Robust Real-Time Audiotransmission over Lossy Channels

Christoph Günzel*

January 19, 1999

Abstract

We present a system for a robust transmission of audio signals over lossy channels. For protection against losses in the network during transmission we use a Forward Error Correcting (FEC) scheme called Erasure Resilient Transmission (ERT). ERT provides multiple levels of redundancy in order to protect the different contents of a data set according to their importance. The task of optimally assigning redundancies for the ERT encoding scheme is investigated for uniform quantized digital audio signals here.

*Dept. of Computer Science, University of Bonn, 53117 Bonn. Email: guenzel@cs.bonn.edu

1 Introduction

The number of computer applications which make use of digital audio files is continously on the rise. At the same time many of these applications are intended for usage in a heterogeneous networking environment which exhibits links with different bandwidths and workstations with varying computational power.

Erasure Resilient Transmission (ERT), based on [BKK+ 95] [G 96] [ABEL 94] is an approach for sending realtime applications over today's networking environments. A major goal of ERT is to provide graceful degradation for digital audio in order to allow users with different hardware equipment and connectivity to share the same application. Like for MPEG-1 video streams [GR 98] [GRW 97] [Le 94] ERT protects the different contents of digital audio signals according to their importance via a multilevel redundancy scheme and information spreading.

The intention is to increase the likelihood that the more important parts of the information can still be recovered if the audio signal is corrupted by packet losses during transit. These packet losses can be caused by different events like network congestion, fading in satellite transmission, insufficent computational power of the receiving workstations and so on.

Coding the audio signals in such a way that a potential degradation is perceived as graceful is a challenging task. First it must be clear, how the different quality reductions in an audio signal are perceived by the human ear. Therefore a standard objective measure of coded waveform quality is the signal-to-noise ratio (SNR), usually expressed in decibels (dB) [JN 84].

A thorough understanding of the audio prioritization as well as the ERT encoding process is then necessary to analyze what kind of changes lead to which effect, assuming that a certain packet loss behaviour is given.

In chapter 2 I give an introduction to digital audio and describe the partion of linear quantized audio signals in different priority levels. A possible implementation is also presented. In chapter 3 I describe and discuss the ERT encoding scheme. In chapter 4 I introduce the resulting ERT encoding scheme for linear quantized audio signals and analyze the changes of quality due to packet losses.

2 Digital Audio

The digital representation instead of analog representation of audio data offers many advantages such as high noise immunity, stability and reproducibility. Audio in digital form also allows the efficient implementation of many audio processing functions through the computer (e.g. mixing, filtering, prioritizing). The benefits of digital audio are many and well known [Aa 79].

The techniques, I want to present here, to prioritize digital audio apply to general audio signals and are not specially tuned for speech signals.

In this paper I will concentrate on Pulse Code Modulation (PCM). PCM is the best established, the most implemented and the most applied of all digital coding systems. Another reason for the importance of PCM is that it is not signal-specific; rather, it is versatile: for example, PCM is not mismatched to voiceband data waveforms. For this reason, PCM is wideley accepted as a standard.

PCM audio signals are also well known in the Intel/Microsoft world as .wav-files.

2.1 Pulse Code Modulation (PCM)

A PCM coder is nothing more then a waveform sampler followed an amplitude quantizer. The conversion from the analog to the digital domain begins by sampling the audio input in regular, discrete intervals of time and quantizing the sampled values into a discrete number of evenly spaced levels. This process is called linear or uniform quantizing. Thus the digital audio data consists of a sequence of binary values, representing the number of quantizer levels for each audio sample. Figure 1 shows the digital audio process.

