important requirement is that the measurement results are to some extent reproducible. Even though a wireless link has a random and chaotic nature caused by many factors that influence the link quality, it should be possible to reproduce similar measurement efforts.

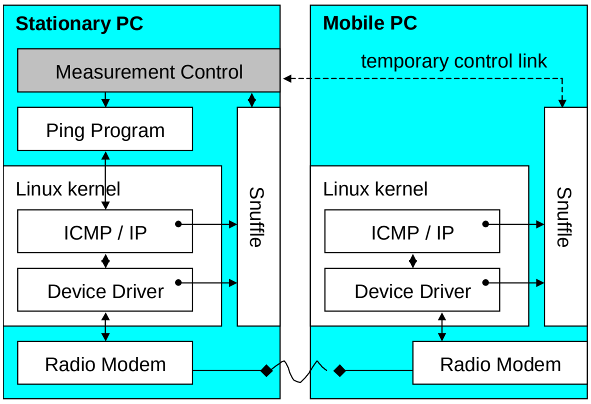


Figure 10.1.: Measurement software.

Furthermore, because of high variability in wireless links, the measurements should sufficiently long to obtain statistical stability in the results. It is clear, that humans cannot be used to move the client, if both the measurements should be reproducible and last for some hours. Therefore, a mechanical *node mover* had to be constructed. In the following, we describe the measurement set-up in a top down approach (see Figure 10.1).

VoIP traffic generator: During all measurements a *simulated telephone call* between two participants was generated. The traffic is carried between one *stationary PC* (referred as the base station) and the *mobile PC* (WLAN phone) over IEEE 802.11b. A VoIP stream consists of a bidirectional flow of packets, which contain the audio data. To simulate such a traffic pattern, we used the "ping" program. The stationary host generates ICMP echo request messages, which the mobile PC answers with an ICMP echo reply.[1] During the measurement the ping program transmits a 20 ms a packet with a length of 78 bytes (including IP headers). This packet size corresponds to a VoIP flow encoded at a rate of 8 kbps. We use the IP's type of service (TOS) field to dynamically change the link layer transmission mode.

IEEE 802.11b Data-Link: For our measurements, we used the PCMCIA reference design of Intersil's Prism2 chipset, which is implemented by ZoomAir and D-Link cards. It was selected as the radio modems comply with the IEEE 802.11b specification. Furthermore, a detailed description of the medium-access-controller can be obtained from Intersil, which allows setting up different transmission configurations. There is stable public source driver support for the Linux Operation System, which facilitated the setup of the measurement environment. We used the Host AP driver [139], version 2.5.2002. We added measurement tracepoints to collect received and transmitted messages and radio-modem specific protocol states.

We changed the device driver to support flow-specific link layer configurations [84], so that

[1] One should note that if the ICMP request message is lost, no ICMP answer message is sent.

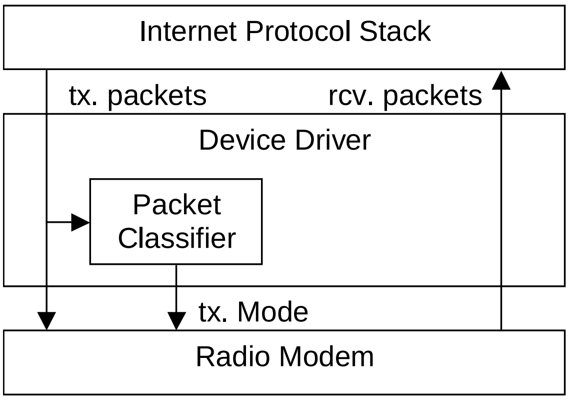


Figure 10.2.: Packet classifier in device driver

Table 10.1.: Flow-specific error control.

Packet type	Modulation type	Maximal number of retransmissions
IPv4, ICMP, IP TOS=1	1 Mbps	8
IPv4, ICMP, IP TOS=2	11 Mbps	8
IPv4, ICMP, IP TOS=3	1 Mbps	0
IPv4, ICMP, IP TOS=4	11 Mbps	0
All other packets	Automatic	Automatic

the error control is changed in according with the current flow classification (see Figure 10.2). Each packet in the device driver that is considered for transmission is analyzed with a packet classifier. The packet classifier looks at the packets' protocol headers. If a packet is identified as an IPv4 and ICMP packet, the error control is changed according to Table 10.1.

Measurement tool Snuffle: To record the packets' transmission, we developed a measurement tool called Snuffle [80, 174]. Snuffle consists of two components: a user-space program to collect and store the measurement data and a kernel extension, which traces how packets traverse through the protocol stack. It can trace internal protocol states, too. During the measurement efforts, ICMP and IEEE 802.11b MAC packets were captured on both hosts.

Experimental controlled motion: Our experimental node mover is a large model train that carries a notebook (Figure 10.3). The locomotive moves along a curved or straight rail, which are both 5 meters long. At the ends of the rail, the locomotive stops automatically and starts back the rails in the reverse direction. The maximum speed of the train is approximately 1 m/s. However, the relative speed between the stationary PC station and mobile PC varies between 0 and 1 m/s, depending on the position and the angle of the rails.

To ensure power to the notebook especially for long measurements, it was necessary to use

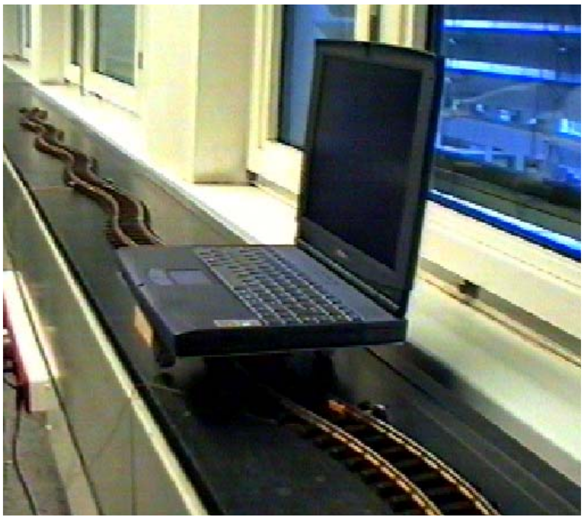


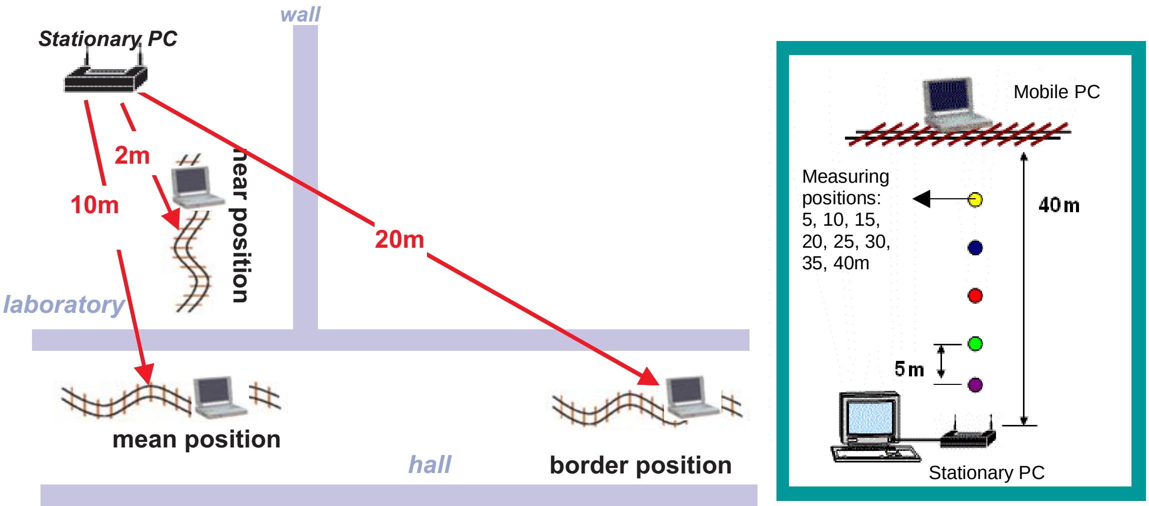
Figure 10.3.: Node mover.

an external power supply as the capacity of the build-in batteries is not large enough. First, we connected the notebook with its power supply unit by a long cable. But this solution was not reliable as the cable had to be pulled behind the locomotive and sometimes got entangled with the rails. To overcome this problem we used additional rails, which were placed between the primary rails. These rails were used as a permanent power supply for the notebook. Two sliding contacts beneath the locomotive connected the notebook with the secondary rails.

Locations: To study the dependence of transmission quality on location, we placed the rails at three different positions within a ferro-concrete building (Figure 10.4a). The first position was close to the stationary PC at a range from 1 to 5 meters. Few transmission errors are to be expected in this case. The second position was at a distance between 8 and 12 m. The third positions were at a critical distance (18-22 m) as the link quality starts to deteriorate. Ferro-concrete walls were between the base station and the second and third position. In order to be independent from the impact of the walls we performed additional measurements in a gymnasium. This was to explore the influence of distance, speed and moving direction on the link quality (Figure 10.4b), if the connection has pure line-of-sight.

Analysis: Each measurement, which lasted several minutes to some hours, was divided into intervals of 128 seconds, which is a length of a typical telephone call. For each telephone call we calculated the mean packet loss rate, which is the number of transmission failures as notified by the stationary PC. Transmission failures occur, if the stationary PC does not receive any immediate acknowledgement even after multiple retransmissions.

If the packet was transmitted successfully we subtracted the finishing timestamp from the



(a) Office

(b) Gymnasium

Figure 10.4.: Measurement locations of the WLAN measurements.

starting time. Both timestamps are measured on the stationary PC by the Snuffle program. The delay, being the difference between the timestamps, includes the queuing delay, the transmission duration of the packets (data and acknowledgement) and the MAC access delay multiplied by the number of transmission attempts.

Due to the automatic selection of the modulation type, and due to multiple transmissions attempts, this duration can vary to a large extent as we will see in the results.

Timestamp accuracy: Timestamps are measured in an operating system, which has a measurement inaccuracy due to interrupt latencies and the lack of real time support. To quantify the interrupt latency, we measured the arrival times of packets. Every 20 ms WLAN packets were received. The arrival times were measured at two positions: First, in the radio modem with a built-in clock. Next, at the interrupt handling routine in the operating system using the kernel system call “gettimeofday”. The clock in the radio modem is not subject to jitter. Therefore, we used this as a reference clock. The radio modem clock and the operating system clock have a drift or frequency offset of approximately 0.38%. For the following comparison we corrected both the frequency offset and an absolute offset. Then, for each received packet we calculated the difference between corrected radio modem timestamp and the OS timestamp. Figure 10.5 shows the distribution of these differences and thus the resolution of the OS clock. Approximately 95% of all measurement values have an error of less than 0.2 ms and 90% less than 0.1 ms.

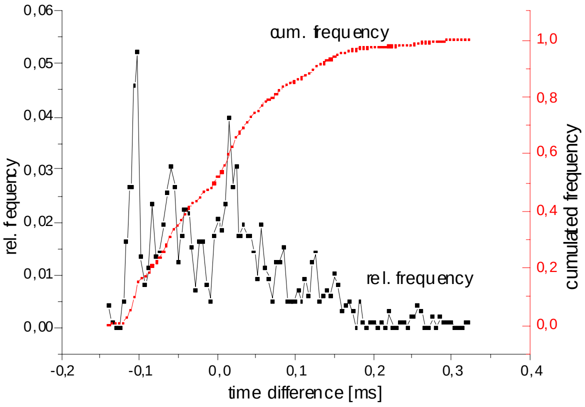


Figure 10.5.: Difference between OS clock and radio modem clock.

10.2. Results

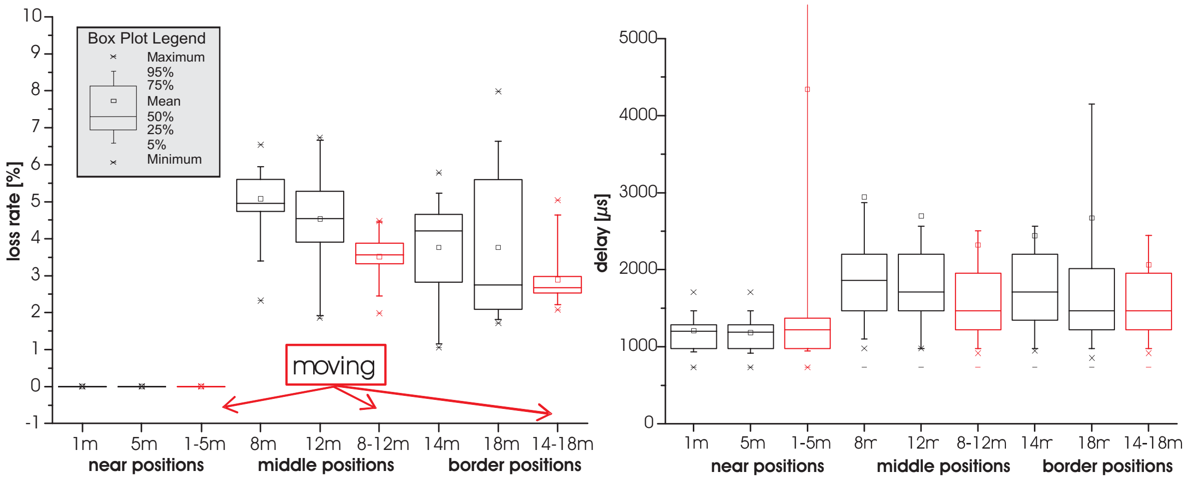
We started three measurement efforts to identify the relationship between slow user motion and link quality. Starting from a black box approach using a standard radio modem configuration and measurement set-up, we tried to remove in the following measurement phases all obstructions to obtain a better understanding of the system behaviour. In the following, we will describe the results obtained by these measurement efforts. A more detailed description of the measurement results is given in [86].

10.2.1. Measurements in the office

We conducted our first measurements with the default configuration of the radio modem, which was adaptive rate selection, DCF MAC mode, no fragmentation, maximal eight immediate MAC retransmissions, and no RTS/CTS. A stationary PC was configured as a base station (master) whereas the mobile PC operated as a client.

The adaptive rate selection automatically switches to the next lower modulation rate after transmission errors have occurred. The supported rates are 1, 2, 5.5, and 11 Mbps. If the link is error-free for some period of time, the modulation rate is increased again. The particular algorithm depends on the firmware implementation of the radio modem and is not standardized.

We conducted nine measurement series, each lasting one hour. The mobile node was placed at different positions, which were roughly 1, 5, 8, 12, 14, and 18 meters from the stationary PC (see Figure 10.4). We moved the mobile node at slow speeds between the ends of the rail. Figure 10.6a shows the packet loss rate as measured at the networking layer. The box plots display the statistics from the simulated telephone calls. They contain the mean, median, percentiles and extremes of the calls' loss rates. We measured the delay (Figure 10.6b) for each successfully transmitted packet. This delay includes the queuing delay (if any), the



(a) IP packet loss rate

(b) Packet delays

Figure 10.6.: Office: Loss and delay vs. position and motion, default radio configuration.

transmission delay of its data packet and the acknowledgement, the scheduling delay and the interrupt latency.

If the distance between the base station and the node is small, no packet losses are observed at the networking layer. As the distance increases and a wall separates the BS and client, the loss rates to 2 or 5%. The delay also increases [86]. This measurement data suggests that if the client moves, the loss rate is slightly lower than in the stationary positions and the link quality is more stable and less variable.

To understand the transmission process of the MAC protocol we looked at the delay distribution (Figure 10.7). Most packets (98%) are transmitted within 5 ms. If we compare the delays at different distances, we see that the further the node, the longer a transmission takes. This is understandable, because the radio modem selects a lower rate as soon as the link quality deteriorates. Furthermore, we can see that few packets are transmitted multiple times, if their transmission fails at the first attempt.

10.2.2. Measurements in the gymnasium

The second measurement effort, we changed the location and moved the equipment to a large, empty gymnasium in a grammar school (Figure 10.8). The rails were straight and the mobile PC's power was supplied by additional rails and not by a cable. We altered the link layer configuration and changed the modulation type (1, 11 Mbps and automatic rate selection) and the number of maximal retransmission attempts (0 and 8).

First, we studied the impact of distance on the link quality. We measured the loss rate and delay at each position for approximately 15 minutes. Figure 10.9a shows the packet loss rate versus the distance at the 11 Mbps modulation and no ARQ. The transmission range in the

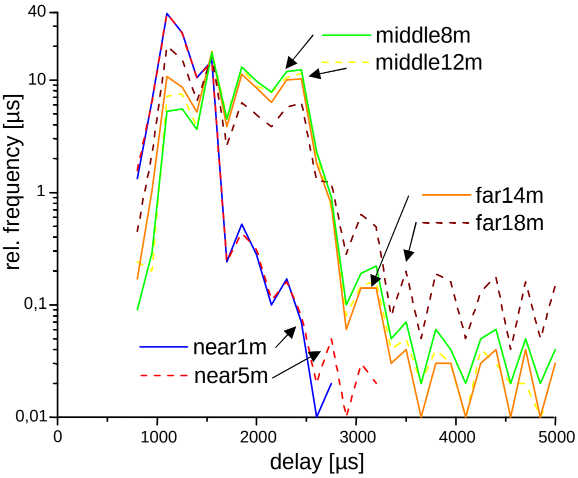


Figure 10.7.: Office: Histogram of transmission delays.
The transmission delay depends on the transmission rate and on the number of transmission attempts.

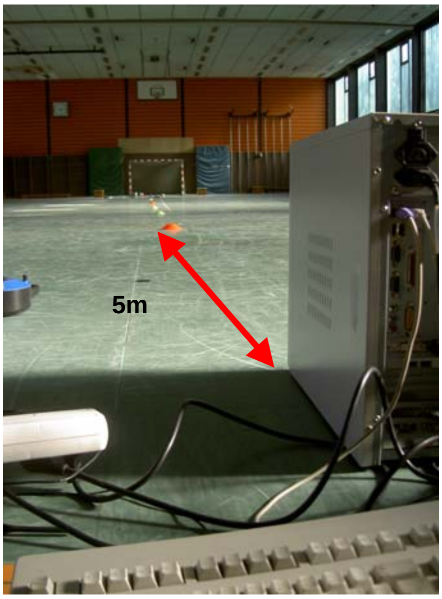
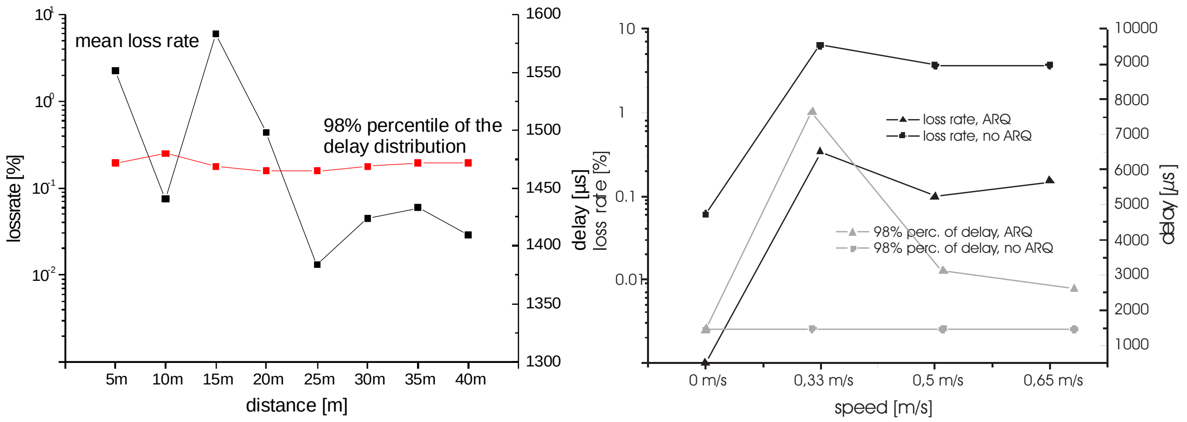


Figure 10.8.: Measuring setup in the gym.



(a) Loss rate vs. distance, no ARQ.

(b) Loss and delay vs. motion speed

Figure 10.9.: Gymnasium measurement results (11 Mbps).

empty hall is much wider than in the office environment. Even at 40 m the link quality is quite good. However, at 5 and 15 m we measured some high packet loss rates.

To calculate mean delay, we omitted 2% of packets with the highest transmission delay, because some delays are very high (>20 ms) so that they would falsify the results. As expected, the delay remains constant due to a fixed modulation scheme and only one transmission attempt (Figure 10.9b).

Next, we measured the impact of the motion speed on the packet loss and delay. We altered the voltage of supply so that the train moved at an average speed of 0.33, 0.5 and 0.65 m/s. The distance of the mobile node was 35 m; the direction of motion was perpendicular to the main wave propagation. If the node does not move at all, the loss rate is lower than during movement. The faster the mobile PC moves, the fewer retransmissions occur (Figure 10.9 and Figure 10.10). The delay without ARQ is constant for all speeds. With ARQ the delay is highest at 0.33 m/s and decreases at increasing speeds.

During movement of the mobile PC, the PC receives its power from secondary rails. Either due to electromagnetic interference, a low quality power supply or short interruption of the power supply, the radio modem performed not optimally. For the perpendicular 30 to 35 m measurements we conducted the same measurement once with a normal power supply and once with battery power. With a normal supply the mean packet loss mode in the 11 Mbps mode was 5.2%, during a battery supply it was 3.6%.

To increase the statistical accuracy we combined all the measurement data, which were gathered in the gymnasium. We distinguished only between the link layer mode and the degree of movement (stationary or moving). Figure 10.11 shows that movement harms the link quality: The packet loss rate increases regardless of the modulation mode or ARQ type.

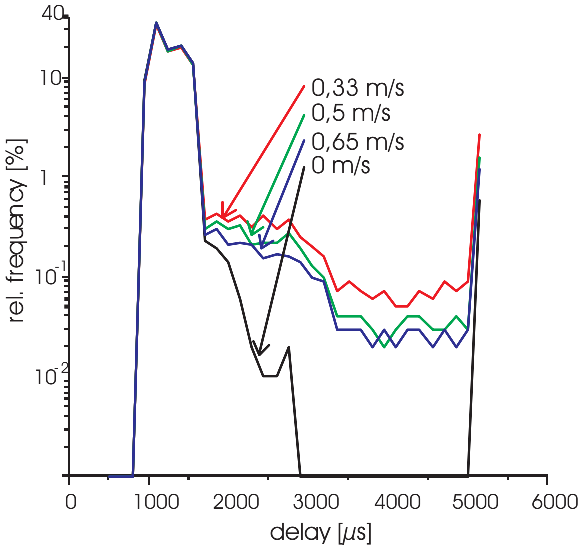


Figure 10.10.: Gymnasium: Delay histogram vs. motion speed, 11 Mbps, ARQ.

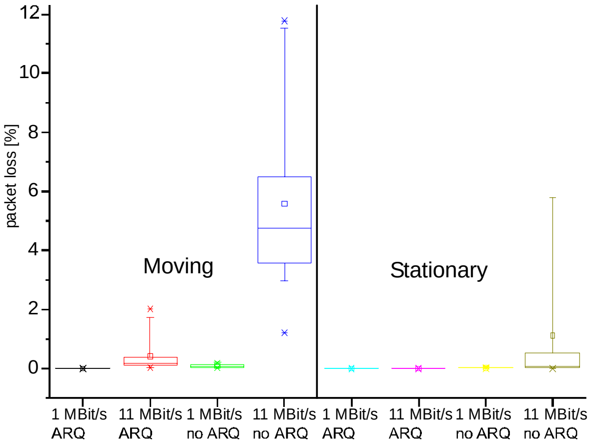


Figure 10.11.: Gymnasium: Combined results.

10.2.3. Summary

As the distance and the attenuation increase the packet loss rate increases too (Figure 10.6). But even if the distance is short and line-of-sight, the link quality can be poor. As long as a wireless connection can be established, the packet loss rate after error control is mostly below 0.1%, even in a "bad" position the loss rate seldom exceeds 5%.

The packet loss rate depends on the modulation type and the number of retransmissions. 1 Mbps is, as expected, more reliable than 11 Mbps. The 1 Mbps mode does not require many retransmissions.

For a fixed packet size and no background traffic, the transmission delay depends on the modulation type and the number of retransmissions. As expected, the transmission delay increases if ARQ is switched on. Measuring the transmission delay also gives an indication of the transmission mode used and current link quality.

During movement, however, if power is supplied using the secondary rails, the link quality is worse than when using the batteries. This effect influenced strongly the measurement results of the second measurement effort. Thus, having an undisturbed power supply is essential.

Movements in the gymnasium worsen the link quality so the loss rate increases. The motion in the office environment has – at least in some measurements – a positive effect on the link quality. Both the loss rate and delay were reduced and the link quality was more stable. The effect was more noticeable in the office environment, if the client and BS were separated by walls. In that case, multiple waves propagate over different paths causing interferences and a Rayleigh fading channel [111]. Also, the positive effect of movement was only observed if the automatic rate selection was active.

10.3. Conclusion

In this chapter, we describe an experimental set-up and methodology for the measurement and performance evaluation of an IEEE 802.11b compliant radio modem. The applied software (Snuffle) is able to trace protocol messages and protocol states. Using this software; we were able to measure the packet loss and delay of IEEE 802.11b WLAN links using commercial radio modem cards.

Concluding, we can say that factors that are more important than the motion speed are the modulation type, the number of retransmissions, the attenuation, the environmental setting and the quality of power supply. The link quality of real wireless systems depends on many different factors, which are non trivial to control.

We also implemented a data-link protocol, which supports per-packet QoS. Such an implementation can be used for semantic data-link protocol. Finally, we conducted measurements that are used in Chapter 12 to study the relationship between round trip times and distance.

11. VoIP over Wi-Fi

As IEEE 802.11-based Internet access is becoming ubiquitously available, it is excepted that WLAN will be used for telephony. In this chapter we follow the objective to access whether WLAN offers *toll quality* calls. Toll quality is the minimal requirement on PSTN-based telephone calls. Via simulations we evaluate the distribution coordination function (DCF) protocol, enhanced distribution coordination channel access (EDCA) and contention free bursting (CFB). We have verified our simulation model from cross-checking it with the results of other researchers. We present precise, quantitative results of the perceptual quality of Voice over Wi-Fi systems.

11.1. Introduction

The IEEE 802.11 standard supports two MAC mechanisms, the Distributed Coordination Function (DCF) and the Point Coordination Function (PCF). These mechanisms are considered to be insufficient for achieving reasonable quality [124] in situations with high background load. Thus, QoS enhancements are intensively studied and evaluated. Currently, the QoS enhanced IEEE 802.11e is under design and in the standardization process [96]. IEEE 802.11e introduces two additional MAC modes: the Enhanced Distributed Channel Access (EDCA) and the HCF Controlled Channel Access (HCCA).

In this chapter, we present an open-source simulation model of EDCA mode for the network simulator ns-2 [213]. Our 802.11e EDCA model includes contention free bursting (CFB, sometimes also called TXOP bursting), which allows the transmission of a series of small packets without intermediate contention. We verified our model by comparing it with previous published results [140]. In the mean time, the correctness of this model have been proved and improved in approximately 10 publications [214].

We apply our simulation model to evaluate the quality of telephone calls when using the EDCA model. To measure the call quality we apply the approach described in Chapter 4. Using this quality model, we are able to evaluate with high precision to what extent various MAC protocol modes are suitable for telephony.

Our simulation scenario is a Basic Service Set (BSS), which consists of a base station and multiple wireless nodes. Bidirectional streams of Voice over IP (VoIP) packets model the telephone calls. Our simulations address the following issues:

- First, we are interested in the effect of best-effort background traffic on voice transmissions. In parallel to a voice call, we transmit UDP or TCP flows with the maximum possible rate. TCP deteriorates the quality of the voice streams less than UDP, as it slows down its sending rate. In DCF mode, UDP competes with voice streams that are transmitted in the same direction. But EDCA and EDCA+CFB enable data and voice traffic at the same time and in the same direction.
- Secondly, we measure the throughput of TCP and UDP with DCF and EDCA without any voice transmission to obtain an idea about the efficiency of EDCA. With EDCA the throughput of the background traffic is lower as it is subjected to a longer contention period.
- Last, we consider how many simultaneous telephone calls are supported at which quality level. We select the different MAC mechanisms DCF, EDCA, and EDCA+CFB and measure the perceptual telephone quality for the uplink and downlink. On the uplink, the basic DCF can transmit the highest number of telephone calls, followed by EDCA and EDCA+CFB. On the downlink, EDCA and EDCA+CFB perform best.

This chapter is structured as follows: First, we describe the ns-2 simulation model and its verification. Next, we describe our VoIP simulations including the simulation scenarios, results, and analyses. Finally we conclude and give an outlook on possible, further research directions.

11.2. Simulation and evaluation environment

11.2.1. IEEE 802.11 EDCA simulation model

We used the discrete event simulator ns-2.26 [204] for our work in which an 802.11 DCF model is already included. The ns-802.11 model does not provide PCF or any MAC-management mechanisms like Association/Reassociation, Authentication/Deauthentication. In addition, superframe structure with Beacon frames and power-saving methods are not supported. We expanded the ns-802.11 model by adding EDCA and CFB. Also, we found a couple of errors in ns-2 and removed them.

EDCA uses four priority queues each with its own backoff instances. The priority parameters of each instance are CW_{min}, CW_{max}, $AIFS$, and $TXOP_{limit}$. More details about the implementation of the simulation model can be found in its open-source distribution and its description [213, 214].

11.2.2. Verification

To ensure the correctness of our simulation model we compared it with the work of Mangold et al. [140]. Mangold has implemented an EDCA simulation model in WARP and conducted

Table 11.1.: Mangold's [140] backoff-parameter set. We had to use the same parameters in our verification.

	High	Medium	Low
AIFS	2	4	7
CWmin	7	10	15
CWmax	7	31	255

Table 11.2.: *Our* and Mangold's results on the throughput of EDCA at different priorities and packet sizes.

Priority	Throughput [Mbps] with a packet size of		
of flow	80 bytes	200 bytes	2304 bytes
High	*3.52*/3.5	-	*19.98*/19.81
Medium	-	*6.32*/6.22	*19.32*/19.16
Low	-	*5.29*/5.21	*18.37*/18.22

some performance evaluations. If our simulation model achieves similar results to Mangold's work we assume it as to be verified and "correct".

Mangold used the IEEE 802.11a-PHY with a data rate of 24 Mbps; therefore we had to adopt the PHY parameters of our model, too. Also, the backoff parameters were chosen as given in Table 11.1.

Throughput: At first we considered and compared the maximal achievable throughput in ns-2.26 with the simulations in [140].

The scenario is a BSS consisting of a QoS AP (QAP) and only one wireless station. On this station one flow is sent to the QAP. Mangold performed separate simulations for each TC with increasing generation rates and larger packets (MSDUs). The throughput results of our and Mangold's simulation are listed in Table 11.2. We obtained similar results with only minor differences.

Number of stations: We took the following scenario (Figure 11.1) to verify that our model behave correctly for different numbers of competing stations: In the simulation scenario the number of the wireless stations was increased from 1 to 15. All stations are in range of one other and the stations were stationary. Each station sends three flows: a high-priority isochronous flow with 128 kbps and 80 byte MSDUs, a medium and a low-priority flow each 160 kbps and 200 byte MSDUs.

Mangold used Poisson regulated flows. We decided to use isochronous flows for all three

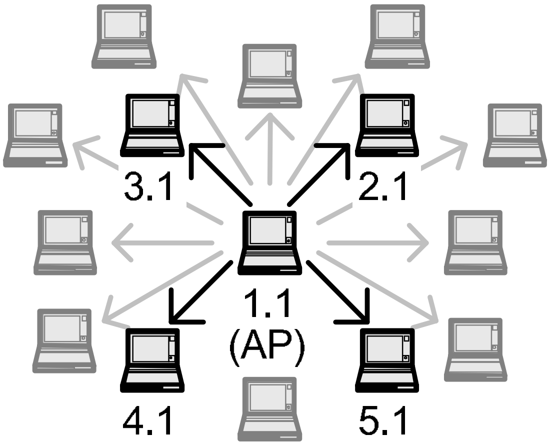


Figure 11.1.: An access point with a variable number of stations [140].

TCs due. The advantage of this simplification is an earlier termination of the simulations.

The Mangold's simulation results are shown in Figure 11.2 while our results are displayed in Figure 11.3. In our simulations, the low and the medium priority flows can carry its traffic only up to a number of 8 respective 11 stations. Mangold's simulation can carry 9 respective 12 stations. At high priority 13 stations can be served in our simulation most until the throughput per station decreases. In Mangold simulation the throughput scales linearly with the number of stations.

We explain this mismatch due to the different retransmission and packet drop behaviour of both simulation models. In our simulation model packets are dropped after the seventh collision. Retransmissions due to collisions often occur if many stations compete for the same medium. Also, the overall throughput reaches an upper limit if the number of stations increases. Even after extensive checking of our simulation code, we see no indications to doubt the results of our simulation model and its implementation.

11.3. Simulations

For the following simulations, we chose an 802.11b physical layer with a basic rate of 1 Mbps and a data rate of 11M bps. The 802.11e priority parameters were taken from the standard draft [96] and are given in Table 11.3. For comparison we list also the 802.11b backoff parameters. We considered a simulation scenario that consists of an access point and N wireless nodes. The access point is connected to a wired network that also contains N nodes (Figure 11.4). Each wireless node communicates with a corresponding wired node.

To simulate telephone calls, we transmitted bidirectional VoIP transmissions consisting of two continues flows of RTP packets containing 20 ms voice encoded with G.711. Voice flows in EDCA and EDCA/CFB modes are always handled with the highest priority. We did

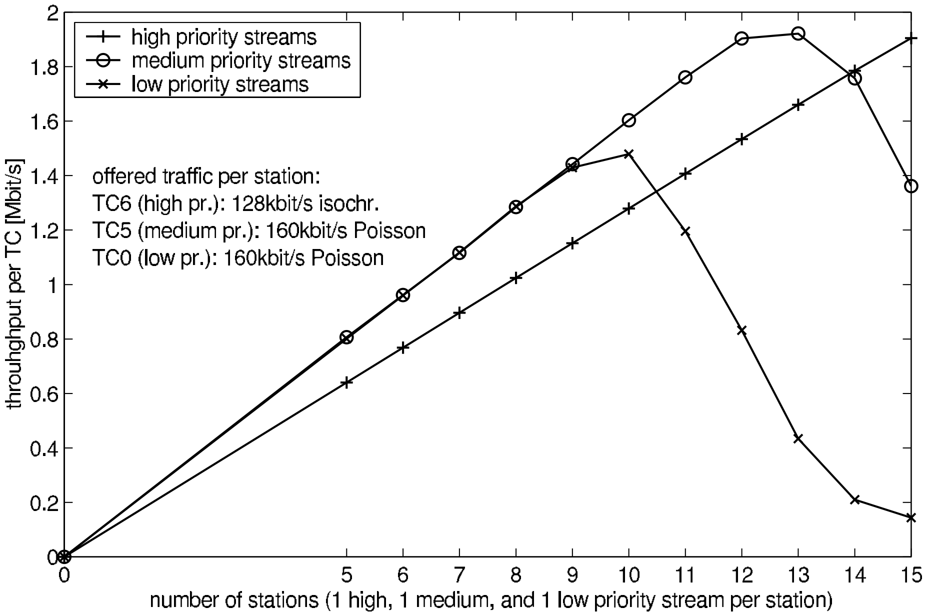


Figure 11.2.: Mangold's results for increasing number stations vs. throughput (source: [140]).

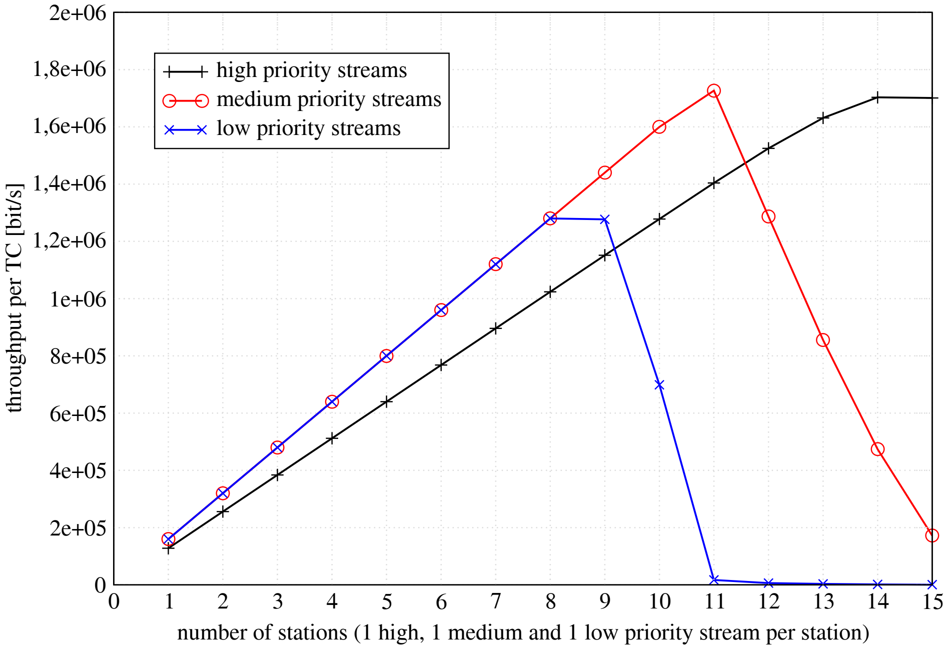


Figure 11.3.: Our results for an increasing number stations vs. throughput.

Table 11.3.: IEEE 802.11e priority parameter sets [96].

	High: TC[0]	**Low: TC[2]**	**802.11b**
AIFS	2	3	(2 = DIFS)
CWmin	7	31	31
CWmax	15	1023	1023
TXOPlimit	3 ms	0	-

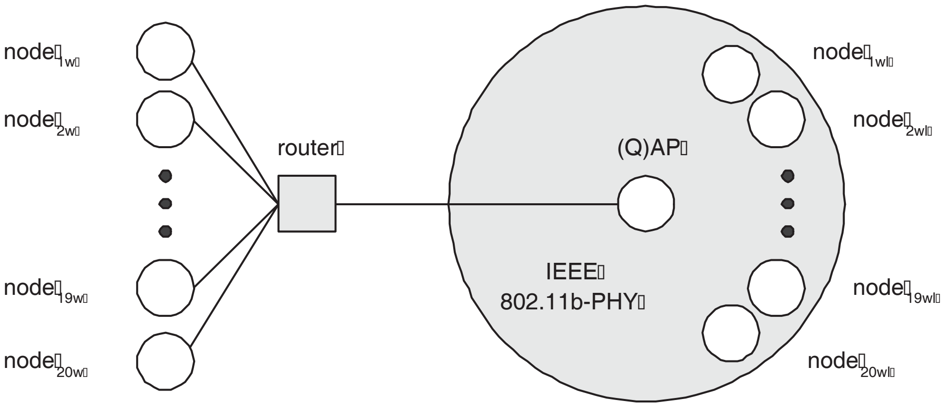


Figure 11.4.: Our simulation scenarios have one access point and different number of wireless and wireline nodes.

not yet used an error model in the simulation. To terminate our simulation we applied the tools Akaroa [158]. If the simulation results are within confidence intervals of 90 percent, the simulations are stopped automatically.

11.3.1. One VoIP call and downlink background traffic

We simulated the vulnerability of a voice stream, when competing TCP or UDP traffic are present. We used a simulation scenario with two wireless nodes and two wired nodes. The first wired/wireless node pair communicates via a bidirectional VoIP flow. Between the second pair of nodes TCP or UDP background traffic is transmitted.

In two separate simulations we ran a CBR and a FTP stream together with VoIP traffic cross the network. The CBR source as well as the FTP source produced 1500 byte large MSDUs at a rate of 11 Mbps. Both flows were transmitted at a low priority and were sent from the wired into the wireless network. To analyze and compare the QoS improvement of 802.11e EDCA for VoIP traffic, we simulated the first scenario with three different MAC modes: the 802.11 DCF, the 802.11e EDCA and the 802.11e EDCA+CFB.

In Figure 11.5 we display the throughput[1] of UDP and TCP flows (light bars) and the call quality of the downlink VoIP flows given in R factors (dark bars).

DCF mode: Using DCF mode and TCP traffic we gain a throughput of about 4 Mbps and bad call quality with an R factor of 38. Using DCF mode and UDP traffic a phone call was virtually not possible and the R factor was zero.

In both cases, both low priority and high priority are handled by the same transmission queue. Thus, both low and high priority traffic experienced the same loss rates. But these loss rates are top high for VoIP.

TCP adapts its sending rate to cope with congested link. However, this mechanism does not protect the VoIP traffic entirely against loss.

EDCA and CFB: Using EDCA and EDCA/CFB modes highly improve the conversational quality of the voice calls and decrease the goodput of TCP and UDP.

11.3.2. One VoIP call and uplink background traffic

We repeated the previous simulation but changed the direction of the background traffic. Now the TCP or UDP flows are transmitted from the wireless node to the wired one. Also, we measured just the quality of VoIP without background traffic.

Figure 11.6 displays the call quality in dependence of the direction of the background traffic. If no background traffic is present, the call quality is good. Also, if EDCA is used, the call

[1] We actually measured the *goodput*, which is the amount of data received.

Table 11.4.: Throughput of UDP or TCP downlink flow using the DCF or EDCF mode.

MAC	UDP	TCP
DCF	5.95 Mbps	4.47 Mbps
EDCA	5.896 Mbps	4.28 Mbps
Throughput costs of EDCA	0.85%	4.25%

quality is fine. Only if both VoIP and background traffic are in the download and the DCF mode is used, the quality is bad as stated previously.

Our results show that DCF are not capable to transmit VoIP if background traffic is transmitted in the same direction using the same transmission queue.

11.3.3. Impact of playout buffering delay

We simulated different playout scheduler schemes using the previous simulation scenarios. In Figure 11.7 the speech quality is shown depending on the playout buffering delay.

In the case of DCF mode and TCP downlink traffic and good quality is only reached if the end-to-end delay is larger than 0.22s. The end-to-end delay includes the transmission, the buffering delay, and an additional 150 ms system delay accounting for coding and packetisation. Using EDCA the end-to-end delay can be as low as 170 ms. This means, the wireless link add about 20 ms including dejittering to the transmission delay of a VoIP flow.

11.3.4. Maximal throughput

To determine the maximal possible background throughput, we transmitted best-effort traffic as in the previous simulations and without any voice transmission. The throughput of the UDP and TCP flows are shown in Table 11.4. EDCA is a slightly lower maximal throughput because of an increased contention period of the low priority TC.

11.3.5. Increasing number of calls

In this simulation we increased the number of the calls and the number of wireless stations from 1 to 20. We wanted to study the capacity of an IEEE 802.11b wireless cell, depending on the MAC mode. Figure 11.8 displays the results.

Downlink: In the downlink direction, EDCA and EDCA+CFB are able to deliver up to 11 voice transmissions; while DCF can serve only 10 VoIP flows. Due to the high-priority parameters (i.e. small CW_{min} and CW_{max}), the EDCA has less contention overhead than DCF.

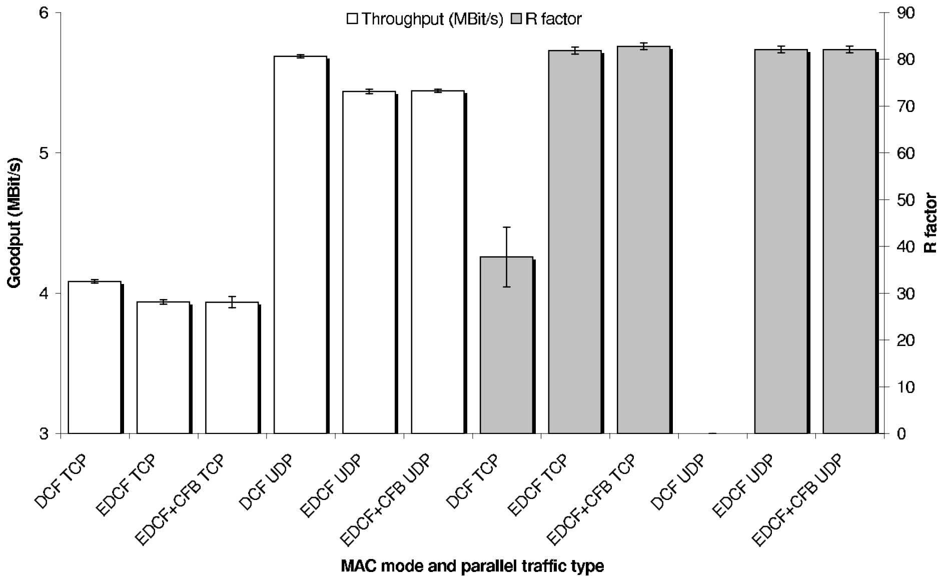


Figure 11.5.: The quality of one VoIP flow (R factor) and the throughput of one UDP or TCP traffic in the case of different MAC modes.