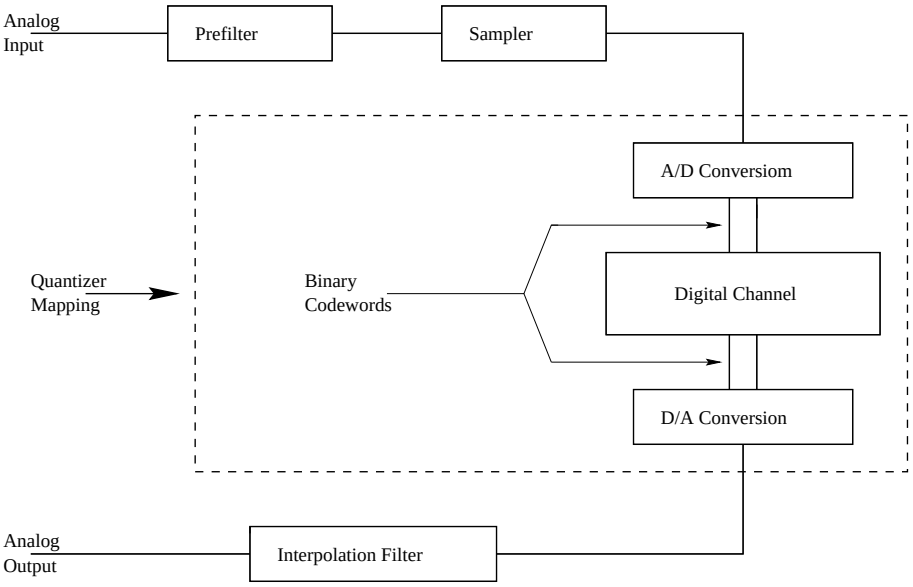

Figure 1: The digital audio process in PCM

According to the Nyquist theory, a time-sampled signal can faithfully represent signals up to half the sampling rate [OS 89]. Typical sampling rates range from 8 kilohertz (kHz) to 48 kHz. The 8-kHz rate covers a frequency range up to 4 kHz and so covers most of the frequencies produced by the human voice.The 48-kHz rate covers a frequency range up to 24 kHz and more than adequately covers the entire audible frequency range, which for humans typically extends to only 20 kHz. In practice, the frequency range is somewhat less than half the sampling rate because of the practical system limitations. The bandwidth of telephone speech is usually 3.2 or 3.4 kHz although the sampling rate is maintained at the standard value of 8 kHz. The 40-kHz rate for music is appropriate for studio quality 20 kHz bandwidth material. Lower bandwidths such as 15 or 7 kHz are sometimes employed in audio systems.

Waveform to Be Digitized	Signal Bandwith W (kHz)	Sampling Rate f_s (kHz)	PCM Bit Rate	
			R (bits/sample)	I (kb/s)
Telephone speech	4	8	8	64
Music	20	40	16	640

Figure 2: Examples of high-quality digitization with a PCM coder

The number of quantizer levels in a PCM system is typically a power of 2 to make full use of a fixed number of bits per audio sample to represent the quantized values.

2.2 Prioritizing of uniform quantized PCM audio data

Figure 2 shows the digitization of two very important applications with a PCM coder, telephone speech and music. W is the input bandwith in kHz of the signal to be digitized. The sampling rate f_s of the PCM system holds the equation

$$f_s >> 2W$$

as we can see in figure 2.

$$I = f_s \cdot R$$

kilobits per second (kb/s) is the transmission rate of the system. As written above, the PCM system uses a single 2^R-level quantizer for discreting all amplitude samples. Here $R = \log_2 2^R$ is simply the number of bits used to represent each sample.

Let us assume that the audio signal is given as an amplitude-versus-time plot (Figure 3), uniform quantized with R = 16.