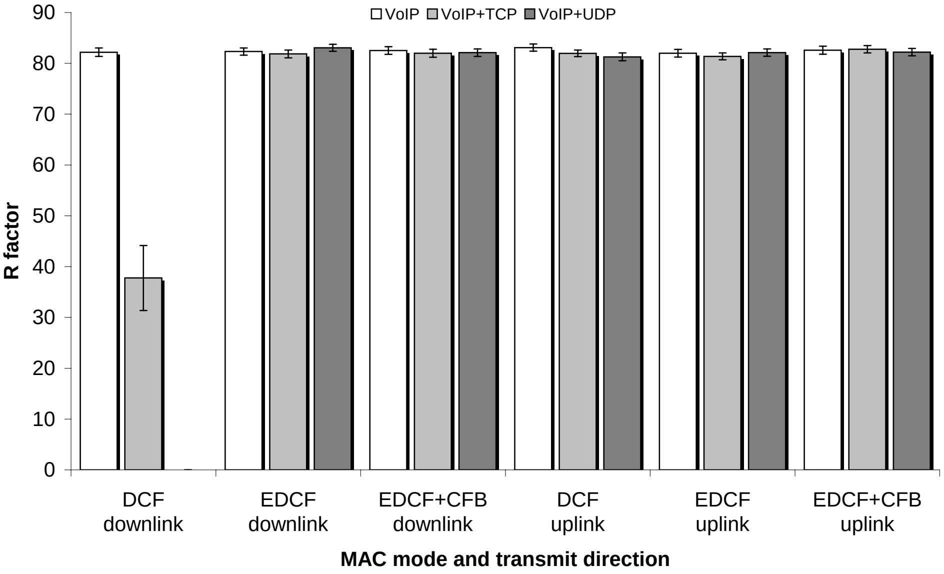


Figure 11.6.: The impact of the background traffic direction on the quality of a VoIP call for different MAC modes..

Uplink: EDCA can provide good conversational quality for up to 13 stations. EDCA+CFB delivers toll quality for 13 stations too, and decreasing quality for more than 13 nodes. DCF realizes a sufficient quality in the uplink direction for up to 18 stations due to its better collision resolution.

Comparing both directions, 802.11e EDCA and EDCA+CFB are sufficient for up to 11 voice calls, which we assume to be satisfactory for most wireless Internet telephony scenarios.

11.4. Comparison to related work

Our DCF results differ to the capacity studies in related work as given in Section 8.3. This can be explained because the simulation scenarios differ in many details. For example, different physical transmission, coding schemes and packetisation lengths have been chosen. Thus, a direct comparison is difficult.

Garg [60] considered a similar DCF scenario as we did. He also applied DCF, G.711 coding, IEEE 802.11b with 11 Mbps, and multiple parallel telephone calls. However, he applied a packetisation of 10 ms instead of 20 ms. Thus, the packet frequency is twice as high. The throughput of IEEE 802.11b is limited by the packet rate (refer to Section 9.4). Thus, one telephone call with 10 ms consumes about as much as two calls with 20 ms packetisation time. Consequently, the capacity that Garg has measured is about half (six calls) as the capacity that we gained (10 to 11 calls).

11.5. Conclusions

In this chapter, we presented our simulation model of the 802.11e EDCA for ns-2.26, which we verified.

We showed that VoIP calls are not always possible even using the DCF mode. Especially, if a VoIP call has to share the same queue with background traffic, the quality of the call can become insufficient. Using EDCA instead increases significantly the QoS and calls can be conducted at toll quality.

However, EDCA comes at cost. The maximal throughput of background traffic is reduced. For example, the throughput of a TCP flow might be lowered by 4% due to a longer contention period. UDP shows only a minor decrease in its goodput (less than 1%).

Finally, we determined the maximal possible number of simultaneous phone calls for the different MACs without any background traffic. With DCF we reached a maximal number of 10 calls while EDCA and EDCA+CFB are able to carry 11 (bidirectional) voice calls. At a higher number of calls, the prioritisation of EDCA mode does not work anymore because the lack of contention.

Altogether we can state that EDCA allows to conduct telephone calls in toll quality for in many different scenarios. At the moment, we do not see a requirement to implement HCCA as EDCA is sufficient.

11.6. Outlook

Many enhancements of VoIP over WLAN are possible:

Our simulation results showed that the capacity of downlink and uplink is different. We suggest that access points should use a different priority parameter set than the wireless nodes. Since the AP has to serve all nodes in the BSS, it should have more frequent access to the medium than an ordinary node.

Also, the concept of packet importance can be added to a VoIP over WLAN system in order to increase its capacity, speech quality, and energy efficiency. For example, highly important packets could be transmitted with a high priority, high amount of FEC or many retransmission. Such enhancements have been studies in our work [84, 85, 183] and in related literature (refer to Sections 8.4 and 8.5). Due time and space reasons we did not include our preliminary, but already published, results in this thesis.

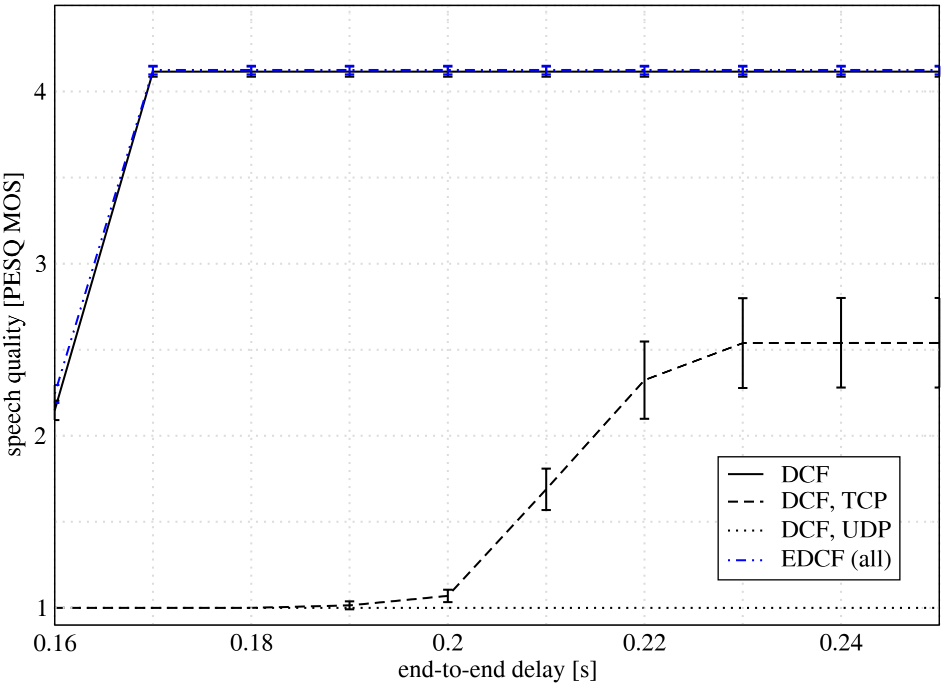
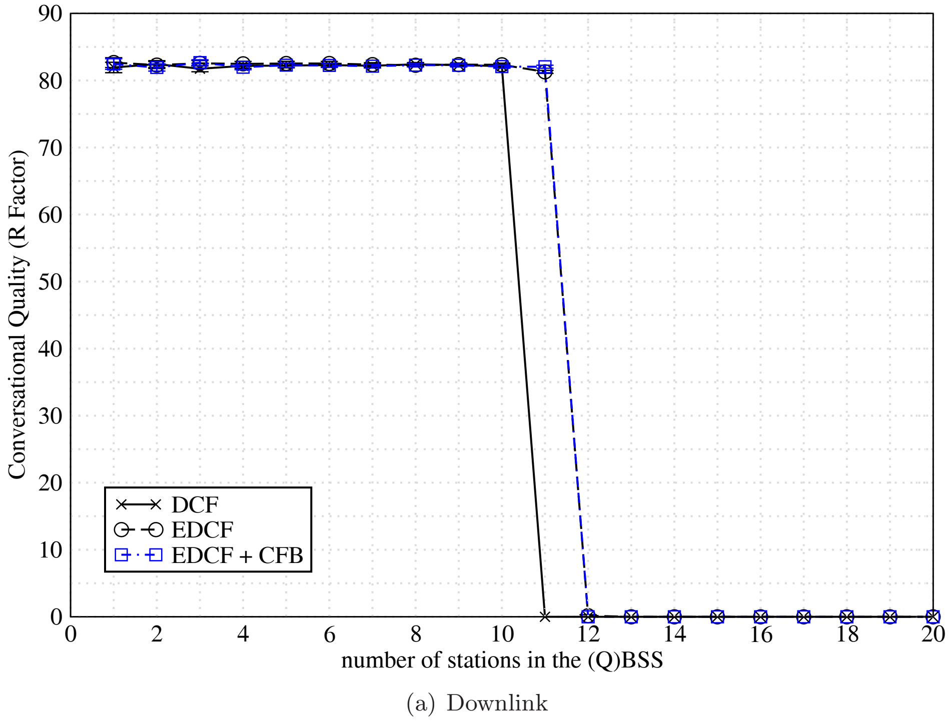
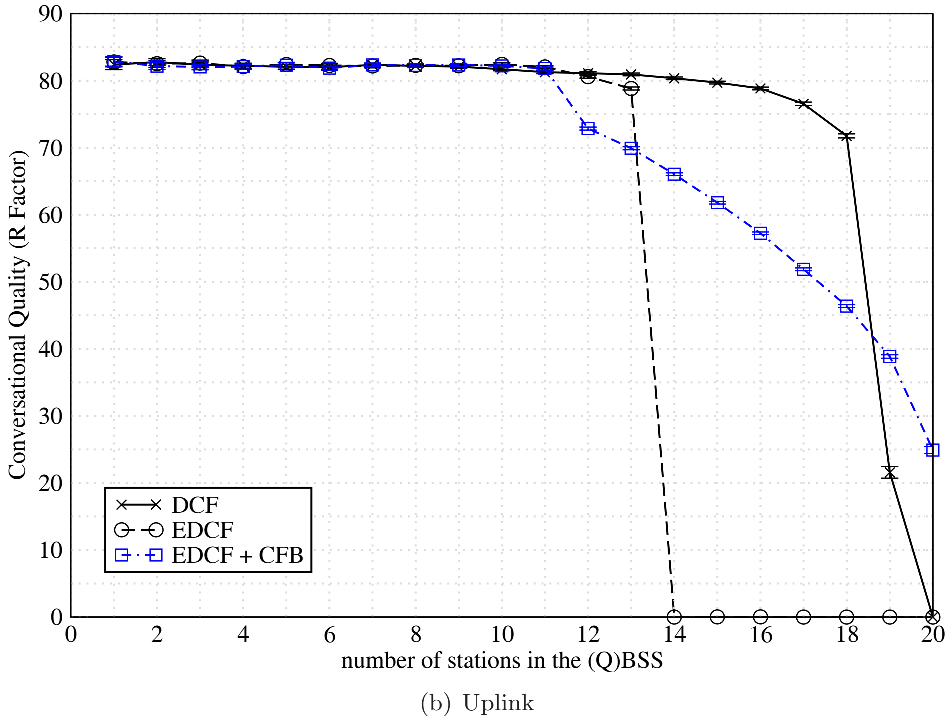


Figure 11.7.: Studying the impact of playout buffering delay on the downlink speech quality considering the presence of one background flow traffic and one VoIP flow.



(a) Downlink



(b) Uplink

Figure 11.8.: R factors for an increasing number of VoIP calls and wireless nodes.

12. Determining the Distance between WLAN nodes

As well as the transmission of VoIP packets over wireless links there is a second data-link task that is important for Internet telephony applications: Determining the position of the caller. Supporting location-based services (LBS) is beneficial to report the location during an emergency call, to control communication behaviours and to trigger communication actions. Let us give examples that have been mainly taken from the publication of Wu and Schulzrinne [220]:

- The user's position can be forwarded to track the location of the caller. Emergency calls have to be tracked by law, at least in the USA. Also, the location information helps to track persons, which are observed by legal enforcement entities.
- The location information helps to select the most suitable communication medium. For example, during driving the instant messaging can be disabled as the drivers need their hands on the wheel. Within a certain area such as a theatre play or a cinema the phone should automatically switch to the vibration alarm to avoid any potential disturbing, loud ringing. Actually, a context-aware communication service is required [65].
- Changes in the position of users can trigger communication actions. For example, if a person enters a room, a notification can be triggered or the light could be switched on.
- Also, communication can be directed to a geographical area: For example, the fire alarm can be forwarded to all persons in a building.
- Furthermore distance helps when deciding the time of handovers, finding the optimal routing path through an ad-hoc network, or enhancing the performance of the MAC protocol.

A location-based service requires a system, which reports the location of the user. Such systems can be implemented by many methods. For example, the position of a wireless node can be estimated using GPS. An alternative solution is to triangulate a mobile station in relation to nearby base stations. An essential part of any location determination algorithm is a method to determine the distance between two wireless nodes, such as a WLAN user and a base station.

This chapter is structured as follows: First we present our idea and give a short introduction. In Section 12.2 we explain our approach to enhance the measurement resolution. In Section 12.3 and 12.4 we describe our experimental measurement efforts. Finally, we briefly summarize the results and contributions of this chapter.

12.1. Introduction

In this chapter we focus on locating techniques that use intrinsic features of WIFI based wireless access. Usually, received signal strength indications are applied to identify the location of wireless nodes. We show that precise distance measurement based on round trip time measurements of WLAN packets is possible even with low-cost, commercial WLAN hardware. We developed an algorithm to determine the wave propagation time indirectly and to improve the accuracy and resolution of the time measurements. We validated our approach with two independent experimental measurement efforts and with an analytical proof.

We utilize the following feature of IEEE 802.11: Each unicast data packet is immediately acknowledged by its receiver (Figure 12.1). We took the time between starting the transmission of a data packet and receiving the corresponding immediate acknowledgement. We will refer to this as the remote delay (d_{remote}). We also measured the duration of receiving one data packet and sending out the immediate acknowledgement. We will call this duration local delay (d_{local}). The overall propagation time is then estimated by subtracting the local from the remote delay.

$$c = \frac{2 \cdot distance}{d_{remote} - d_{local}} \text{where } c \approx 3 \cdot 10^8 \frac{\mathrm{m}}{\mathrm{s}} \text{ being the speed of light.} \tag{12.1}$$

In order to overcome the problem of interrupt latencies and hence inaccuracies when measuring the duration of packet transmission in the operating system, we measured the time on the hardware layer, the WLAN card. Most WLAN solutions record timestamps at a resolution of $1\,\mu s$. However, a packet travels a distance of 300 m in 1 μs, which usually exceeds the range of WLAN transmission. We increase this resolution by using multiple delay observations and applying statistical methods to enhance the accuracy.

We take advantage of the fact that both the local and remote clocks drift and interfere. The interference is caused by the data-acknowledgement sequence. As a result the observations contain a beat frequency that is equal to the frequency offset of the local and remote clock crystals. The beat frequency introduces measurement noise, which we utilize to identify a weak signal below the timers' quantization resolution. The weak signal is the propagation delay.

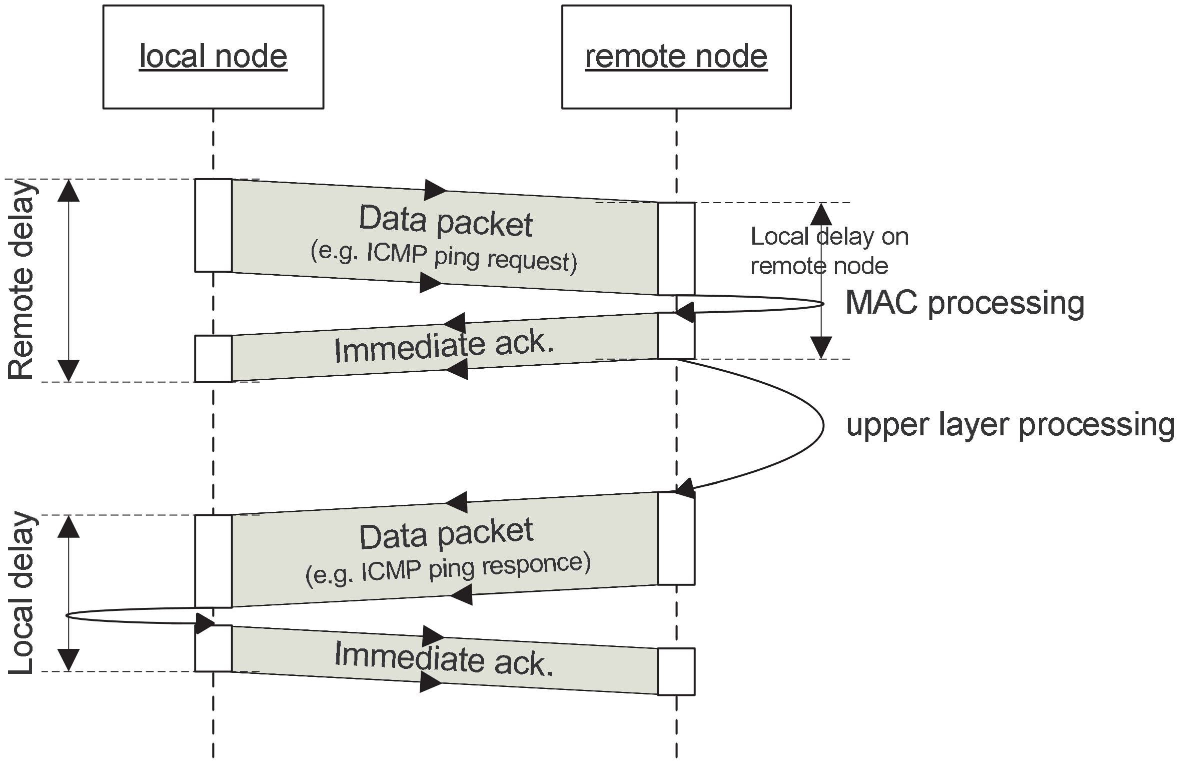


Figure 12.1.: Transmission of an ICMP ping sequence.

12.2. Approach

Inspired by the approach presented in [130], which used ping packets to measure the speed of light, we also use the mean round trip time delay of packets to determine the distance as given in Equation 12.1. In order to keep the time measurements as unbiased as possible, we tried to remove any disruption caused by operating system activities. To do so, we took the following actions:

Firstly, we utilized the IEEE 802.11 data/acknowledgement sequence instead of the ICMP-Ping request/response packet sequence. As the ping response is generated by the operating system the time it takes is subject to a highly variable delay. In contrast, the immediate acknowledgements are handled by the hardware of the WLAN radio and hence predictable: On standardized IEEE 802.11 the MAC processing time (SIFS interval) is 10 μs with a tolerance up ±25 ppm. Acknowledgements are only valid if they are received after the SIFS interval with a tolerance of ±2 μs. Thus, if the WLAN card is implemented according to the standard, the transmission delays are deterministic. We can also assume that the MAC processing times on both nodes are identical [1].

Secondly, we did not measure the timestamps of packet arrivals and transmissions at the

[1] In practice, the MAC processing time also depends on the chip set hardware and firmware of the actual WLAN cards in use. To account for this a model-specific absolute delay offset needs to be considered.

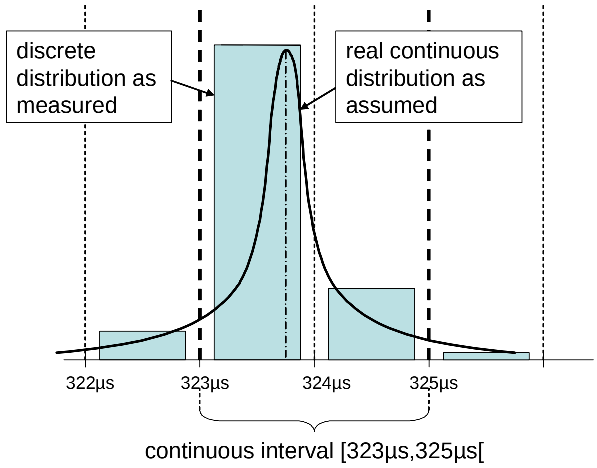


Figure 12.2.: Discrete distribution of noisy delay measurements.

operating system layer, but on the WLAN card hardware layer. These timestamps are interrupt latencies. In Figure 10.5 we showed that measuring the time of a packet's arrival in the operating system's kernel (e.g. during an interrupt) lead to imprecise results. Indeed, in our experiments OS time measurements did not work for distance measurements. The resolution of hardware timestamps, which are implemented in most current WLAN products, is 1 μs corresponding to 300 m. In terms of the achievable accuracy this discrete time resolution is not precise enough. The resolution increases by averaging several observations. In the following we consider three phenomena that help to achieve a higher resolution.

12.2.1. Gaussian noise

The presence of measurement noise is assumed. Thus, the delay values are not limited to only one value.[2] If one assumes a Gaussian noise distribution with a suitable strength, we can simply take the sample mean to enhance the resolution. Also, it can be expected that different discrete delay values are randomly distributed.

But which effects introduce noise? The measurement noise can be caused by thermal noise present in the received radio signal. Thus, synchronization of the modulation symbols might vary. In a multi-path environment, the dominant propagation path varies leading to changes in the propagation delay. Also, the crystal clocks of the WLAN equipment are subject to a constant clock drift and variable clock noise.

[2] In Figure 12.2 not only 323 μs can be observed but also other values.

12.2.2. Stochastic resonance

Instead of the explanation above the authors of [130] suggested another statistic effect called stochastic resonance. The concept of stochastic resonance was originally introduced as an explanation for the periodically recurrent ice ages. In the last two decades, it has been applied to explain many physical phenomena [58, 148]. In the realm of signal detecting [73] stochastic resonance allows for detecting signals below the resolution of the measuring units as the signal becomes detectable with the help of noise. Noise is added to the signal so that the signal sometimes exceeds the threshold given by the resolution of the detecting device.

For example, in a bi-stable system a state change occurs only if the noise and weak signal are together higher than a barrier between both states. The length of the period that the system stays in one state is random. If one measures discrete values the probability is high that one value remains constant for the next observation. This effect results in blocks of the same values and these blocks have random lengths.

12.2.3. Beat frequencies

In our experiments (see Figure 12.7) it was observed that the 323 and 324 values occur in blocks of regular patterns. This observation cannot be explained with stochastic resonance.

We explain this observation with another effect: Both WLAN cards are driven by built-in crystal oscillators that nearly have the same frequency. Due to tolerances, there is a slight drift between both clocks which causes varying *quantization errors*.

Let us consider the impact of a discrete time resolution on the measurement error. Firstly, we construct a model of the experimental setup. Instead of using packets, we assume that a delta pulse is sent from the local node to the remote node. After the delta pulse's arrival another delta pulse is sent back to the local node representing an acknowledgement. The local node can process the impulses only in discrete time steps $t_{local} \in \mathbf{N}$ described by natural numbers. The same is also valid for the remote node. It only reacts in discrete time steps, which are $t_{remote} = \delta + n$ where $n \in \mathbf{N}$ and phase offset is $\delta \in [0; 1[$. We assume that the clocks operate at the same frequency but have a phase offset. Moreover, a further assumption is that phase offset changes over time but not over the duration of a round trip. The transmission of a delta impulse from one node to the other last $d_{prop} \in \mathbf{R}^+$, which is equal to the propagation time.

Let us assume that a delta impulse is sent from the local node at the time t^{out}_{local}. It arrives the remote node after a period of d_{prop}. Due to the discrete MAC processing, the delta impulse is only identified at the next remote clock impulse, which is:

$$t^{in}_{remote} = \left\lceil \left(t^{out}_{local} + d_{prop} \right) - \delta \right\rceil + \delta \tag{12.2}$$

Assuming a MAC processing duration equal to zero and $t^{out}_{remote} = t^{in}_{remote}$, the remote node

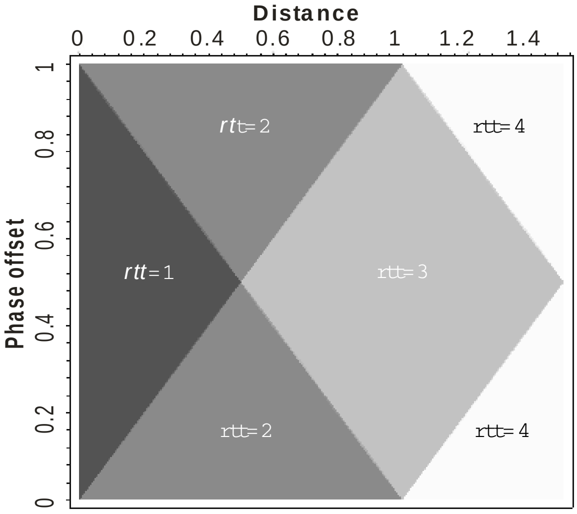


Figure 12.3.: Round trip time versus distance and phase offsets.

immediately sends back a delta impulse representing the acknowledgement. It arrives at the local node after a period of d_{prop}, but is again only recognized at the next local clock, which is

$$t_{local}^{in} = \left\lceil t_{remote}^{out} + d_{prop} \right\rceil \tag{12.3}$$

Then, the observed round trip time rtt is calculated with Equation 12.4 that is displayed in Figure 12.3.

$$\begin{aligned} rtt &= t_{local}^{in} - t_{local}^{out} = \lceil \lceil t_{local}^{out} + d_{prop} - \delta \rceil + \delta + d_{prop} \rceil - t_{local}^{out} \\ &= \lceil d_{prop} + \delta \rceil + \lceil d_{prop} - \delta \rceil \end{aligned} \tag{12.4}$$

Next, we assume that the phase changes from one measurement to the next. The change is constant and is repeated after each phase period starting at zero again. In the following, we only consider one phase period and assume that round trip times are measured at all times. Thus, the number of observations is infinite. The mean rtt over all phase offsets is calculated as follows.

$$\overline{rtt} = \int_0^1 rtt \, d\delta = \int_0^1 \lceil d_{prop} + \delta \rceil + \lceil d_{prop} - \delta \rceil \; d\delta = \; 2 \cdot d_{prop} + 1 \tag{12.5}$$

The variance of the quantization error is calculated as follows and is simplified to a cubic function of the fractional part of the round trip distance. Both the mean and variance are displayed in Figure 12.4.

$$\sigma^2 = \int_0^1 \left(\overline{rtt} - rtt\right)^2 \, d\delta = \{2d_{prop}\} - \{2d_{prop}\}^2 = \tfrac{1}{4} - \left(\{2d_{prop}\} - \tfrac{1}{2}\right)^2 \tag{12.6}$$

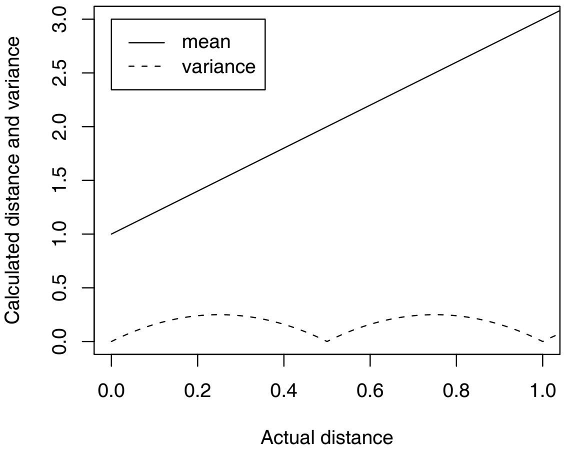


Figure 12.4.: Theoretical mean distance and variance of distance.

The *rtt* function produces a pattern that is repeated every phase period. This reoccurrence introduces a frequency component present in the observations. If two clocks interfere, their phases are equal every beat period, which is the reciprocal of the beat frequency. The beat frequency is the difference of the frequencies of the two interfering waves (12.7). Thus, the impact of quantization errors causes a similar effect as the two interfering waves – namely a beat frequency.

$$f_{beat} = |f_1 - f_2| \tag{12.7}$$

12.2.4. Limits

The accuracy of location and distance sensing algorithms has fundamental limits [29, 37, 52, 56, 153]. For example, the analytic calculations above do not take into account clock drift during one RTT observation. Assuming a frequency stability of ±25 ppm and a length of a transmission sequence of 60 μs and 320 μs, the maximal error can be up to 0.9 m and 4.8 m respectively.

Furthermore, one should note that only in vacuum light travels at the speed of light c. In materials the propagation speed depends on the square root of the dielectric constant ε. For example, dry ferro-concrete has an ε of about 9 and electromagnetic waves traverse through ferro-concrete 3 times slower than in vacuum. Most other materials used in buildings have lower dielectric constants [224].

Another source of possible errors is due to non-line-of-sight conditions. This results in an overestimation of the distance between the two nodes [45]. Multi-path propagation might introduce measurement errors because the dominant path can vary depending on the current transmission conditions. Multi-path propagation is only present if reflections are given. Re-

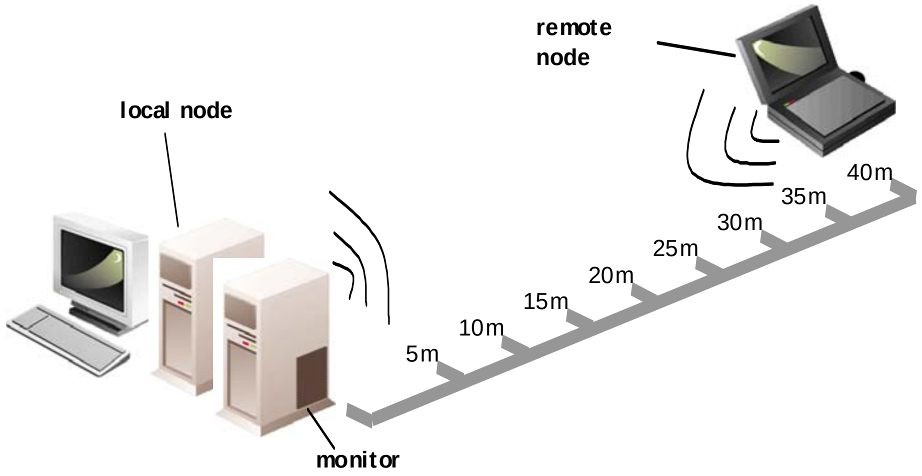


Figure 12.5.: First measurements: schematic experimental setup.

flections can have a large impact on the signal strength but only lower one on the propagation delay. Thus, in the presence of multi-path propagation or reflections, we assume time delay measurements as being more precise than those based on the RSSI.

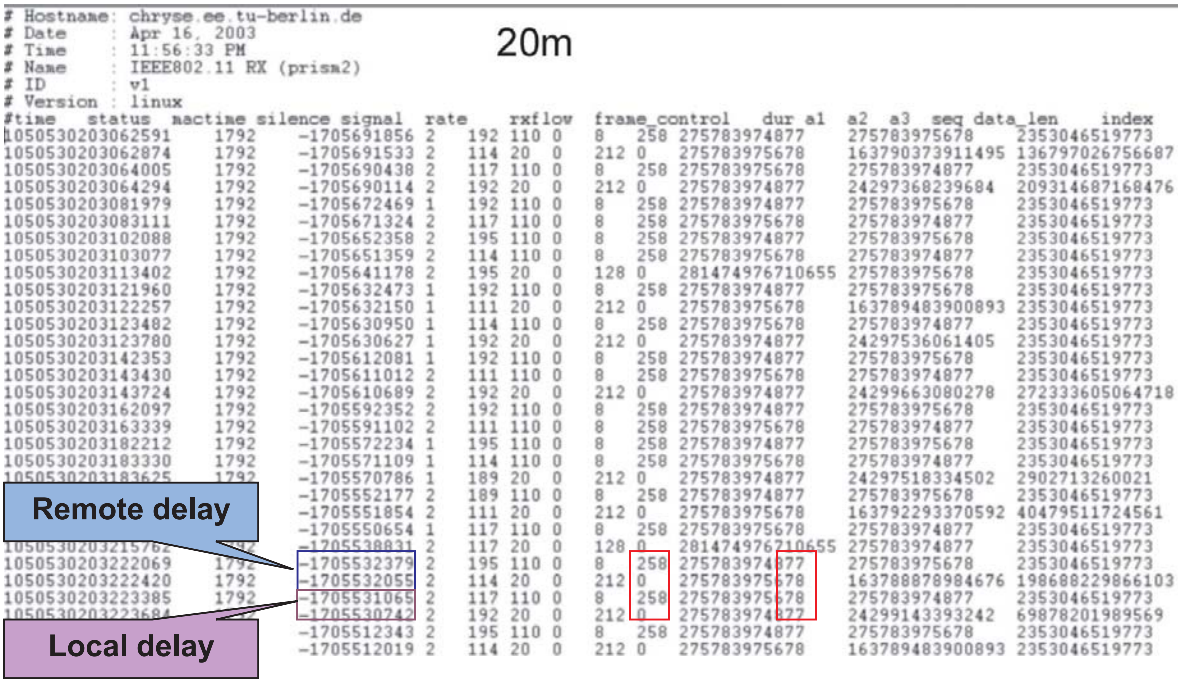
12.2.5. Verification

In order to verify these hypotheses and identify the real measurement resolution, we conducted several experiments. The first measurement effort was conducted to study the impact of slow-user motion on packet loss and delay as described in Chapter 10 and [86]. At the same time, we also measured the impact of distance on the round trip times. One year later, we embarked on a second measurement effort. We changed the radio modem technology, the location, the analysis software, and the staff. The consistency of both results proves the reliability and correctness of our approach.

12.3. First measurement effort

Experimental setup: The measurement was done in a gymnasium (Figure 10.8 and 12.5). The data communication takes place between the local and the remote node. ICMP ping packets were transmitted every 20 ms. The measurements of RTT were conducted for several distances (5, 10, 15, 20, 25, 30, 35, and 40 m). At each distance, we measured for about 15 minutes. One should note that the wireless LAN cards were close to the ground. Also, the directions of the antennas were selected at random and were not recorded.

Equipment: All PCs were running Suse 6.4 Linux system with a 2.4.17 kernel. D-Link cards featuring an Intersil's (now Conexant) Prism2 chipset were used as the wireless interface.



```
# Hostname: chryse.ee.tu-berlin.de
# Date    : Apr 16, 2003
# Time    : 11:56:33 PM
# Name    : IEEE802.11 RX (prism2)
# ID      : v1
# Version : linux
#time   status  mactime silence signal  rate    rxflow  frame_control   dur a1   a2  a3  seq data_len    index
1050530203062591      1792      -1705691856 2    192 110 0    8    258 275783974877     275783975678    2353046519773
1050530203062874      1792      -1705691533 2    114 20  0    212 0    275783975678     163790373911495 136797026756687
1050530203064005      1792      -1705690438 2    117 110 0    8    258 275783975678     275783974877    2353046519773
1050530203064294      1792      -1705690114 2    192 20  0    212 0    275783974877     24297368239684  209314687168476
1050530203081979      1792      -1705672469 1    192 110 0    8    258 275783974877     275783975678    2353046519773
1050530203083111      1792      -1705671324 2    117 110 0    8    258 275783975678     275783974877    2353046519773
1050530203102088      1792      -1705652358 2    195 110 0    8    258 275783974877     275783975678    2353046519773
1050530203103077      1792      -1705651359 2    114 110 0    8    258 275783975678     275783974877    2353046519773
1050530203113402      1792      -1705641178 2    195 20  0    128 0    281474976710655 275783975678    2353046519773
1050530203121960      1792      -1705632473 1    192 110 0    8    258 275783974877     275783975678    2353046519773
1050530203122257      1792      -1705632150 1    111 20  0    212 0    275783975678     163789483900893 2353046519773
1050530203123482      1792      -1705630950 1    114 110 0    8    258 275783975678     275783974877    2353046519773
1050530203123780      1792      -1705630627 1    192 20  0    212 0    275783974877     24297536061405  2353046519773
1050530203142353      1792      -1705612081 1    192 110 0    8    258 275783974877     275783975678    2353046519773
1050530203143430      1792      -1705611012 2    111 110 0    8    258 275783975678     275783974877    2353046519773
1050530203143724      1792      -1705610689 2    192 20  0    212 0    275783974877     24299663080278  272333605064718
1050530203162097      1792      -1705592352 2    192 110 0    8    258 275783974877     275783975678    2353046519773
1050530203163339      1792      -1705591102 2    111 110 0    8    258 275783975678     275783974877    2353046519773
1050530203182212      1792      -1705572234 1    195 110 0    8    258 275783974877     275783975678    2353046519773
1050530203183330      1792      -1705571109 1    114 110 0    8    258 275783975678     275783974877    2353046519773
1050530203183625      1792      -1705570786 1    189 20  0    212 0    275783974877     24297518334502  2902713260021
                                -1705552177 2    189 110 0    8    258 275783974877     275783975678    2353046519773
                                -1705551854 2    111 20  0    212 0    275783975678     163792293370592 40479511724561
                                -1705550654 1    117 110 0    8    258 275783975678     275783974877    2353046519773
1050530203215762                -1705538831 2    117 20  0    128 0    281474976710655 275783974877    2353046519773
1050530203222069      1792      -1705532379 2    195 110 0    8    258 275783974877     275783975678    2353046519773
1050530203222420      1792      -1705532055 2    114 20  0    212 0    275783975678     163788878984676 198688229866103
1050530203223385      1792      -1705531065 2    117 110 0    8    258 275783975678     275783974877    2353046519773
1050530203223684                -1705530742 2    192 20  0    212 0    275783974877     24299143393242  69878201989569
                                -1705512343 2    195 110 0    8    258 275783974877     275783975678    2353046519773
                                -1705512019 2    114 20  0    212 0    275783975678     163789483900893 2353046519773
```

Figure 12.6.: Snuffle trace file showing recorded data traffic (20 m measurement).

Packets were directly sniffed at the MAC layer by the measurement tool ‘Snuffle’.

Configuration: WLAN networking technologies based on the IEEE 802.11 standards transmit data packets via the air. Each data packet is immediately acknowledged if it is received without errors. To avoid potential packet delay effects, in this experiments the maximal number of retransmissions was set to zero.

Time measurements: The Prism2 cards only implement the recording timestamps of incoming packets. But we needed both the sending and receiving timestamps. Therefore, we decided to use a third PC to monitor the packets that the local node sends and receives. The monitor PC was placed close to the sender to avoid any additional propagation delays that could invalidate the measurements.

Data collection & processing: Snuffle provides the packet traces of all 802.11 packets received at the monitoring node. We filtered-out only the successful ping sequences that consist of an ICMP request, an acknowledgement, an ICMP response and finally an acknowledgement (Figure 12.6). Other packets like erroneous transmissions, beacons, ARQ messages etc. are dropped. Due to hardware limitations of the WLAN card only a fraction of the observations were recorded (Table 12.1).

Only the delays in the interval [323 μs, 324 μs] are considered in further calculations (Figure 12.7). Only a very few delay measurements were observed with values of 322 and 325 μs.

Table 12.1.: Numbers of missing, invalid and valid observations.

distance	trace file entries	good entries	corrupted entries	share of corrupted entries
5 m	14371	12722	1649	11.5%
10 m	21256	18450	2806	13.2%
15 m	89877	77440	12437	3.8%
20 m	10316	9344	972	9.4%
25 m	9864	8822	1042	10.6%
30 m	20095	18124	1971	9.8%
35 m	40776	35682	5094	12.5%
40 m	32750	29216	3534	10.8%

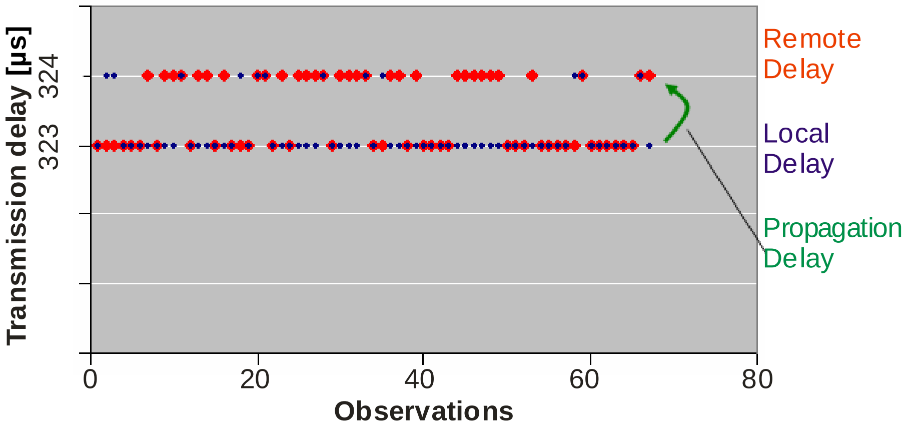


Figure 12.7.: Remote and local delay observations over time.

These and all other delays were considered as measurement errors. Given these packet sequences, the mean and variance of the remote delay and local delay were calculated. To check for stationary process properties, the autocorrelation function was calculated. The screening of data entries and the subsequent calculations were executed by a self created C program [66].

Results: The distance was directly derived from the measured propagation delay using Equation 12.1. Assuming a Gaussian error distribution, we also plotted the confidence intervals in Figure 12.8 and Table 12.2. The calculated distances were always higher than the real distances. Also, in some measurements (e.g. 35 m) the air propagation time was significantly higher. Due to the experimental setup, we could not ensure that the direct line-of-sight path was taken. The remote node was placed directly on the ground. Thus, the *Fresnel zone* was violated and the direct transmission path was hampered. In radio communications, a Fresnel zone is a concentric ellipsoid, covering the radiation path. Fresnel zones result from diffraction by the circular aperture.

In Figure 12.9 the signal strength is displayed as a function of the distance. Theoretically,

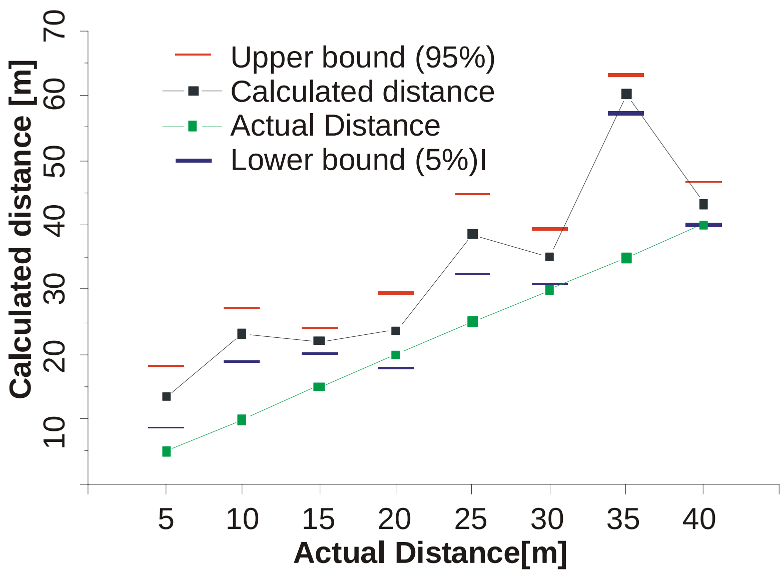


Figure 12.8.: Distance as calculated from RTT versus actual distance between both nodes. 95% confidence levels are given.

Table 12.2.: Delays, calculated distance and standard deviation versus real distance.

actual distance [m]	remote delay [μs]	local delay [μs]	one-way delay [ns]	calculated distance [m]	standard deviation [m]
5	323.297	323.207	45.0	13.44	8.4400
10	323.359	323.205	77.0	23.12	13.1125
15	323.377	323.230	73.5	22.07	7.0690
20	323.396	323.238	79.0	23.74	3.7395
25	323.465	323.208	128.5	38.62	13.6165
30	323.450	323.216	117.0	35.11	5.1105
35	323.567	323.166	200.5	60.21	25.2050
40	323.481	323.192	144.5	43.31	3.3090

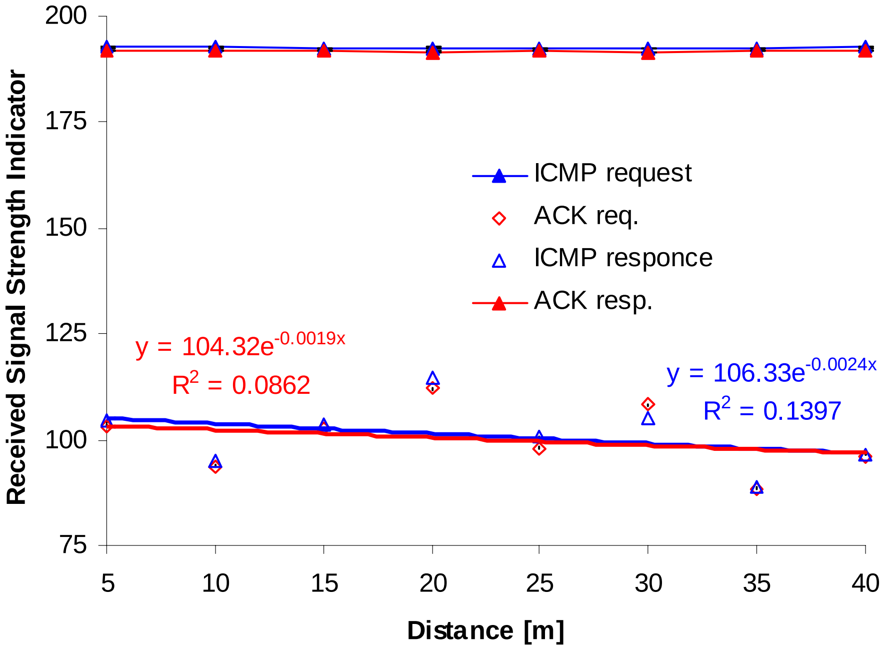
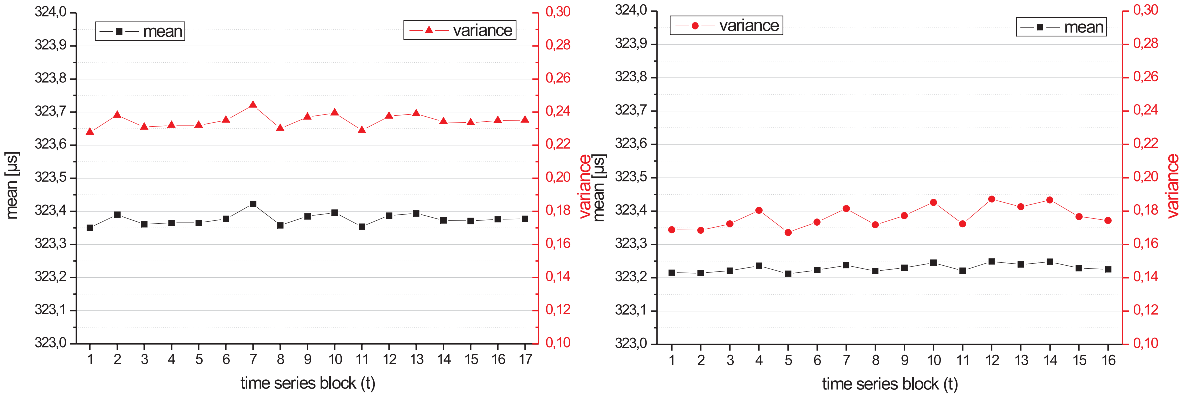


Figure 12.9.: Received signal strength indication versus distance. Confidence intervals are too small to be shown.

the signal strength should decrease with distance. In this measurement effort other factors, such as reflection, seem to be dominant. If one compares Figures 12.8 and 12.9, it is clear that time measurements reflect the distance more precisely than the RSSI but they have a higher variance and a larger confidence interval.