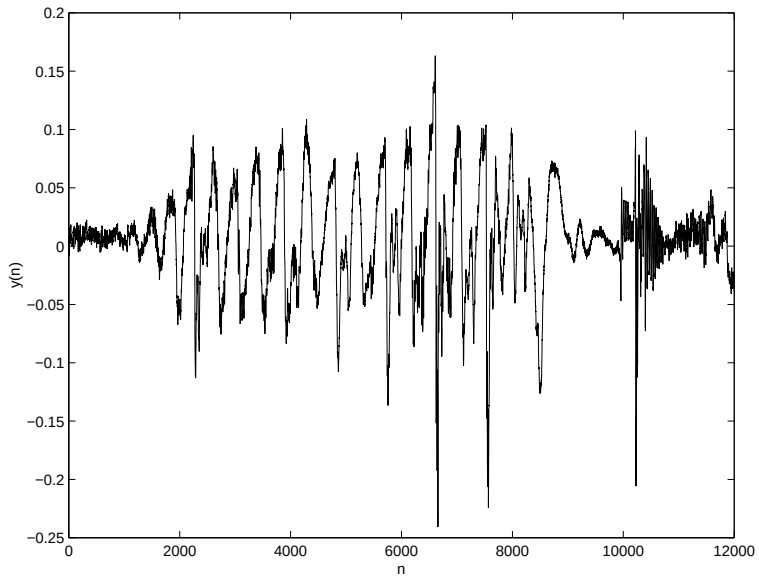

Figure 3: typical amplitude-versus-time plot of an input signal

For a prioritization of the data we have to decide which parts of the data are absolutely necessary for understanding the submitted message which parts are less necessary and only improve the quality of the signal. Furthermore we have to watch the quality changes of the audio signal due to packet loss to get a graceful degradation of the audio quality. The idea is to weight single bits of each sample to built levels of different importance. The construction of two priority levels will be described in the following example: Let us assume we want to devide the 16 bits per sample in two priority classes P_1 and P_2. P_1 should be more important than P_2. The algorithm works as follows:

1. We convert the 16 bit samples into 8 bit samples. Therefore we have to map an amplitude range of -32767 to 32768 to an range of -127 to 128. Of course this can not be done without any errors but a very useful method where we can easily correct the errors is to devide the 16 bit values by 256. This is equivalent to a logical AND of the 16 bit words (the samples) with 65280. In this case we split the 16 bit words in two 8 bit words, one containing the 8 'higher' bits, that's the corresponding 8 bit sample of the original 16 bit sample, the other the 8 'lower' bits.
2. We send the 8 bit samples containing the 'higher' bits as priority class P_1 the other as P_2 to the receiver.
3. There are two possible situations at the receiver:
   (a) Both P_1 and P_2 arrive. In this case we can very easily rebuild the original 16 bit sample. We simply concatenate the two 8 bit words.
   (b) Only priority P_1 arrives. Now we have two possibilities: Either we can switch our soundcard from 16 bit to 8 bit resolution which can not be done without noise and play the 8 bit sample or we convert the 8 bit sample back to a 16 bit sample.

Figure 4 shows an example which illustrates the three steps described above.

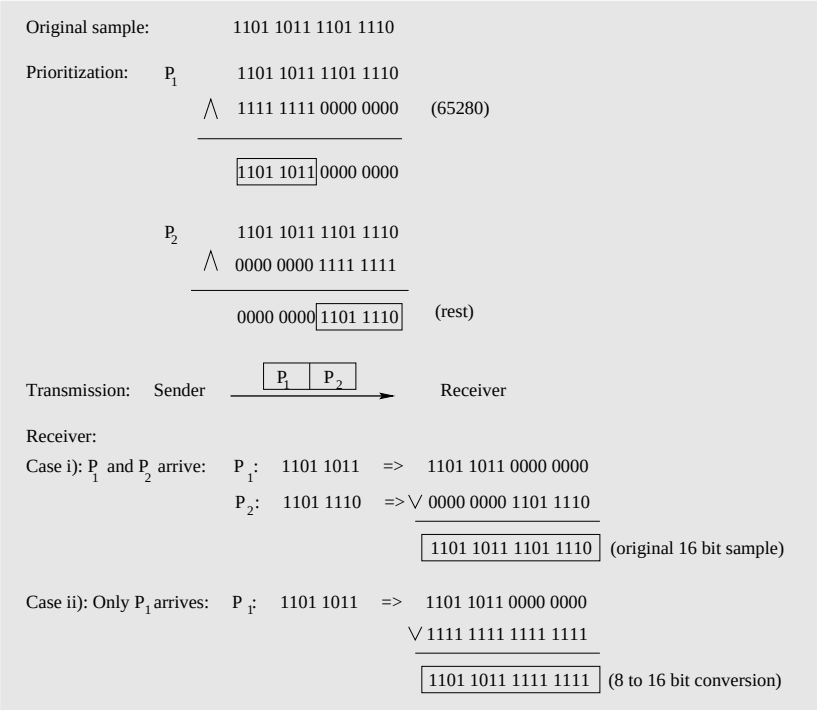

Figure 4: Illustration of the algorithm

With this method we can convert the 16 bit samples to every resolution between 0 and 16 bit with different loss of quality. Very easily we can build more than two priority levels.