Analysis: In [66] we show that the measurements follow a weak stationary process, with a constant mean, variance and covariance (for a constant lag) (Figure 12.10). Thus, further statistical methods are applicable.

Confidence intervals are only meaningful if the observations are independent. This assumption can be verified by the autocorrelation function. The time-lag dependent autocorrelation coefficients are presented as a graph in Figure 12.11. The 15 m and 40 m results are shown as examples. The graphs at other distances look similar. The autocorrelation for the local delay is low. It is smaller than ρ=0.05. Thus, the local delay measurements can be seen as independent. The autocorrelation of remote delay values has the form of a decaying cosine wave. This kind of autocorrelation curve is found if the observations have a constant frequency component. Indeed, this pattern manifests itself in the delay traces. The values of 323 and 324 occur block-wise in bursts. We also calculated an FFT over the packet delays.



(a) Remote delay

(b) Local delay

Figure 12.10.: Mean and variance over time at 15 m.
The distance measurements are a stationary process.

Assuming that each observation follows the previous after 20 ms, we identified a dominant frequency at about 3.5 Hz independent of the distance (Figure 12.12). However, the lower the packet error rate, the stronger this effect is.

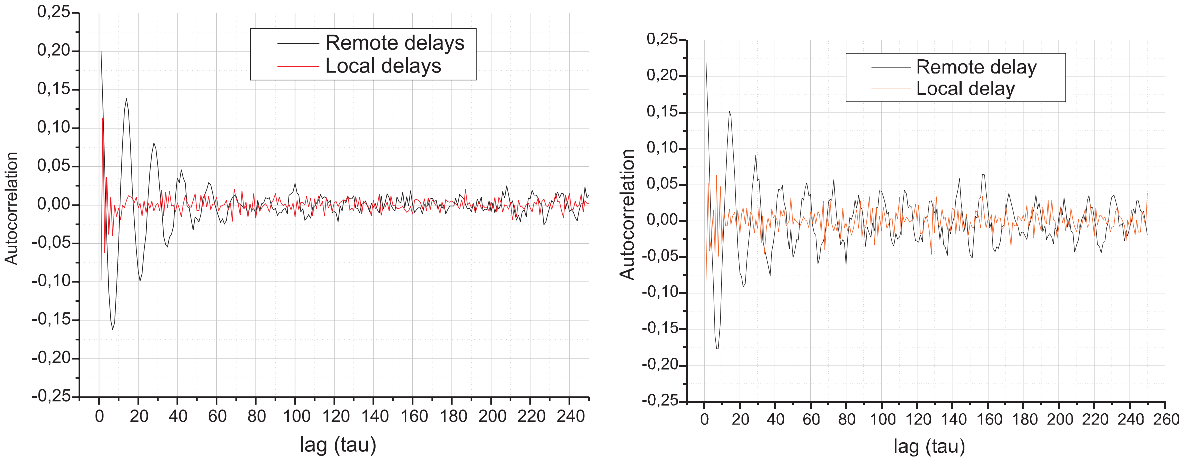
We explain the effect displayed in Figure 12.11 with the interference of both remote and local crystal clocks. Given this explanation of quantization errors, we can calculate the clock drift between both signals. Assuming a clocking of the MAC protocol at 1 MHz, the drift between both clocks is approximately $drift = \frac{f_{beat}}{f_1} = \frac{3.5Hz}{1MHz} = 3.5ppm$. Usually, the tolerance of consumer grade quartz clocks is up to 25 ppm. Thus, we consider this explanation to be plausible.

Interestingly, the MAC processing is conducted in steps of 1 μs. Thus, the MAC processing time is not precisely the SIFS interval but is rounded up to the next 1 μs. However, the error is small so that receivers tolerate it.

In our quantization error analysis we calculated the variance which is up to $\frac{1}{4}$. A distance of one and a time unit of one in the analysis refer to 300 m or 1 μs in the experiments. Then, the standard deviation should be 18.75 m or 62.5 ns at most. The measured standard deviation is between 3.3 and 25 m. Thus, the quantization error is not the only dominant effect, and others factors are important, too.

12.4. Second measurement effort

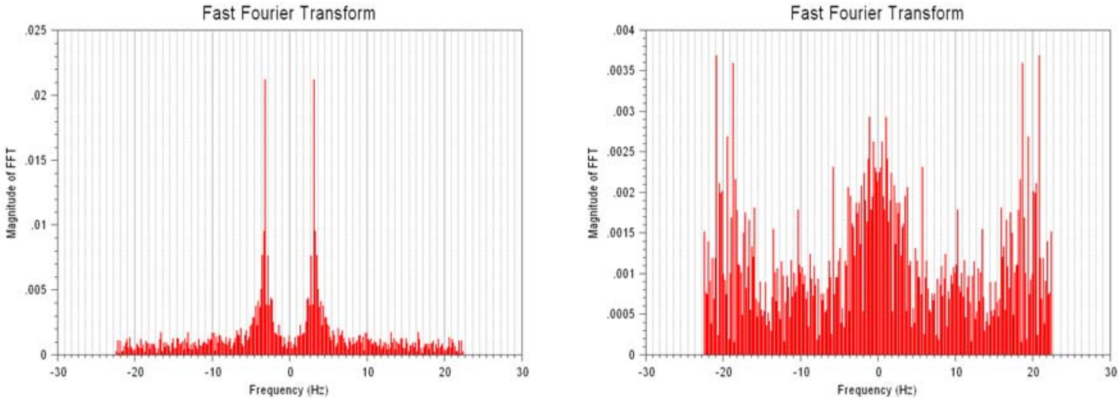
Experimental setup: The measurements were conducted outside in the countryside where one could expect the channel to be free of disturbing noise coming from other radiating devices. The measurements were extended to the maximal transmission range of 100 m. The sender



(a) at 15m

(b) at 40m

Figure 12.11.: Autocorrelation (=cross correlation of itself) is oscillating for remote delays – indicating a fundamental frequency component in observations.



(a) Remote delays

(b) Local delays

Figure 12.12.: The Fourier transformation of the observations shows a dominant frequency at 3.5 Hz, which is only present in the remote delays.
Taken from the 40m measurements.

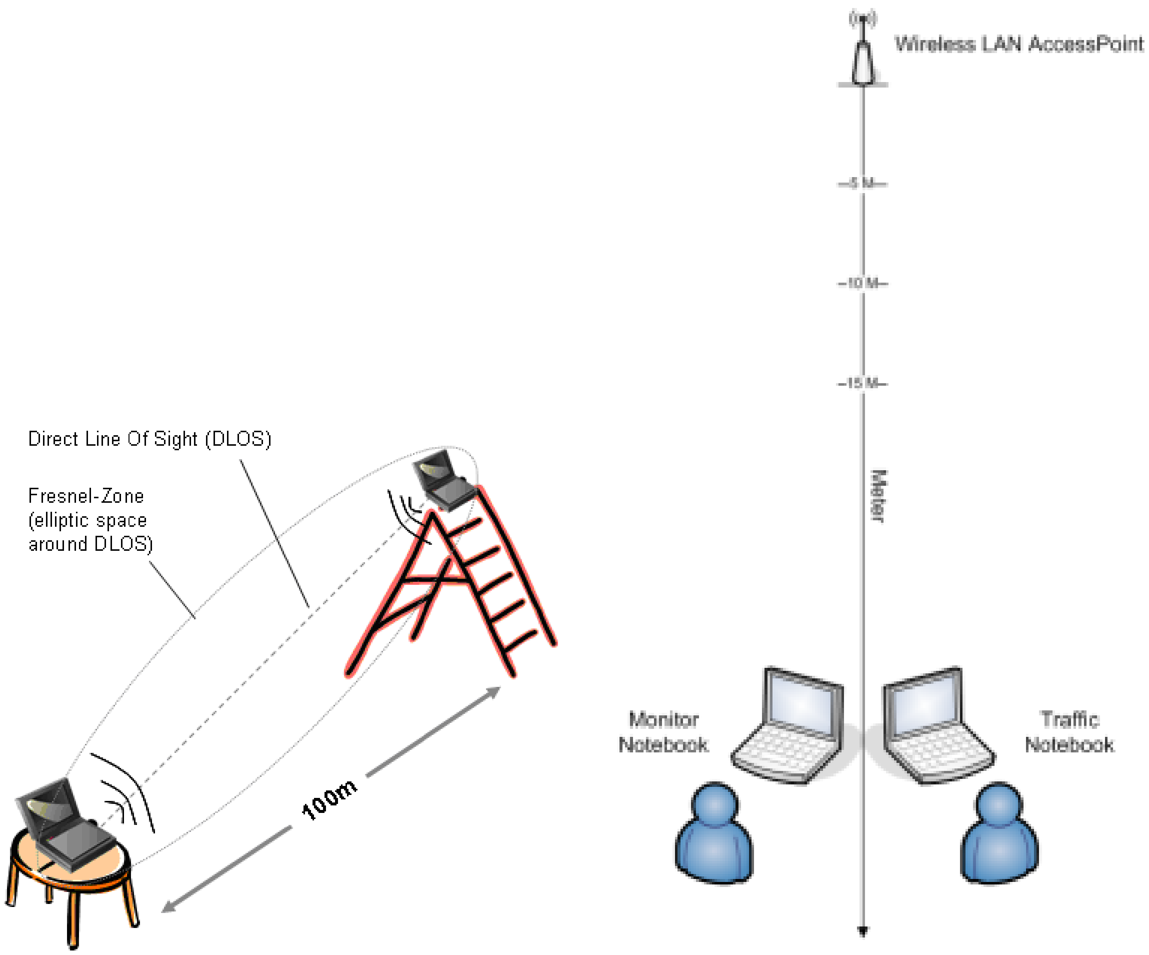


Figure 12.13.: Setup of the second effort.

was placed on a table, whereas the receiver was installed at the top of a 1.5 m ladder. This was to guarantee that a large percent of the Fresnel-zone is free of any obstacles disturbing the transmissions. This time, the antennas were directed at each other. The schematic setup is displayed in Figure 12.13. A notebook acting as a local node was sending out ICMP request packets. An access point was used as a remote node. Again, a monitoring PC close to the local node was required. Ping packets were sent every 10ms until the monitor received up to 20.000 packets.

Equipment: We used an access point (Netgear FWAG114) supporting 802.11b/g and as a remote node. The PCs were running under Linux, Suse 9.1, with a 2.6 kernel. We used two different WLAN cards containing chipsets from Atheros and Conexant implementing IEEE 802.11 a,b and g. The Atheros cards (branded Netgear WAG-511, contained an AR5212 chip) are supported by the Madwifi device driver. We used the software version downloaded

Table 12.3.: Configuration: Modulation speed of MAC packets depending on direction and type.

Mode	Chipset	Monitor	Request	Ack.	Response	Ack.
		CPU	l→r	r→l	r→l	l→r
amilo_ath_36m	Atheros	850 MHz	54	24	36	24
amilo_ath_54m	Atheros	850 MHz	54	24	54	24
asus_ath 36m	Atheros	1.5 GHz	54	24	36	24
asus_ath_54m	Atheros	1.5 GHz	54	24	54	24
asus_prism_36-54m	PrismGT	1.5 GHz	54	24	36	24
asus_prism_36m	PrismGT	1.5 GHz	36	24	36	36
asus_prism_54m	PrismGT	1.5 GHz	54	54	54	54

from the CVS server on the August 30^{th}, 2004. The Conexant cards (brand: Longshine LCS-8531G contained Prism-GT chipset with an ISL3890 as MAC-Controller) are controlled by the prism54.org device driver (date 28-06-2004, firmware 1.0.4.3.arm). During each measurement both the sender and monitor were equipped with cards of the same brand. We also altered the notebook to study the impact of the CPU speed: An Asus Centrino 1.5GHz and an Amilo Celeron 850MHz notebook were used. To gather the packet traces, we used tcpdump and libpcap instead of Snuffle.

Configuration: The measurements were conducted in seven different configurations to study the impact of the WLAN card, CPU clock and modulation type. We used the default configuration of the WLAN cards and access point but changed the supported standard to 802.11g and set the modulation type to either 36 or 54 Mbps (Table 12.3 and 12.4). The frame length of the data packets and the acknowledgements is 65 bytes and 14 bytes.

Time measurements: The Atheros and Prism54 chipsets support timestamps of received packets with resolution of 1 μs similar to the Prism II chipset. Thus, again, a second notebook near the sender is required to measure both the sending and receiving timestamps. Also, we modified the device drivers to record the reception of a packet. After each interrupt, which is generated to notify the operating system about received or transmitted packets, the timestamps are saved. The time was measured by a libpcap timestamp. We also used a feature of Intel CPUs, which counts the CPU clock cycles. Linux supports reading the timestamp counter (TSC) with the `rdtsc(...)` function if the OS kernel has support included.

Data collection & processing: Tcpdump recorded the packet trace and wrote them to files. After the measurements we used tcpdump to convert these files to text files. Tcpdump had to been modified in order to print out the prism link-layer headers.

Table 12.4.: Configuration of the WLAN cards on a Linux system.

```
Sender configuration:
> iwpriv ath0|eth0 mode 3           # 802.11g mode
> iwconfig ath0 rate 36M            # (or 54M) set a fix tx rate (Atheros)
> iwpriv eth0 rate 36M              # (or 54M) set a fix tx rate (Prism54)
> ping -i 0.01 -s 1 $IPADDR         # send pings each 10 ms
```

```
Monitor configuration:
> iwconfig ath0|eth0 mode monitor   # Monitormodus
> sysctl -w devath.ctlpkt=-2        # trace all headers (Atheros)
> iwpriv eth0 set_prismhdr 1        # trace all headers (Prism54)
> tcpdump -i ath0|eth0 -c 20000     # trace 20000 packets including
-n -e -s 231 -tt --w trace_filename # link layer header
```

For statistical analysis the R project software turned out to be quite efficient. Thus, this time we applied R programs to calculate the mean, variance and autocorrelation of the data.

Results: Similarly to the first effort we calculated the distance from the time delay measurements. Figure 12.16 displays the remote (blue) and local delay (red) measurements, the number of overall observations (#) and the correlation coefficient (R) for the given configuration. A clear correlation between the actual distance and calculated distance can be identified. In the middle graph, one can see that the larger the distance (hence the worse the link quality), the larger the confidence interval becomes.

Figure 12.14 and 12.15 show the relation between distance and signal strength. The received signal strength (blue lines) decreases with distance. The correlation coefficient (R) is also given but one should consider that signal strength usually decays exponentially. Thus, the correlation coefficient should not be compared directly with Figure 12.16. Our measurement data also shows that the signal strength of the received data packets and received acknowledgement packets are nearly the same regardless of the packet length.

We also calculated the distances using time measurements in the interrupt routine. We could not identify which time gathering method is better. There was a slightly better result using a faster CPU. However, measuring the propagation time with the interrupt routine is too imprecise.

Analysis: We calculated the autocorrelation of local and remote delays (Figure 12.17). We found only high and alternating correlation coefficients if we used the Prism chipsets. With increasing distance and increasing error rate, the pattern vanishes. At greater distances, the observations, which are only based on successful transmissions, might not follow each other after exactly 20 ms but after a multiple of 20 ms. Thus, we can conclude that the effect is

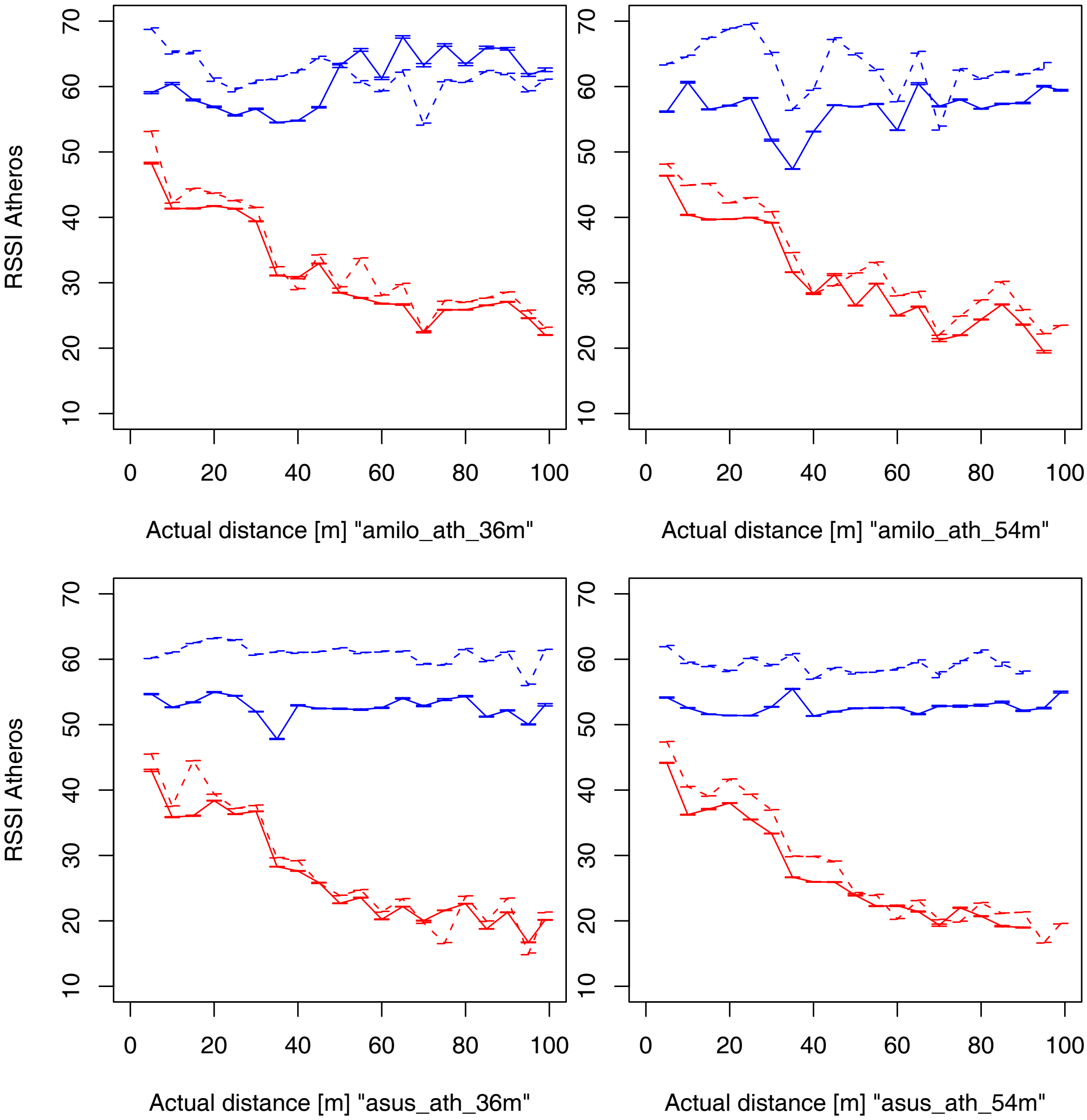


Figure 12.14.: Atheros:Received signal strength indication (RSSI) vs. actual distance (plus 95% confidence intervals). Blue=remote packets' RSSI; red=local packets' RSSI; lines=data packets; dotted=acknowledgements.

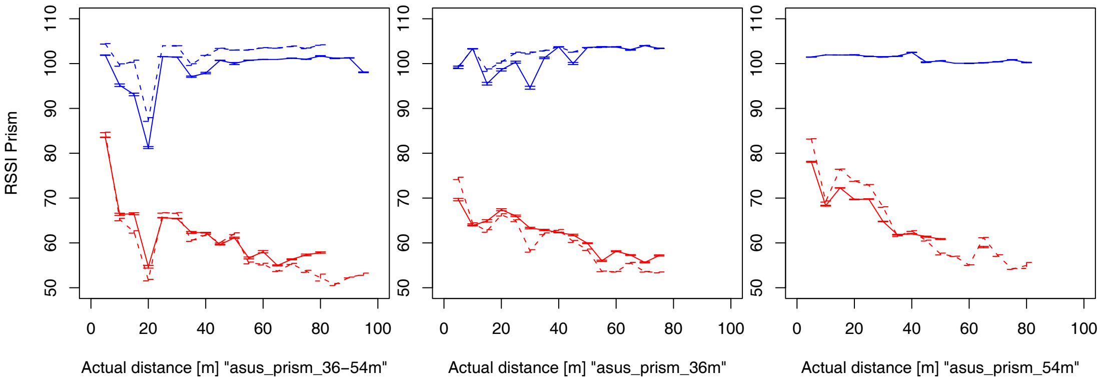


Figure 12.15.: PrismGT: Received signal strength indication (RSSI) vs. actual distance (plus 95% confidence intervals). Blue=remote packets' RSSI; red=local packets' RSSI; lines=data packets; dotted=acknowledgements.

rather due to the elapsed time than to the number of successful transmissions.

We measured at each distance between 5 to 15 minutes. Is it really required to measure that long? In Figure 12.18 and 12.19 we consider only a sub setup of all *rtt* observations. We display the cross correlation R and the standard error over the number of observations per distance. With 500 to 1000 observations per position nearly the optimal accuracy is achieved. If one assumes that a packet transmission takes 1 ms, the position can be estimated after 1 s of continuous transmission.

12.5. Conclusion

We presented an algorithm to measure the air propagation time of IEEE 802.11 packets with a higher accuracy. Using two different experimental setups, we determined the precision of round trip time measurements. We used commercial WLAN cards, supporting IEEE 802.11b and 802.11g, implemented with three different WIFI chip sets. We have shown that such time measurements are possible even with off-the-shelf, commercial WLAN equipment and without additional signal processing hardware.

To overcome the low resolution of the clocks, numerous observations have to be combined and smoothened. This can be carried out during ongoing data transmission at no additional cost. We explained why smoothing indeed helps to enhance the resolution of the time difference measurement so that distance estimates are possible. This effect can be due to the presence of measurement noise and to the beat frequency resulting from drifting clocks. To the best of our knowledge, especially the latter explanation is novel.

Our finding suggests that instead of RSSI the round trip time should be measured as it is correlated more strongly with distance. (In our gymnasium measurement the RSSI has

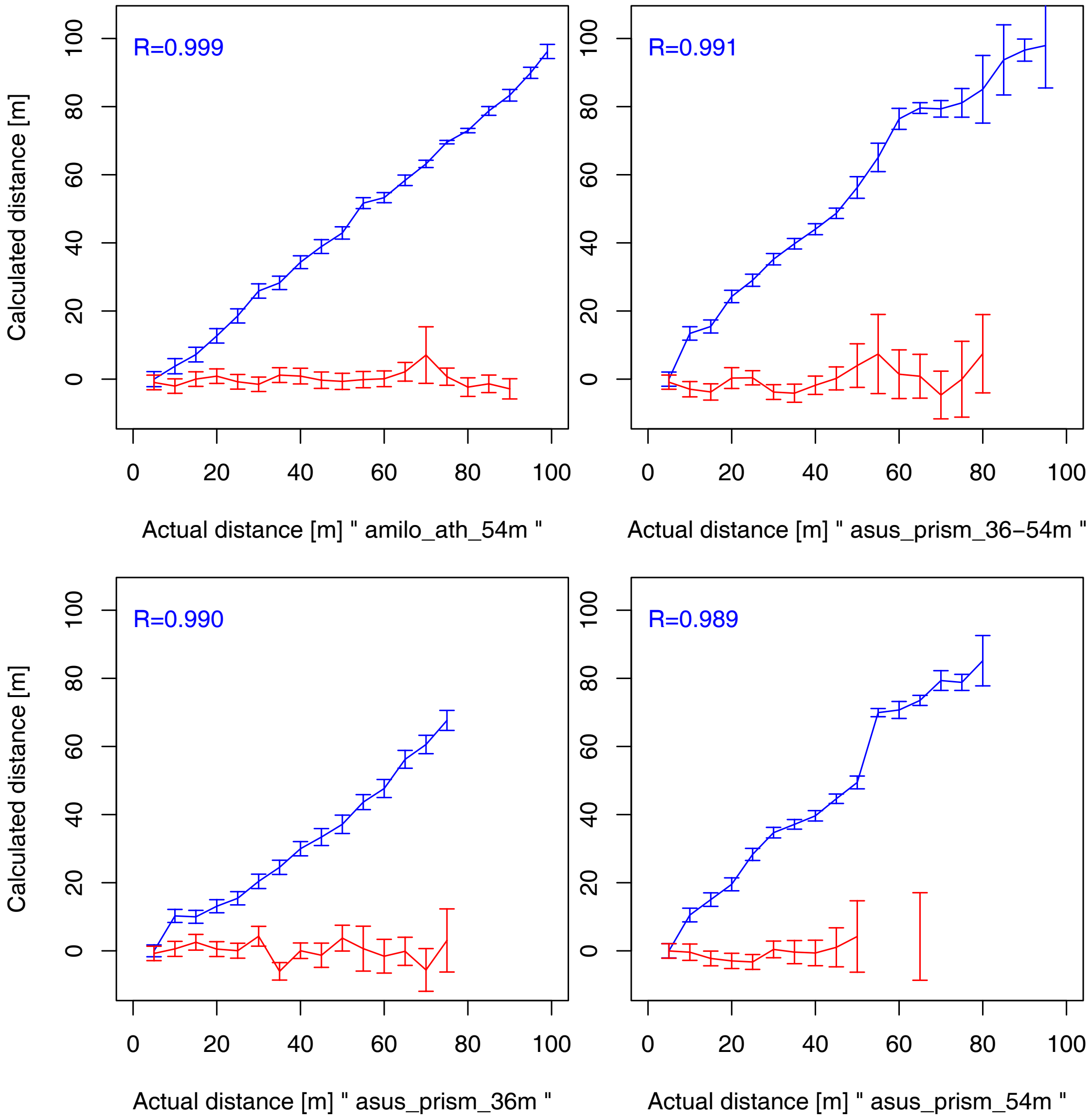


Figure 12.16.: Propagation delay (=calculated distance) vs. actual distance (plus 95% conf. intervals). (blue/upper lines=biased remote delay, red/lower lines=biased local delays). Each value is based on at least 1000 observations.

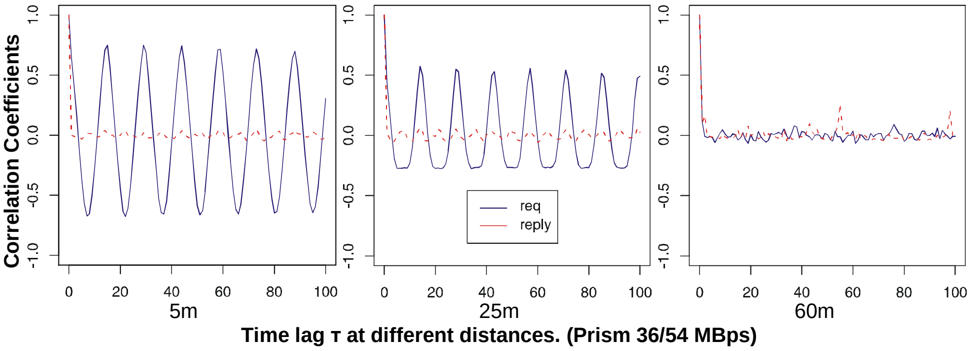


Figure 12.17.: Autocorrelation of local and remote delay. A delay of at minimal 20 ms is present between two observations.

not been useful to identify the distance because – due to reflections – the attenuation varied largely.)

In our measurements we had to use a third monitor node to record the timestamp of outgoing packets. It would be straightforward to alter the software and firmware of WLAN cards to record transmission timestamps, too. However, due to legal constraints, we were not able to implement these changes. We expect that WLAN chipset manufactures will provide firmware updates to support precise timestamps as they will benefit from customers using WLAN for location-aware services. Until then, we are required to use the third monitoring node.

The contribution of this work is to show that neither synchronized, precise clocks nor special hardware is required if the propagation delay between two WLAN nodes is to be measured. This allows the implementation of easy-to-use, cheap and precise indoor positioning systems, which do not require maps containing signal strength distributions. However, WLAN chipset manufacturers should update their firmware so that it reports the round trip time of packets with an accuracy of at least 1 μs. Then, a 1000 packets transmission – achievable in less than one second – can measure the distance with an error deviation of less than 8 m.

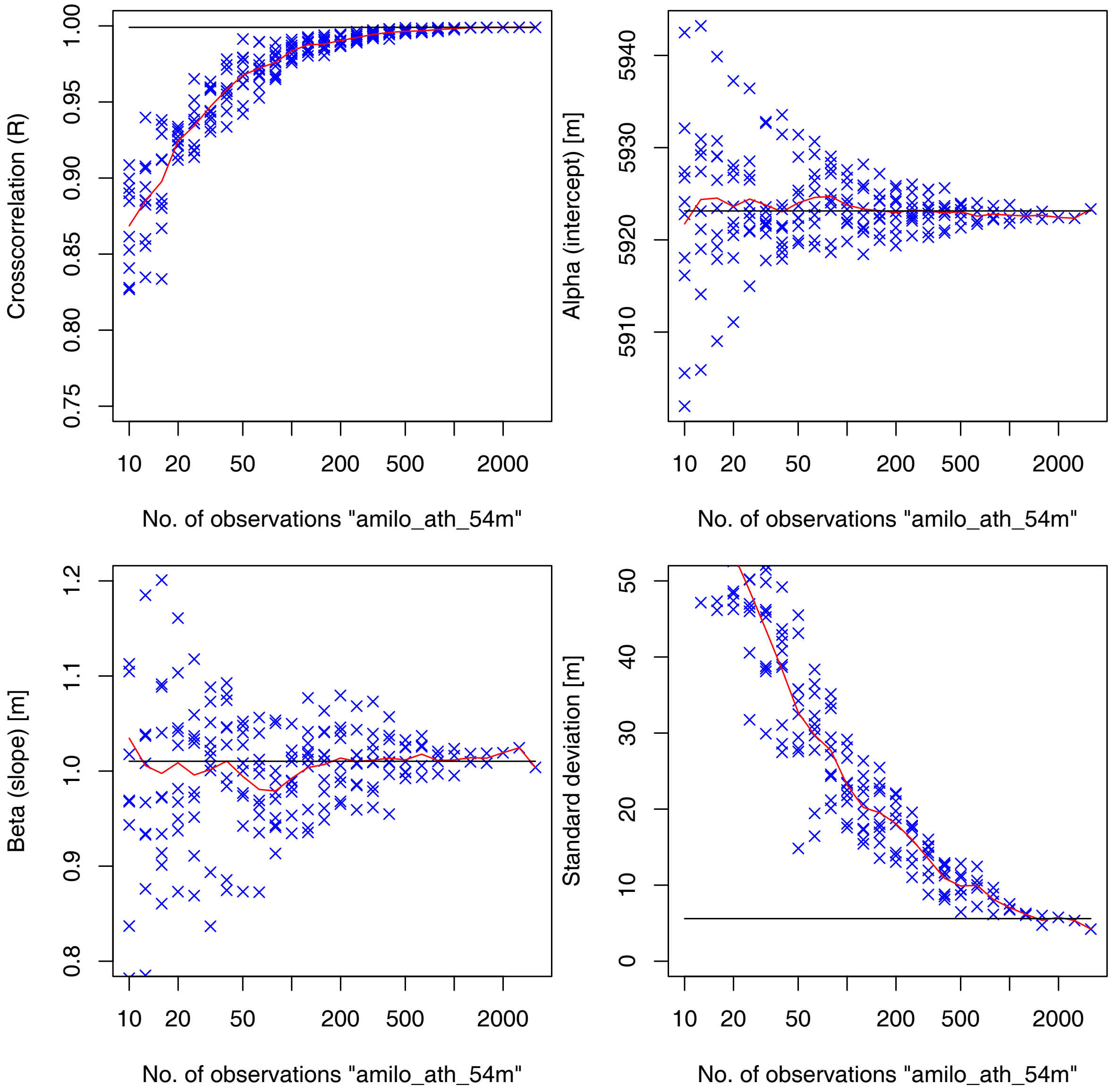


Figure 12.18.: The accuracy (cross correlation, alpha, beta, and standard deviation) over the number of observations per position.

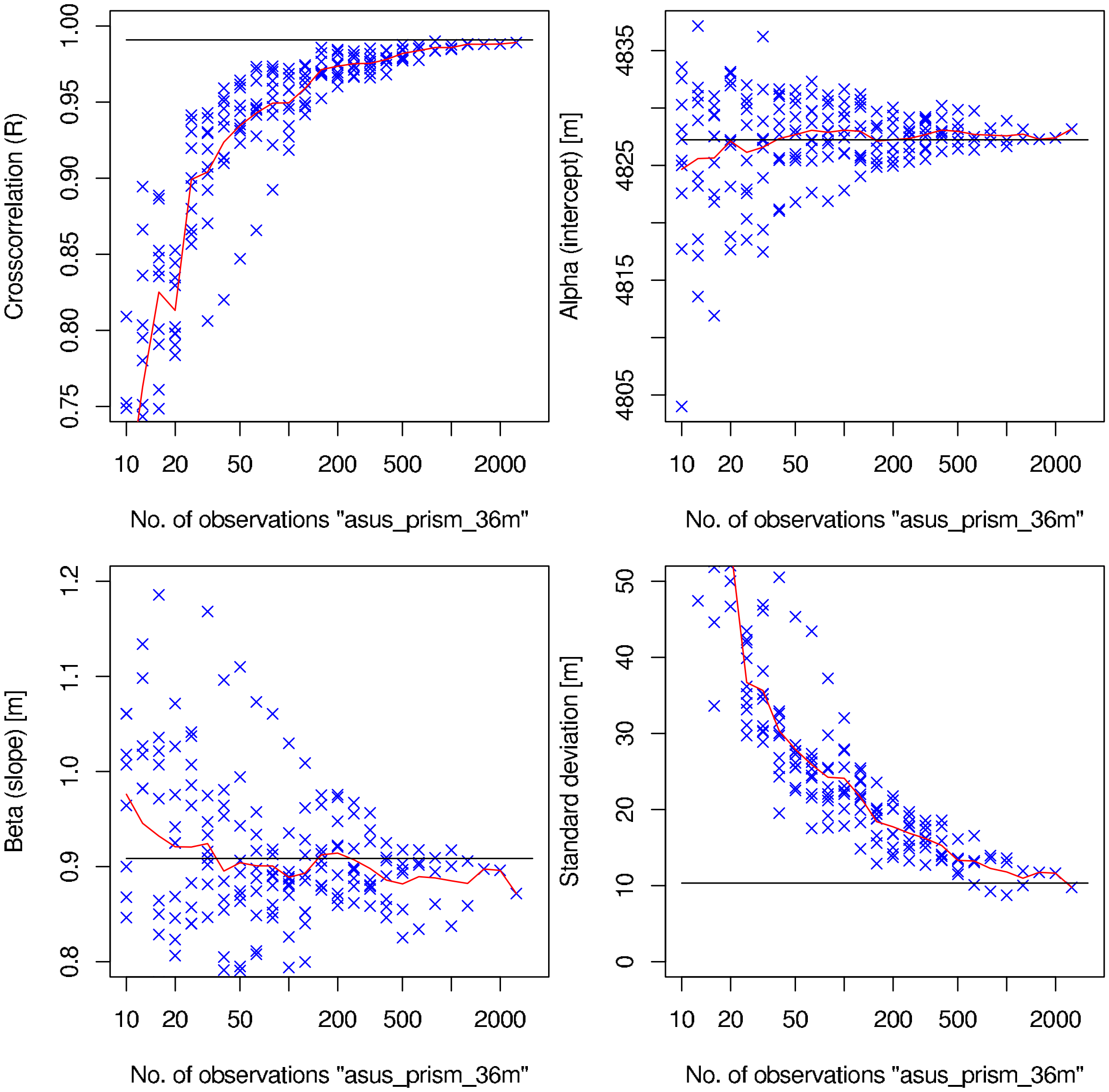


Figure 12.19.: As Figure 12.18 but for the Prism chipset.

13. Summary

This thesis proposes methods to enhance the efficiency of packetized voice communication over wireless links. Novel and substantial achievements have been achieved as summarised in the following:

We presented an improved approach to assess the quality of VoIP transmissions. We identified important sources of quality degradation that can occur in VoIP systems: Especially the impact of playout rescheduling and non-random packet losses have not been addressed in previous approaches. We combined PESQ, the E-model, different coding schemes and playout schedulers to analyze VoIP packet traces. PESQ was verified with formal listening-only tests to identify its prediction accuracy. Thus, we are able to predict the quality of VoIP transmissions at a high precision that has not been reached before.

At the application layer, we developed quality models that help to parameterize adaptive VoIP applications and algorithms so that they can achieve high perceptual quality ratings. Using this model, we demonstrated that as soon as bandwidth becomes limited it is more efficient to change the packet rate rather than the coding rate. Also our results also indicate that a playout buffer should not adjust its playout delay if single delay spikes occur.

We investigated the impact of individual packet losses on the perceptual speech quality. An off-line measurement procedure is described, which predicts the impact of loss on speech quality and quantifies this impact. We developed a metric and an aggregation function that calculates the impact multiple frame losses by considering the importance of the respective single frames. Also, an algorithm is presented that determines the importance of speech frames in real-time.

Using the knowledge of packet importance we showed that significant performance gains can be achieved if only important packets are transmitted. Using these algorithms on mobile phones, for example, can reduce transmission energy significantly and thus extend their battery lifetime.

On the physical and data link we conducted many experimental measurements. We explored to what extent slow user motion influences the wireless link quality. We also proved that the propagation delay of WLAN packets can be measured precisely using today's commercial, inexpensive equipment.

At the MAC layer we provided an open-source ns-2 implementation of IEEE 802.11e EDCA, which is used in many research projects and cited in many publications. Also, WLANs ability to support voice traffic was qualitative and quantitative studied.

13.1. Impact on research

In many areas, further research and enhancements is possible because in this new field of research only a fraction of all questions were answered. For example, the following issues can be addressed in future research:

- The perceptual quality models for VoIP can be further enhanced while reducing their complexity.
- Also, the algorithms to classify the importance one or multiple speech frames can be improved further.
- The distance measurement algorithm can be extended to support the location tracking of wireless nodes.
- A VoIP over WLAN system can be developed that takes advantage of packet importances to increase the call quality, cell capacity, and decrease the energy consumption.

This thesis will give our future research a solid foundation. Also other researchers are already inspired by some of this work, as an increasing number of citations show.

13.2. Impact on standards

The algorithms to assess of the quality VoIP flows are currently standardized in the ITU. Our results contributed to the ITU E-Model and the ITU P.VTQ standards. Currently we continue to follow and participate in the development and enhancement of ITU standards.

Our work also suggests to standardized playout schedulers as it would easy the assessment of VoIP packet traces. Playout schedulers are a well understood field of research. Thus, chosen the best algorithm should be straight forward, especially with the help of the novel quality assessment tools. We think that an playout scheduler standard might be reached in few years. However, until now, work has not started.

This approach outperforms the previously used signal strength indications and is currently considered to be included IEEE standards and is also followed by Intel research [61, 170].

13.3. Impact on industry

A most interesting question is whether this thesis helps the development of new products. A promising area for patenting are the packet classification algorithms. Patenting these algorithms and selling them to producers of wireless telephone equipment promises economical success because they can improve the efficiency of wireless communication. Thus, founding a start-up company, which focuses on the development of semantic data-link protocol and algorithms to classify VoIP packets, would be a worthwhile goal.

List of Personal Publications

Book chapters and journal publications

[88] C. Hoene, H. Karl, and A. Wolisz, “A Perceptual Quality Model Intended For Adaptive VoIP Applications”, International Journal of Communication Systems, Wiley, 2005.

[183] H. Sanneck, W. Mohr, L. Le, C. Hoene, and A. Wolisz, ”Quality of Service Support for Voice Over IP over Wireless”, In S. Dixit and R. Prasad, editors, Wireless IP and Building the Mobile Internet, chapter 10 Artech House, Norwood, MA, USA, December 2002.

[217] A. Willig, M. Kubisch, C. Hoene, and A. Wolisz, ”Measurements of a Wireless Link in an Industrial Environment using an IEEE 802.11-Compliant Physical Layer”, IEEE Transactions on Industrial Electronics, vol. 43, no. 6, pp. 1265-1282, December 2002.

Articals in conference proceedings

[93] C. Hoene, S. Wiethölter, and A. Wolisz, ”Calculation of Speech Quality by Aggregating the Impacts of Individual Frame Losses”, In Proc. of Thirteenth International Workshop on Quality of Service (IWQoS 2005), Passau, Germany, June 2005.

[68] A. Günther and Ch. Hoene, ”Measuring Round Trip Times to Determine the Distance between WLAN Nodes”, In Networking 2005, Waterloo, Canada, May 2005.

[90] C. Hoene, S. Wiethölter, and A. Wolisz, ”Predicting the Perceptual Service Quality Using a Trace of VoIP Packets”, In Proc. of Fifth International Workshop on Quality of future Internet Services (QofIS’04), Barcelona, Spain, September 2004.

[87] C. Hoene, H. Karl, and A. Wolisz, ”A Perceptual Quality Model for Adaptive VoIP Applications”, In Proc. of International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS’04), San Jose, California, USA, July 2004, Paper won the Best Paper Award of the conference.

[92] C. Hoene and E. Dulamsuren-Lalla, ”Predicting Performance of PESQ in Case of Single Frame Losses”, In Proc. of MESAQIN 2004, Prague, CZ, June 2004.

[86] C. Hoene, A. Günther, and A. Wolisz, ”Measuring the Impact of Slow User Motion on Packet Loss and Delay over IEEE 802.11b Wireless Links”, In Proc. of Workshop on Wireless Local Networks (WLN) 2003, Bonn, Germany, October 2003.

[4] A. Aguiar, C. Hoene, J. Klaue, H. Karl, H. Miesmer, and A. Wolisz, ”Channel-aware Schedulers for VoIP and MPEG4 based on Channel Prediction”, In Proc. of 8th Intl. Workshop on Mobile Multimedia Communications (MoMuC'03), October 2003.

[89] C. Hoene, B. Rathke, and A. Wolisz, ”On the Importance of a VoIP Packet”, In Proc. of ISCA Tutorial and Research Workshop on the Auditory Quality of Systems, Herne, Germany, April 2003.

[84] C. Hoene, I. Carreras, T. Chen, and A. Wolisz, ”Design and Deployment of Link-Layer Boosters for Per-flow Improvement of QoS in Wireless Internet Access”, In Proc. of European Wireless 2002, pp. 170-176, Florence, Italy, February 2002.

[85] C. Hoene, I. Carreras, and A. Wolisz, ”Voice Over IP: Improving the Quality Over Wireless LAN by Adopting a Booster Mechanism - An Experimental Approach”, In Proc. of SPIE 2001 - Voice Over IP (VoIP) Technology, pp. 157-168, Denver, Colorado, USA, August 2001.

Technical reports

[67] A. Günther and Ch. Hoene, ”Measuring Round trip Times to Determine the Distance between WLAN Nodes”, Technical Report TKN-04-016, Telecommunication Networks Group, Technische Universität Berlin, December 2004.

[215] S. Wiethölter, C. Hoene, and A. Wolisz, ”Perceptual Quality of Internet Telephony over IEEE 802.11e Supporting Enhanced DCF and Contention Free Bursting”, Technical Report TKN-04-011, Telecommunication Networks Group, Technische Universität Berlin, May 2004.

[213] S. Wiethölter and C. Hoene, ”Design and Verification of an IEEE 802.11e EDCF Simulation Model in ns-2.26”, Technical Report TKN-03-019, Telecommunication Networks Group, Technische Universität Berlin, November 2003.

Other publications

[150] S. Möller and C. Hoene, ”Information About a New Method For Deriving the Transmission Rating Factor R From MOS in Closed Form”, ITU - Telecommunication Standardization Sector, May 2002, Temporary Document for the Study Group 12.

Bibliography

[1] *Universal Mobile Telecommunications System (UMTS), AMR Speech Codec, General Description*, 3GPP Std. TS 26.071 Version 5.0.0 Release 6, June 2002.

[2] *Universal Mobile Telecommunications System (UMTS), Adaptive Multi-Rate (AMR) speech codec;Source controlled rate operation*, 3GPP Std. TS 26.093 Version 6.0.0 Release 6, Mar. 2003.

[3] *Universal Mobile Telecommunications System (UMTS), Adaptive Multi-Rate (AMR) speech codec; Voice Activity Detector (VAD)*, 3GPP Std. TS 26.094 Version 6.0.0 Release 6, Dec. 2004.

[4] A. Aguiar, C. Hoene, J. Klaue, H. Karl, H. Miesmer, and A. Wolisz, "Channel-aware schedulers for VoIP and MPEG4 based on channel prediction," in *Eight International Workshop on Mobile Multimedia Communications (MoMuC'03)*, Oct. 2003.

[5] J. Allnatt, *Transmitted-picture Assessment.* New York, USA: John Wiley & Sons, 1983.

[6] N. Alsindi, X. Li, and K. Pahlavan, "Performance of TOA estimation algorithms in different indoor multipath conditions," in *IEEE Wireless Communications and Networking Conference (WCNC 2004)*, vol. 1, Mar. 2004, pp. 495–500.