2.3 Quality of audio signals

A standard objective measure of coded waveform quality is the signal-to-noise ratio (SNR), usually expressed in decibels (dB):

$$SNR(dB) = 10 \log_{10}(\sigma_x^2/\sigma_r^2)$$

with

$x(n)$ is the input to the waveform coding system
$y(n)$ is the output of the waveform coding system, the digital format
$r(n)$ is the reconstruction error, defined as $r(n) = x(n) - y(n)$
σ_x^2 is a common measure for the input signal level
σ_r^2 is the reconstruction error variance

For PCM systems with uniform quantizer step spacing, each additional bit has the potential of increasing the SNR, or equivalently the dynamic range, of the quantized amplitude by roughly 6 dB [JN 84] [Pa 93]:

$$\sigma_r^2 = a2^{-2R}\sigma_x^2$$
$$SNR(dB) = 6R - 10 \log_{10} a$$
a is a constant in the order of 1 to 10.

The typical number of bits per sample used for digital audio ranges from 8 to 16. The dynamic range capability of these representations thus ranges from 48 to 96 dB, respectively. To put these ranges into perspective, if 0 dB represents the weakest audio sound pressure level, then 25 dB is the minimum noise level in a typical recording studio, 35 dB is the noise level inside a quiet home, and 120 dB is the loudest level before discomfort begins.

2.4 Priority levels for PCM audio data

In this section I want to introduce usable priority levels for uniform quantized PCM audio data. As discussed above, priority P_1 should be the highest priority level. In case of errors or losses in lower priority levels P_2,P_3, ..., the quality of the audio signal recovered from P_1 has to guarantee a certain basic quality to satisfy most of the auditors, listening to the transferred audio signal. Thus, the quality of the down quantized samples generated by the conversion has to be acceptable for the majority of the auditors. As described the typical number of bits per sample used for digital audio ranges from 8 to 16. Therefore it makes no sense to quantize the 16 bit data to less than 8 bit. P_1 will content the quantized values with $R = 8$. That will be the 'higher' 8 bits produced by our algorithm. This will guarantee an accepted basic quality of the audio signal. Only a small digital noise is audible. As we know each additional bit, that we can recover will increase the SNR by 6 dB. In common opinion, high quality PCM coding of audio material requires at least 10 to 12 bit quantization. Thus priority level P_2 contents the 4 bits with numbers 8 to 11 of the original 16 bit sample. If P_1 and P_2 are received, the user will get high quality audio as accepted by common opinion. P_3 will content the remaining 4 bits with numbers 12 to 15 of each sample. Figure 5 illustrates the resulting priorization of an audio sample.

For speech transmission, it is possible to devide the signal in priority levels demonstrated in Figure 6.

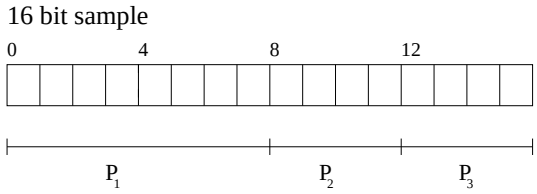

Figure 5: Prioritization of a 16 bit audio sample

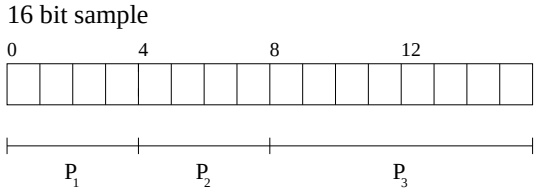

Figure 6: Prioritization of a 16 bit speech sample

The basic quality guaranteed by P_1 is even better than most signals in mobile communication.

As you can see, we can easily build more then three priority levels.

Figure 7 demonstrates PCM performance with an audio input sampled at $f_s = 44.1$ kHz and quantized with $R = 16$. Figure 8 shows the effects when the audio input is quantized down to $R = 8$. Figure 9 illustrates a sample quantized with $R = 4$ (speech signal).