[7] C. Aras, J. Kurose, D. Reeves, and H. Schulzrinne, "Real-time communication in packet-switched networks," *Proceedings of the IEEE*, vol. 82, no. 1, pp. 122–139, 1994.

[8] M. Arranz, R. Aguero, L. Munoz, and P. Mahonen, "Behavior of UDP-based applications over IEEE 802.11 wireless networks," in *12th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 2, Sept. 2001, pp. 72–77.

[9] P. Bahl and V. N. Padmanabhan, "RADAR: an in-building RF-based user location and tracking system," in *Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2000)*, vol. 2, Tel-Aviv, Israel, Mar. 2000, pp. 775–784.

[10] A. Barberis, C. Casetti, J. C. De Martin, and M. Meo, "A simulation study of adaptive voice communications on IP networks," *Computer Communications*, vol. 24, no. 9, pp. 757–767, May 2001.

[11] G. Barberis, "Buffer sizing of a packet-voice receiver," *IEEE Transactions on Communications*, vol. 29, no. 2, pp. 152–156, Feb. 1981.

[12] G. Barberis, M. Calabrese, L. Lambarelli, and D. Roffinella, "Coded speech in packet-switched networks: Models and experiments," *IEEE Journal on Selected Areas in Communications*, vol. 1, no. 6, pp. 1028–1038, Dec. 1983.

[13] G. Barberis and D. Pazzaglia, "Analysis and optimal design of a packet-voice receiver," *IEEE Transactions on Communications*, vol. 28, no. 2, pp. 217–227, 1980.

[14] J. A. Bargh and K. Y. A. McKenna, "The internet and social life," *Annual Review of Psychology*, vol. 55, no. 1, pp. 573–590, 2004.

[15] J. G. Beerends, "Measuring the quality of speech and music codecs: An integrated psychoacoustic approach," in *98th Convention of the Audio Engineering Society, preprint 3945*, Jan. 1995.

[16] J. G. Beerends, A. P. Hekstra, A. W. Rix, and M. P. Hollier, "Perceptual evaluation of speech quality (PESQ), the new ITU standard for end-to-end speech quality assessment, part II - psychoacoustic model," *Journal of the Audio Engineering Society*, vol. 50, no. 10, pp. 765–778, Oct. 2002.

[17] J. G. Beerends and J. A. Stemerdink, "A perceptual speech-quality measure based on a psychoacoustic sound representation," *Journal of the Audio Engineering Society*, vol. 42, no. 3, pp. 115–123, Mar. 1994.

[18] J. G. Beerends and J. M. van Vugt, "An extension of PESQ for assessing the quality of speech degraded by severe time clipping and linear frequency response distortions," in *Joint conference of the German and French acoustical society, DAGA 2004*, Strasbourg, France, Mar. 2004.

[19] F. Beritelli, S. Casale, and G. Ruggeri, "Performance comparison between VBR speech coders for adaptive VoIP applications," *IEEE Communications Letters*, vol. 5, no. 10, pp. 423–425, 2001.

[20] T. Bially, B. Gold, and S. Seneff, "A technique for adaptive voice flow control in integrated packet networks," *IEEE Transactions on Communications*, vol. 28, no. 3, pp. 325–333, 1980.

[21] T. Bially, A. McLaughlin, and C. Weinstein, "Voice communication in integrated digital voice and data networks," *IEEE Transactions on Communications*, vol. 28, no. 9, pp. 1478–1490, 1980.

[22] U. Black, *Voice over IP*, 2nd ed. Prentice Hall, 2002.

[23] S. Blake, D. Black, M. Carlson, E. Davies, Z. Wang, and W. Weiss, "An architecture for differentiated services," RFC 2475, IETF Network Working Group, Dec. 1998.

[24] J.-C. Bolot, "End-to-end packet delay and loss behavior in the internet," in *Conference proceedings on Communications architectures, protocols and applications (SIGCOMM '93).* ACM Press, 1993, pp. 289–298.

[25] J.-C. Bolot, S. Fosse-Parisis, and D. F. Towsley, "Adaptive FEC-based error control for internet telephony," in *Eighteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '99)*, vol. 3, New York, NY, Apr. 1999, pp. 1453–1460.

[26] J.-C. Bolot and A. Vega-Garcia, “Control mechanisms for packet audio in the internet,” in *Fifteenth Annual Joint Conference of the IEEE Computer Societies (INFOCOM ’96)*, vol. 1, San Franisco, CA, Apr. 1996, pp. 232–239.

[27] J.-C. Bolot and A. Vega-Garcia. (1997) The case for FEC-based error control for packet audio in the internet. [Online]. Available: http://citeseer.ist.psu.edu/bolot97case.html

[28] J.-C. Bolot, H. Crepin, and A. V. Garcia, “Analysis of audio packet loss in the internet,” in *Proceedings of the 5th International Workshop on Network and Operating System Support for Digital Audio and Video (NOSSDAV ’95)*. London, UK: Springer-Verlag, May 1995, pp. 154–165.

[29] F. Bouchereau and D. Brady, “Bounds on range-resolution degradation using RSSI measurements,” in *IEEE International Conference on Communications (ICC’04)*, vol. 6, June 2004, pp. 3246–3250.

[30] C. Boutremans and J.-Y. L. Boudec, “Adaptive joint playout buffer and FEC adjustment for internet telephony,” in *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, vol. 1, San-Francisco, CA, Apr. 2003, pp. 652–662.

[31] P. T. Brady, “A model for generating on-off speech patterns in two-way conversation,” *The Bell System Technical Journal*, vol. 48, no. 9, pp. 2445–2472, Sept. 1969.

[32] G. Carle and E. Biersack, “Survey of error recovery techniques for ip-based audio-visualmulticast applications,” *IEEE Network*, vol. 11, no. 6, pp. 24–36, 1997.

[33] C. Casetti and C.-F. Chiasserini, “Improving fairness and throughput for voice traffic in 802.11e EDCA,” in *15th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC 2004)*, vol. 1, Barcellona, Spain, Sept. 2004, pp. 525–530.

[34] J. Chakareski, J. Apostolopoulos, S. Wee, W.-t. Tan, and B. Girod, “R-D hint tracks for low-complexity R-D optimized video streaming,” in *IEEE International Conference on Multimedia and Expo (ICME ’04)*, vol. 2, 2004, pp. 1387–1390.

[35] J. Chakareski and B. Girod, “Rate-distortion optimized packet scheduling and routing for media streaming with path diversity,” in *Data Compression Conference (DCC 2003)*, 2003, pp. 203–212.

[36] R. Chandramouli, K. P. Subbalakshmi, and N. Ranganathan, “Stochastic channel-adaptive rate control for wireless video transmission,” *Pattern Recognition Letters*, vol. 25, no. 7, pp. 793–806, 2004.

[37] C. Chang and A. Sahai, “Estimation bounds for localization,” in *First Annual IEEE Communications Society Conference on Sensor and Ad Hoc Communications and Networks (SECON 2004)*, Santa Clara, CA, Oct. 2004, pp. 415–424.

[38] C. Chen, M. Asawa, and B. Foreman, “Outdoor measurements on WaveLAN radio,” Path Lab, Department of EECS, U.C. Berkeley, Tech. Rep., 1995.

[39] D. Chen, S. Garg, M. Kappes, and K. S. Trivedi, "Supporting VBR VoIP traffic in IEEE 802.11 WLAN in PCF mode," Avaya Research Labs, Washington DC, Tech. Rep. ALR-2002-026, Aug. 2002.

[40] S. Choi, J. del Prado, S. S. N, and S. Mangold, "IEEE 802.11e contention-based channel access (EDCF) performance evaluation," in *IEEE International Conference on Communications (ICC '03)*, vol. 2, Anchorage, AL, May 2003, pp. 1151–1156.

[41] P. A. Chou and Z. Miao, "Rate-distortion optimized streaming of packetized media," Microsoft Research, Tech. Rep. MSR-TR-2001-35, Feb. 2001.

[42] C.-N. Chuah and R. H. Katz, "Characterizing packet audio streams from internet multimedia applications," in *IEEE International Conference on Communications (ICC 2002)*, vol. 2, College Park, MD, Apr. 2002, pp. 1199–1203.

[43] A. Clark, "Modeling the effects of burst packet loss and recency on subjective voice quality," in *Internet Telephony Workshop*, New York, NY, Mar. 2001, pp. 123–127.

[44] 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, pp. 277–280.

[45] L. Cong and W. Zhuang, "Nonline-of-sight error mitigation in mobile location," in *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004)*, vol. 1, Hong Kong, Mar. 2004, pp. 650–659.

[46] F. D'Agostino, E. Masala, L. Farinetti, and J. De Martin, "A simulative study of analysis-by-synthesis perceptual video classification and transmission over diffserv IP networks," in *IEEE International Conference on Communications (ICC '03)*, vol. 1, 2003, pp. 572–576.

[47] L. A. DaSilva, D. W. Petr, and V. Frost, "A class-oriented replacement technique for lost speech packets," in *Eighth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '89)*, 1989, pp. 1098–1105.

[48] J. C. De Martin, "Source-driven packet marking for speech transmission over differentiated-services networks," in *Proceedings of the IEEE International Conference on Audio, Speech and Signal Processing*, Salt Lake City, UT, May 2001, pp. 753–756.

[49] S. Deering and R. Hinden, "Internet protocol, version 6 (IPv6)," RFC 2460, Dec. 1998.

[50] E. Diethorn, "A low-complexity, background-noise reduction preprocessor forspeech encoder," in *IEEE Workshop on Speech Coding For Telecommunications Proceeding*, 1997, pp. 45–46.

[51] J. Eberspächer, "Konvergenz der Kommunikationsnetze: Wird das Internet alles übernehmen?" 2004.

[52] E. Elnahrawy, X. Li, and R. P. Martin, "The limits of localization using signal strength: A comparative study," in *First IEEE International Conference on Sensor and Ad hoc Communications and Networks (SECON 2004)*, Santa Clara, CA, Oct. 2004.

[53] P. Enge and P. Misra, Eds., *Special issue on Global Positioning System.* IEEE, Jan. 1999.

[54] G. Enzner and P. Vary, "A soft-partitioned frequency-domain adaptive filter for acoustic echo cancellation," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '03)*, vol. 5, 2003, pp. V–393–6.

[55] G. Erber, T. Köhler, C. Lattemann, B. Preissl, and J. Rentmeister, "Rahmenbedingungen für eine Breitbandoffensive in Deutschland," p. 83, Jan. 2004, im Auftrag der Deutschen Telekom AG, T-Com.

[56] T. Eren, D. Goldenberg, W. Whiteley, Y. R. Yang, A. S. Morse, B. Anderson, and P. Belhumeu, "Rigidity, computation, and randomization in network localization," in *Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004)*, vol. 4, Hong Kong, Mar. 2004, pp. 2673–2684.

[57] C. Fraleigh, F. Tobagi, and C. Diot, "Provisioning IP backbone networks to support latency sensitive traffic," in *Twenty-Second Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2003)*, vol. 1, San Francisco, CA, Apr. 2003, pp. 375–385.

[58] L. Gammaitoni, P. Hanggi, P. Jung, and F. Marchesoni, "Stochastic resonance," *Reviews of Modern Physics*, vol. 70, no. 1, pp. 223–287, 1998.

[59] P. Garg, R. Doshi, R. Greene, M. Baker, M. Malek, and X. Cheng, "Using IEEE 802.11e MAC for QoS over Wireless," in *Conference Proceedings of the 2003 IEEE International Performance, Computing, and Communications Conference (IPCCC 2003)*, Phoenix, AZ, Apr. 2003, pp. 537–542.

[60] S. Garg and M. Kappes, "On the Throughput of 802.11b Networks for VoIP," Avaya Labs Research, Washington DC, Tech. Rep. ALR-2002-012, Mar. 2002.

[61] S. Golden, "Key issues about WIFI location," IEEE 802.11-05/1079r0, Nov. 2005.

[62] D. Goodman and R. Nash, "Subjective quality of the same speech transmission conditions in seven different countries," in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP '82)*, vol. 7, 1982, pp. 984–987.

[63] J. Gruber and L. Strawczynski, "Subjective effects of variable delay and speech clipping in dynamically managed voice systems," *IEEE Transactions on Communications*, vol. 33, no. 8, pp. 801–808, 1985.

[64] Y. Gwon, R. Jain, and T. Kawahara, "Robust indoor location estimation of stationary and mobile users," in *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004)*, vol. 2, Hong Kong, Mar. 2004, pp. 1032–1043.

[65] M. Görtz, R. Ackermann, J. Schmitt, and R. Steinmetz, "Context-aware communication services: A framework for building enhanced IP telephony services," in *13th International Conference on Computer Communications and Networks (ICCCN 2004)*, 2004, pp. 535–540.

[66] A. Günther, "Accuracy of propagation delay measurements in wireless LANs," intermediate thesis, Telecommunication Networks Group, Technische Universität Berlin, Aug. 2004.

[67] A. Günther and C. Hoene, "Measuring round trip times to determine the distance between WLAN nodes," Telecommunication Networks Group, Technische Universität Berlin, Tech. Rep. TKN-04-016, Dec. 2004. [Online]. Available: http://www.tkn.tu-berlin.de/publications/papers/tkn_04_16_paper3.pdf

[68] A. Günther and C. Hoene, "Measuring round trip times to determine the distance between WLAN nodes," in *Networking 2005*, Waterloo, Canada, May 2005.

[69] A. Haeberlen, E. Flannery, A. M. Ladd, A. Rudys, D. S. Wallach, and L. E. Kavraki, "Practical robust localization over large-scale 802.11 wireless networks," in *Tenth annual international conference on Mobile computing and networking (MOBICOM)*. Philadelphia, PA: ACM Press, Sept. 2004, pp. 70–84.

[70] O. Hagsand, K. Hanson, and I. Marsh, "Measuring internet telephony quality: Where are we today?" in *Global Telecommunications Conference (GLOBECOM '99)*, vol. 3, Rio De Janeiro, Brazil, Dec. 1999, pp. 1838–1842.

[71] F. Hammer, P. Reichl, and T. Ziegler, "Where packet traces meet speech samples: An instrumental approach to perceptual QoS evaluation of VoIP," in *Twelfth IEEE International Workshop on Quality of Service (IWQOS 2004)*, Montreal, Canada, June 2004, pp. 273–280.

[72] V. Hardman, M. A. Sasse, M. Handley, and A. Watson, "Reliable audio for use over the Internet," in *Internet Society's International Networking Conference (INET)*, Oahu, Hawaii, June 1995, pp. 171–178.

[73] G. Harmer, B. Davis, and D. Abbott, "A review of stochastic resonance: circuits and measurement," *IEEE Transactions on Instrumentation and Measurement*, vol. 51, no. 2, pp. 299–309, Apr. 2002.

[74] M. Harteneck and R. Stewart, "Acoustic echo cancelation using a pseudo-linear regression and QR-decomposition," in *IEEE International Symposium on Circuits and Systems (ISCAS '96)*, vol. 2, 1996, pp. 233–236.

[75] T. He, C. Huang, B. M. Blum, J. A. Stankovic, and T. Abdelzaher, "Range-free localization schemes for large scale sensor networks," in *9th annual international conference on Mobile computing and networking (MOBICOM)*. San Diego, CA: ACM Press, Sept. 2003, pp. 81–95.

[76] P. Heitkämper, "Optimization of an acoustic echo canceller combined with adaptive gain control," in *International Conference on Acoustics, Speech, and Signal Processing (ICASSP-95)*, vol. 5, 1995, pp. 3047–3050.

[77] O. Hersent, D. Gurle, and J.-P. Petit, *IP Telephony: Packet-based Multimedia Communications Systems*. Addision Wesley, 1999.

[78] J. Hightower and G. Borriello, "Location systems for ubiquitous computing," *IEEE Computer*, vol. 34, no. 8, pp. 57–66, Aug. 2001.

[79] T. Hindelang, M. Kaindl, J. Hagenauer, M. Schmautz, and W. Xu, “Improved channel coding and estimation for adaptive multi rate (AMR) speech transmission,” in *IEEE 51st Vehicular Technology Conference Proceedings (VTC 2000-Spring)*, vol. 2, Tokyo, Japan, 2000, pp. 1210–1214.

[80] C. Hoene. (2001, Mar.) Easysnuffle - a tool to measure the performance of multimedia flows over IEEE 802.11b. [Online]. Available: http://www.tkn.tu-berlin.de/research/easysnuffle/

[81] C. Hoene. (2003, Nov.) Batch Distribution - a tool to run a batch of computation jobs distributed. [Online]. Available: http://www.tkn.tu-berlin.de/equipment/bd/

[82] C. Hoene. (2004, July) A perceptual quality model for adaptive VoIP applications: Software distribution. [Online]. Available: http://www.tkn.tu-berlin.de/research/simquamol/

[83] C. Hoene. (2004, Apr.) Software tool Mongolia. [Online]. Available: http://www.tkn.tu-berlin.de/research/mongolia/

[84] C. Hoene, I. Carreras, T. Chen, and A. Wolisz, “Design and deployment of link-layer boosters for per-flow improvement of qos in wireless internet access,” in *European Wireless 2002*, Florence, Italy, Feb. 2002, pp. 170–176.

[85] C. Hoene, I. Carreras, and A. Wolisz, “Voice Over IP: Improving the Quality Over Wireless LAN by Adopting a Booster Mechanism - An Experimental Approach,” in *SPIE 2001 - Voice Over IP (VoIP) Technology*, Denver, CO, Aug. 2001, pp. 157–168.

[86] C. Hoene, A. Günther, and A. Wolisz, “Measuring the impact of slow user motion on packet loss and delay over ieee 802.11b wireless links,” in *Workshop on Wireless Local Networks (WLN 2003)*, Bonn, Germany, Oct. 2003.

[87] C. Hoene, H. Karl, and A. Wolisz, “A perceptual quality model for adaptive VoIP applications,” in *International Symposium on Performance Evaluation of Computer and Telecommunication Systems (SPECTS’04)*, San Jose, CA, July 2004.

[88] C. Hoene, H. Karl, and A. Wolisz, “A perceptual quality model intended for adaptive VoIP applications,” *International Journal of Communication Systems, Wiley*, Aug. 2005.

[89] C. Hoene, B. Rathke, and A. Wolisz, “On the importance of a VoIP packet,” in *ISCA Tutorial and Research Workshop on the Auditory Quality of Systems*, Mont-Cenis, Germany, Apr. 2003.

[90] C. Hoene, S. Wiethölter, and A. Wolisz, “Predicting the perceptual service quality using a trace of VoIP packets,” in *Fifth International Workshop on Quality of future Internet Services (QofIS’04)*, Barcelona, Spain, Sept. 2004.

[91] C. Hoene. (2004, June) Simulating playout schedulers for VoIP - software package. [Online]. Available: http://www.tkn.tu-berlin.de/research/qofis/

[92] C. Hoene and E. Dulamsuren-Lalla, "Predicting performance of PESQ in case of single frame losses," in *Measurement of Speech and Audio Quality in Networks Workshop (MESAQIN)*, Prague, CZ, June 2004.

[93] C. Hoene, S. Wiethölter, and A. Wolisz, "Calculation of speech quality by aggregating the impacts of individual frame losses," in *Thirteenth International Workshop on Quality of Service (IWQoS 2005)*, Passau, Germany, June 2005.

[94] J. Holub, R. Smid, and J. Ocenasek, "Processing power optimisation for PESQ," in *Measurement of Speech and Audio Quality in Networks (MESAQIN)*, Prague, CZ, 2004, pp. 57–60.

[95] *Wireless LAN Medium Access Control (MAC) and Physical Layer (PHY) Specification*, IEEE Std. 802.11, 1997.

[96] "Draft supplement to IEEE standard 802.11-1999: Medium access control (MAC) enhancements for quality of service (QoS)," 2003.

[97] *Pentium II Application Note: Using the RDTSC instruction for performance monitoring*, Intel Corporation, 1998. [Online]. Available: http://developer.intel.com/drg/pentiumII/appnotes/RDTSCPM1.HTM

[98] *Pulse Code Modulation (PCM) of Voice Frequencies*, Recommendation G.711, ITU-T Std., Nov. 1988.

[99] *Artificial conversational speech*, Recommendation P.59, ITU-T Std., May 1993.

[100] *Methods for subjective determination of transmission quality*, Recommendation P.800, ITU-T Std., Aug. 1996.

[101] *Modulated noise reference unit (MNRU)*, Recommendation P.810, ITU-T Std., Feb. 1996.

[102] *Coded-speech Database*, Recommendation P.Supplement 23, ITU-T Std., Feb. 1998.

[103] *Application of the E-model: A planning guide*, Recommendation G.108, ITU-T Std., Sept. 1999.

[104] *A High Quality Low-Complexity Algorithm for Packet Loss Concealment with G.711*, Recommendation G.711 Appendix I, ITU-T Std., Sept. 1999.

[105] *The E-Model, a Computational Model for Use in Transmission Planning*, Recommendation G.107, ITU-T Std., May 2000.

[106] *Methodology for derivation of equipment impairment factors from subjective listening-only tests*, Recommendation P.833, ITU-T Std., Feb. 2001.

[107] *Perceptual Evaluation of Speech Quality (PESQ), an Objective Method for End-To-End Speech Quality Assessment of Narrowband Telephone Networks and Speech Codecs*, Recommendation P.862, ITU-T Std., Feb. 2001.

[108] *Mean Opinion Score (MOS) terminology*, Recommendation P.800.1, ITU-T Std., Mar. 2003.

[109] *Packet-based multimedia communications systems*, Recommendation H.323, ITU-T Std., July 2003.

[110] *Single-ended method for objective speech quality assessment in narrow-band telephony applications*, Recommendation P.563, ITU-T Std., May 2004.

[111] W. C. Jakes, *Microwave Mobile Communications.* New York: IEEE Press, 1974.

[112] W. Jiang and H. Schulzrinne, “Perceived quality of packet audio under bursty losses,” Department of Computer Science, Columbia University, New York, NY, Tech. Rep. CUCS-009-01, 2001.

[113] W. Jiang and H. Schulzrinne, “Analysis of on-off patterns in VoIP and their effect on voice traffic aggregation,” in *Ninth International Conference on Computer Communications and Networks*, 2000, pp. 82–87.

[114] W. Jiang and H. Schulzrinne, “Comparison and optimization of packet loss repair methods on voip perceived quality under bursty loss,” in *NOSSDAV*, 2002, pp. 73–81.

[115] W. Jiang and H. Schulzrinne, “Assessment of voip service availability in the current internet,” in *Passive & Active Measurement Workshop (PAM)*, San Diego, CA, Apr. 2003.

[116] M.-H. Jin, E. H.-K. Wu, Y.-B. Liao, and H.-C. Liao, “802.11-based positioning system for context aware applications,” in *IEEE Global Telecommunications Conference (GLOBECOM ’03)*, vol. 2, Dec. 2003, pp. 929–933.

[117] K. Kaemarungsi and P. Krishnamurthy, “Modeling of indoor positioning systems based on location fingerprinting,” in *Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004)*, vol. 2, Hong Kong, Mar. 2004, pp. 1012–1022.

[118] M. Kalman and B. Girod, “Modeling the delays of successively-transmitted internet packets,” in *IEEE International Conference on Multimedia and Expo (ICME ’04)*, vol. 3, 2004, pp. 2015–2018.

[119] M. Kalman, E. Steinbach, and B. Girod, “R-D optimized media streaming enhanced with adaptive media playout,” in *IEEE International Conference on Multimedia and Expo (ICME ’02)*, vol. 1, 2002, pp. 869–872.

[120] M. Kalman, E. Steinbach, and B. Girod, “Adaptive media playout for low-delay video streaming over error-prone channels,” *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 14, no. 6, pp. 841–851, 2004.

[121] T. Kawata, S. Shin, and A. G. Forte, “Using dynamic PCF to improve the capacity for VoIP traffic in IEEE 802.11 networks,” in *IEEE Wireless Communications and Networking Conference*, vol. 3, Mar. 2005, pp. 1589–1595.

[122] H. Kim, M.-J. Chae, and I. Kang, “The methods and the feasibility of frame grouping in internet telephony,” *IEICE Transactions on Communications*, vol. E85-B, no. 1, pp. 173–182, Jan. 2002.

[123] N. Kitawaki and K. Itoh, “Pure delay effects on speech quality in telecommunications,” *IEEE Journal on Selected Areas in Communications*, vol. 9, no. 4, pp. 586–593, 1991.

[124] A. Koepsel, J.-P. Ebert, and A. Wolisz, “A Performance Comparison of Point and Distributed Coordination Function of an IEEE 802.11 WLAN in the Presence of Real-Time Requirements,” in *7th Intl. Workshop on Mobile Multimedia Communications (MoMuC 2000)*, Tokyo, Japan, Oct. 2000.

[125] A. Koepsel and A. Wolisz, “Voice Transmission in an IEEE 802.11 WLAN Based Access Network,” in *WoWMoM 2001*, Rom, Italy, July 2001, pp. 24–33.

[126] P. Krishnan, A. Krishnakumar, W.-H. Ju, C. Mallows, and S. Gamt, “A system for LEASE: Location estimation assisted by stationary emitters for indoor RF wireless networks,” in *Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004)*, vol. 2, Hong Kong, Mar. 2004, pp. 1001–1011.

[127] J. Kumagai, “Speech recognition: talk to the machine,” *IEEE Spectrum*, vol. 39, no. 9, pp. 60–64, 2002.

[128] A. Kurtenbach and P. Wintz, “Quantizing for noisy channels,” *IEEE Transactions on Communications*, vol. 17, no. 2, pp. 291–302, 1969.

[129] N. Laoutaris and I. Stavrakakis, “Intrastream synchronization for continuous media streams: A survey of playout schedulers,” *IEEE Network Magazine*, vol. 16, no. 3, pp. 30–40, May 2002.

[130] J. Lepak and M. Crescimanno, “Speed of light measurement using ping,” American Physical Society - Meeting Abstracts, Apr. 2002, abstract B2.009+.

[131] X. Li, K. Pahlavan, and J. Beneat, “Performance of TOA estimation techniques in indoor multipath channels,” in *13th IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*, vol. 2, Sept. 2002, pp. 911–915.

[132] 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.

[133] Y. J. Liang, E. G. Steinbach, and B. Girod, “Real-time voice communication over the internet using packet path diversity,” in *Proceedings of the ninth ACM international conference on Multimedia (MULTIMEDIA ’01)*. New York, NY: ACM Press, 2001, pp. 431–440.

[134] A. Lindgren, A. Almquist, and O. Schelén, “Quality of service schemes for IEEE 802.11 wireless LANs - an evaluation,” *Mobile Networking and Applications (MONET)*, vol. 8, no. 3, pp. 223–235, June 2003.

[135] F. Liu, J. Kim, and C.-C. J. Kuo, “Adaptive delay concealment for internet voice applications with packet-based time-scale modification,” in *IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP ’01)*, vol. 3, May 2001, pp. 1461–1464.

[136] F. Liu, J. Kim, and C.-C. J. Kuo, “Quality enhancement of packet audio with time-scale modification,” in *Proc. SPIE ITCOM‘2002: Multimedia Systems and Applications*, Boston, MA, July 2002.

[137] C. Mahlo, C. Hoene, A. Rostami, and A. Wolisz, “Adaptive coding and packet rates for TCP-friendly VoIP flows,” in *Proc. of 3rd International Symposium on Telecommunications (IST2005)*, Shiraz, Iran, Sept. 2005.

[138] C. Mahlo, “Congestion control for low-rate interactive speech,” Master’s thesis, Technische Universität Berlin, Germany, Feb. 2003, adviser: C. Hoene.

[139] J. Malinen. (2004) Host ap driver for intersil prism2/2.5/3. [Online]. Available: http://hostap.epitest.fi/

[140] S. Mangold, S. Choi, P. May, O. Klein, G. Hiertz, and L. Stibor, “IEEE 802.11e wireless LAN for quality of service (invited paper),” in *European Wireless*, 2002.

[141] A. Markopoulou, F. Tobagi, and M. Karam, “Assessing the quality of voice communications over internet backbones,” *IEEE/ACM Transactions on Networking*, vol. 11, no. 5, pp. 747– 760, Oct. 2003.

[142] A. P. Markopoulou, “Assessing the quality of multimedia communications over internet backbones,” Ph.D. dissertation, Stanford University, USA, Oct. 2002.

[143] I. Marsh, F. Li, and G. Karlsson, “Wide area measurements of VoIP quality,” in *Quality of Future Internet Services (QOFIS).* Stockholm, Sweden: Springer, Oct. 2003.

[144] E. Masala and J. De Martin, “Analysis-by-synthesis distortion computation for rate-distortion optimized multimedia streaming,” in *International Conference on Multimedia and Expo (ICME ’03)*, vol. 3, 2003, pp. III–345–8.

[145] S. Mohamed, F. Cercantes-Perez, and H. Afifi, “Integrating networks measurements and speech quality subjective scores for control purposes,” in *Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2001)*, vol. 2, Anchorage, AK, Apr. 2001, pp. 641–649.

[146] S. B. Moon, J. Kurose, and D. Towsley, “Packet audio playout delay adjustments: performance bounds and algorithms,” *ACM/Springer Multimedia Systems*, vol. 27, no. 3, pp. 17–28, Jan. 1998.

[147] D. Moore, J. Leonard, D. Rus, and S. Teller, “Robust distributed network localization with noisy range measurements,” in *Second international conference on Embedded networked sensor systems.* ACM Press, 2004, pp. 50–61.

[148] F. Moss, “Stochastic resonance: a signal+noise in a two state system,” in *45th Annual Symposium on Frequency Control*, May 1991, pp. 649–658.

[149] S. Möller, *Assessment and Prediction of Speech Quality in Telecommunications.* Kluwer Academic Publishers, 2000.

[150] S. Möller and C. Hoene, “Information about a new method for deriving the transmission rating factor R from MOS in closed form,” ITU - Telecommunication Standardization Sector, May 2002, temporary document for the Study Group 12.

[151] S. Möller, "Contribution 37: The e-model: An analysis of the source and comparision with published and new test results," ITU-T Study Group 12, Dec. 1997.

[152] G. T. Nguyen, R. H. Katz, B. Noble, and M. Satyanarayanan, "A trace-based approach for modeling wireless channel behavior," in *28th conference on Winter simulation.* ACM Press, 1996, pp. 597–604.

[153] D. Niculescu and B. Nath, "Error characteristics of ad hoc positioning systems (aps)," in *Fifth ACM international symposium on Mobile ad hoc networking and computing (MobiHoc'04).* ACM Press, 2004, pp. 20–30.

[154] Nielson. (2005, Mar.) Nielson//netranking web page. [Online]. Available: http://www.netrankings.com

[155] P. Noll, "Digital audio coding for visual communications," *Proceedings of the IEEE*, vol. 83, no. 6, pp. 925–943, 1995.

[156] P. Noll, V. Leesemann, and G. Wessels, *Paketorientierte Sprachübertragung*, ser. Mitteilung 86-05. Deutsche Versuchsanstalt für Luft- und Raumfahrt (DFVLR), 1986.

[157] J. Ott and D. Kutscher, "Drive-thru internet: IEEE 802.11b for "automobile" users," in *Twenty-third AnnualJoint Conference of the IEEE Computer and Communications Societies (INFOCOM 2004)*, vol. 1, 2004.

[158] K. Pawlikowski, G. Ewing, and D. McNickle. (2003) Project akaroa. [Online]. Available: http://www.cosc.canterbury.ac.nz/research/RG/net_sim/simulation_group/akaroa/

[159] S. Pennock, "Accuracy of the perceptual evaluation of speech quality (PESQ) algorithm," in *Measurement of Speech and Audio Quality in Networks (MESAQIN 2002)*, Apr. 2002.

[160] C. Perkins, O. Hodson, and V. Hardman, "A survey of packet loss recovery techniques for streaming audio," *IEEE Network*, vol. 12, pp. 40–48, Sept. 1998.

[161] C. Perkins, I. Kouvelas, O. Hodson, V. Hardman, M. Handley, J. Bolot, A. Vega-Garcia, and S. Fosse-Parisis, "RTP payload for redundant audio data," RFC 2198, IETF Network Working Group, Sept. 1997.

[162] D. Petr, J. DaSilva, L.A., and V. Frost, "Priority discarding of speech in integrated packet networks," *IEEE Journal on Selected Areas in Communications*, vol. 7, no. 5, pp. 644–656, 1989.

[163] D. Petr and V. Frost, "Nested threshold cell discarding for ATM overload control: optimization under cell loss constraints," in *Tenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM '91)*, 1991, pp. 1403–1412.

[164] M. Petracca, A. Servetti, and J. C. De Martin, "Voice transmission over 802.11 wireless networks using analysis-by-synthesis packet classification," in *First International Symposium on Control, Communications and Signal Processing*, Hammamet, Tunesia, Mar. 2004, pp. 587–590.

[165] J. Pinto and K. J. Christensen, “An algorithm for playout of packet voice based on adaptive adjustment of talkspurt silence periods,” in *IEEE 24th Conference on Local Computer Networks (LCN)*, Lowell, MA, Oct. 1999, pp. 224–231.

[166] M. Podolsky, S. McCanne, and M. Vetterli, “Soft ARQ for layered streaming media,” University of California Berkeley, Computer Science Division, Berkeley, CA, Tech. Rep. UCB/CSD-98-1024, Nov. 1998.

[167] M. Podolsky, C. Romer, and S. McCanne, “Simulation of FEC-based error control for packet audio on the internet,” in *Seventeenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM ’98)*, vol. 2, San Francisco, CA, Mar. 1998, pp. 505–515.

[168] J. Postel, “User datagram protocol,” RFC 768, IETF Network Working Group, Aug. 1980.

[169] J. Postel, “Internet protocol,” RFC 761, IETF Network Working Group, Sept. 1981.

[170] (2005, Sept.) Precise location research at Intel. [Online]. Available: http://www.intel.com/research/precision_location.htm

[171] Psytechnics, “Delayed contribution 175: High level description of psytechnics ITU-T P.VTQ candidate,” ITU-T Study Group 12, Sept. 2003.

[172] R Development Core Team, *R: A Language and Environment for Statistical Computing*, R Foundation for Statistical Computing, Vienna, Austria, 2005, ISBN 3-900051-07-0. [Online]. Available: http://www.R-project.org

[173] R. Ramjee, J. F. Kurose, D. F. Towsley, and H. Schulzrinne, “Adaptive playout mechanisms for packetized audio applications in wide-area networks,” in *13th Proceedings IEEE INFOCOM ’94*, Toronto, Canada, June 1994, pp. 680–688.

[174] B. Rathke, T. Assimakopoulos, R. Morich, G. Schulte, and A. Wolisz, “SNUFFLE: integrated measurement and analysis tool for wireless internet and its use for in-house environment,” in *Tenth Intl. Conf. f. Computer Performance Evaluation, TOOLS’98*, Palma de Mallorca, Spain, Sept. 1998.

[175] D. R. Reddy, “Pitch period determination of speech sounds,” *Commun. ACM*, vol. 10, no. 6, pp. 343–348, 1967.

[176] A. W. Rix, M. P. Hollier, A. P. Hekstra, and J. G. Beerends, “Perceptual evaluation of speech quality (PESQ), the new ITU standard for end-to-end speech quality assessment, part I - time alignment,” *Journal of the Audio Engineering Society*, vol. 50, no. 10, p. 755, June 2002.

[177] M. Roder, J. Cardinal, and R. Hamzaoui, “On the complexity of rate-distortion optimal streaming of packetized media,” in *Data Compression Conference (DCC 2004)*, 2004, pp. 192–201.

[178] J. Rosenberg, “G.729 error recovery for internet telephony,” Columbia University Computer Science, Prof. H. Schulzrinne, New York, NY, Tech. Rep. CUCS-016-01, Dec. 2001.

[179] J. Rosenberg, L. Qiu, and H. Schulzrinne, “Integrating packet FEC into adaptive voice playout buffer algorithms on the internet,” in *Nineteenth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2000)*, vol. 3, Tel Aviv, Israel, Mar. 2000, pp. 1705–1714.

[180] J. Rosenberg and H. Schulzrinne, “An RTP payload format for generic forward error correction,” RFC 2377, IETF Network Working Group, Dec. 1999.

[181] J. Rosenberg, H. Schulzrinne, G. Camarillo, A. Johnston, J. Peterson, R. Sparks, M. Handley, and E. Schooler, “SIP: session initiation protocol,” RFC 3261, IETF Network Working Group, June 2002.

[182] M. Röder, J. Cardinal, and R. Hamzaoui, “Branch and bound algorithms for rate-distortion optimized media streaming,” *IEEE Trans. Multimedia*, 2005, in press.

[183] H. Sanneck, W. Mohr, L. Le, C. Hoene, and A. Wolisz, “Quality of service support for voice over IP over wireless,” in *Wireless IP and Building the Mobile Internet*, S. Dixit and R. Prasad, Eds. Norwood, MA: Artech House, Dec. 2002, ch. 10.

[184] H. Sanneck, N. Tuong, L. Le, A. Wolisz, and G. Carle, “Intra-flow loss recovery and control for VoIP,” in *Ninth ACM international conference on Multimedia (MULTIMEDIA ’01)*. New York, NY: ACM Press, 2001, pp. 441–454.

[185] C. Schmidmer, “3SQM - a breakthrough in single ended voice quality testing,” in *30th German Convention on Acoustics (DAGA)*, Strasbourg, France, Mar. 2004.

[186] H. Schulzrinne, “RTP profile for audio and video conferences with minimal control,” RFC 1890, IETF Network Working Group, Jan. 1996.

[187] H. Schulzrinne, S. Casner, R. Frederick, and V. Jacobson, “RTP: a transport protocol for real-time applications,” RFC 1889, IETF Network Working Group, Jan. 1996.

[188] A. Servetti and J. C. De Martin, “Adaptive interactive speech transmission over 802.11 wireless LANs,” in *Int. Workshop on DSP in Mobile and Vehicular Systems*, Nagoya, Japan, Apr. 2003.

[189] E. Setton, X. Zhu, and B. Girod, “Congestion-optimized multi-path streaming of video over ad hoc wireless networks,” in *IEEE International Conference on Multimedia and Expo, 2004. ICME ’04*, vol. 3, 2004, pp. 1619–1622.

[190] S. Sibal, K. Parthasarathy, and K. S. Vastola, “Sensing the state of voice sources to improve multiplexer performance,” in *INFOCOM ’95. Fourteenth Annual Joint Conference of the IEEE Computer and Communications Societies*, vol. 1, 1995, pp. 56–63.

[191] B. W. Silverman, *Density Estimation for Statistics and Data Analysis.* New York, NY: Chapman and Hall, 1986.

[192] J. Sjoberg, M. Westerlund, A. Lakaniemi, and Q. Xie, “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,” RFC 3267, IETF Network Working Group, June 2002.

[193] C. Sreenan, J.-C. Chen, P. Agrawal, and B. Narendran, "Delay reduction techniques for playout buffering," *IEEE Transactions on Multimedia*, vol. 2, no. 2, pp. 88–100, June 2000.

[194] K. Sriram and D. Lucantoni, "Traffic smoothing effects of bit dropping in a packet voicemultiplexer," *IEEE Transactions on Communications*, vol. 37, no. 7, pp. 703–712, 1989.

[195] R. Steinmetz, "Human perception of jitter and media synchronization," *IEEE Journal on Selected Areas in Communications*, vol. 14, no. 1, pp. 61–72, 1996.

[196] A. Stenger, K. B. Younes, R. Reng, and B. Girod, "A new error concealment technique for audio transmission with packet loss," in *European Signal Processing Conference (EUSIPCO 96)*, Trieste, Italy, Sept. 1996.

[197] W. R. Stevens, *TCP/IP Illustrated, Volume 1 - The Protocols*, 14th ed. Addison Wesley, 1999.

[198] L. Sun and E. Ifeachor, "New models for perceived voice quality prediction and their applications in playout buffer optimization for VoIP networks," in *IEEE International Conference on Communications (ICC 2004)*, Paris, France, June 2004, pp. 1478 – 1483.

[199] L. Sun and E. C. Ifeachor, "Prediction of perceived conversational speech quality and effects of playout buffer algorithms," in *IEEE International Conference on Communications (ICC 2003)*, Anchorage, USA, May 2003, pp. 1–6.

[200] L. Sun, G. Wade, B. Lines, and E. Ifeachor, "Impact of packet loss location on perceived speech quality," in *Second IP-Telephony Workshop (IPTEL '01)*, New York, NY, Apr. 2001, pp. 114–122.

[201] L. Sun, "Subjective and objective speech quality evaluation under bursty losses," in *Measurement of Speech and Audio Quality in Networks (MESAQIN 2002)*, Apr. 2002.

[202] A. Takahashi, H. Yoshino, and N. Kitawaki, "Perceptual QoS assessment technologies for VoIP," *IEEE Communications Magazine*, vol. 42, no. 7, pp. 28–34, June 2004.

[203] Telchemy, "Delayed contribution 105: Description of VQmon algorithm," ITU-T Study Group 12, Jan. 2003.

[204] (2003) The network simulator ns-2. [Online]. Available: http://www.isi.edu/nsnam/ns

[205] J. Tourrilhes, "Packet Frame Grouping: Improving IP Multimedia Performance over CSMA/CA," Hewlett Packard Laboratories, Bristol, UK, Tech. Rep. HPL-97-132, 1997.

[206] M. Veeraraghavan, N. Cocker, and T. Moors, "Support of voice services in IEEE 802.11 wireless LANs," in *Twentieth Annual Joint Conference of the IEEE Computer and Communications Societies (INFOCOM 2001)*, vol. 1, Los Alamitos, CA, 2001, pp. 488–497.

[207] H. Velayos and G. Karlsson, "Limitations in range estimation for wireless LAN," in *First Workshop on Positioning, Navigation and Communication (WPNC'04)*, Hannover, Germany, Mar. 2004.

[208] M. A. Visser and M. E. Zarki, "Voice and data transmission over an 802.11 wireless network," in *Sixth IEEE International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC'95)*, vol. 2, Toronto, Canada, Sept. 1995, pp. 648–652.

[209] J.-F. Wang, J.-C. Wang, J.-F. Yang, and J.-J. Wang, "A voicing-driven packet loss recovery algorithm foranalysis-by-synthesis predictive speech coders over internet," *IEEE Transactions on Multimedia*, vol. 3, no. 1, pp. 98–107, 2001.

[210] C. Weinstein and J. Forgie, "Experience with speech communication in packet networks," *IEEE Journal on Selected Areas in Communications*, vol. 1, no. 6, pp. 963–980, 1983.

[211] E. W. Weisstein. (2005, May) Number theory. from MathWorld–A Wolfram Web Resource. [Online]. Available: http://mathworld.wolfram.com/NumberTheory.html

[212] J. Werb and C. Lanzl, "Designing a positioning system for finding things and people indoors," *IEEE Spectrum*, vol. 35, no. 9, pp. 71–78, Sept. 1998.

[213] S. Wiethölter and C. Hoene, "Design and verification of an IEEE 802.11e EDCF simulation model in ns-2.26," Telecommunication Networks Group, Technische Universität Berlin, Tech. Rep. TKN-03-019, Nov. 2003.

[214] S. Wiethölter and C. Hoene. (2003, Nov.) An IEEE 802.11e EDCF and CFB simulation model for ns-2. [Online]. Available: http://www.tkn.tu-berlin.de/research/802.11e_ns2/

[215] S. Wiethölter, C. Hoene, and A. Wolisz, "Perceptual quality of internet telephony over IEEE 802.11e supporting enhanced dcf and contention free bursting," Telecommunication Networks Group, Technische Universität Berlin, Tech. Rep. TKN-04-011, May 2004.

[216] (2005) Wikipedia, the free encyclopedia. [Online]. Available: http://www.wikipedia.org

[217] A. Willig, M. Kubisch, C. Hoene, and A. Wolisz, "Measurements of a wireless link in an industrial environment using an IEEE 802.11-compliant physical layer," *IEEE Transactions on Industrial Electronics*, vol. 43, no. 6, pp. 1265–1282, Dec. 2002.

[218] T. Wimmer, "Developing a quality model to predict the importance of a VoIP packet," Studienarbeit, Technische Universität Berlin, Berlin, Germany, Mar. 2005.

[219] L. Wolf, C. Griwodz, and R. Steinmetz, "Multimedia communication," *Proceedings of the IEEE*, vol. 85, no. 12, pp. 1915–1933, 1997.

[220] X. Wu and H. Schulzrinne, "Location-based services in Internet telephony systems," in *IEEE Consumer Communications and Networking Conference*, Jan. 2005.

[221] W. Yang, "Enhanced modified bark spectral distortion (EMBSD): An objective speech quality measure based on audible distortion and cognition model," Ph.D. dissertation, Temple University, Philadelphia, PA, May 1999.

[222] N. Yin, S.-Q. Li, and T. E. Stern, "Congestion control for packet voice by selective packet discarding," *IEEE Transactions on Communications*, vol. 38, no. 5, pp. 674–683, May 1990.

[223] M. Youssef, A. Agrawala, and U. Shankar, “WLAN location determination via clustering and probability distributions,” in *Pervasive Computing and Communications, 2003. (PerCom 2003). Proceedings of the First IEEE International Conference on*, Mar. 2003, pp. 143–150.

[224] S. Zinal, “Beton,” email, Feb. 2004, TU-Berlin.

[225] M. Zorzi, R. Rao, and L. Milstein, “On the accuracy of a first-order markov model for data transmission on fading channels,” in *Fourth IEEE International Conference on Universal Personal Communications*, Nov. 1995, pp. 211–215.

[226] M. Zorzi and R. R. Rao, “Lateness probability of a retransmission scheme for error control on a two-state markov channel,” *IEEE Transactions on Communications*, vol. 47, no. 10, pp. 1537 – 1548, Oct. 1999.

[227] E. Zwicker and H. Fastl, *Psychoacoustics, facts and models.* Springer Verlag, 1990.