3 Erasure Resilient Transmission (ERT)

From chapter 2 it has become clear that due to quality aspects of the audio signal the bits per sample differ in their importance. The main idea in ERT is to assign a certain amount of redundancy to the bits with the redundancy being unevenly distributed among the different bits. ERT is at the moment implemented based on encoding of information via Cauchy matrices [BKK$^+$ 95] [G 96].

3.1 A simplified view of ERT

Only those issues of the ERT encoding scheme will be described here that are crucial for the decision process of optimally assigning priorities. The basic ideas of ERT can be explained with the aid of Figure 10. It can be seen that a set of samples, which constitutes the message to be encoded, is encoded into n packets of a certain size. The mapping of the set of samples onto the n packets is done in such a way that information from every priority level (Chapter 2) is contained in each of the packets. As a consequence the information is spread among the n packets which renders improved robustness in the presence of bursty errors which are common in today's networking environment. The idea of information spreading has a long tradition and has been used extensively. The second idea in ERT is to provide error correcting properties on a multilevel basis.Therefore the most important bits of the samples (in our example in chapter 2 the 8 higher bits) are endowed with relatively more redundancy

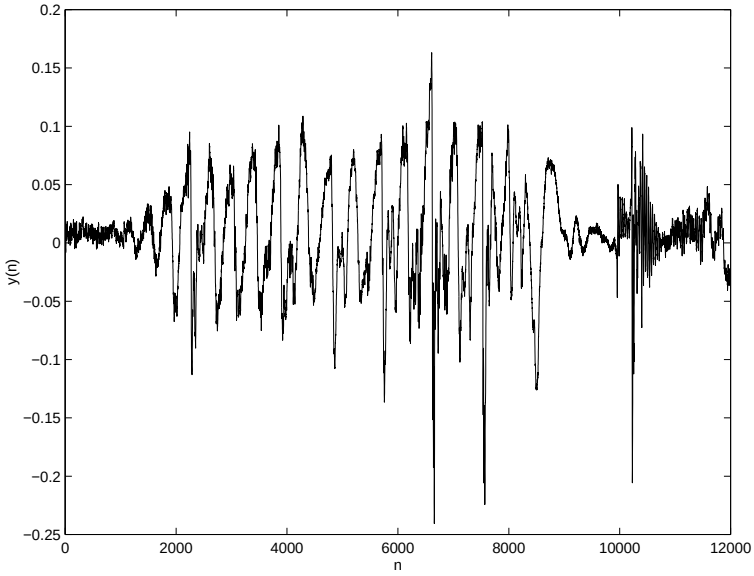

Figure 7: Audio input quantized with $R = 16$

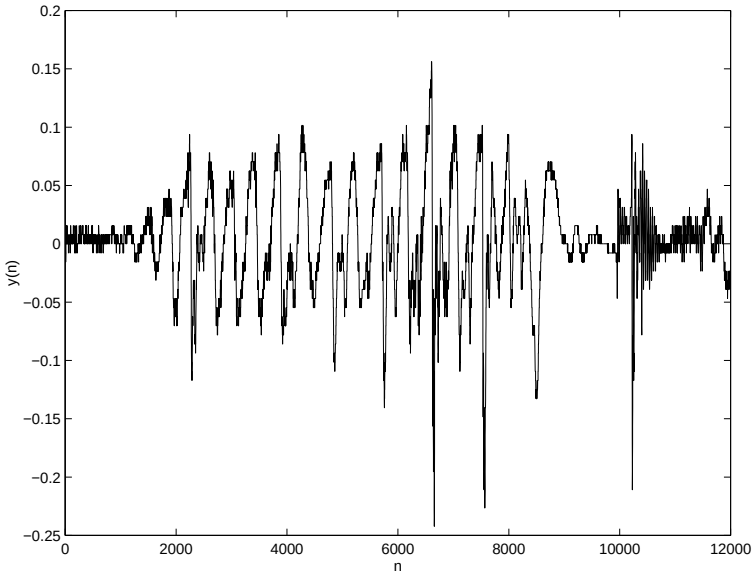

Figure 8: Audio input quantized with $R = 8$

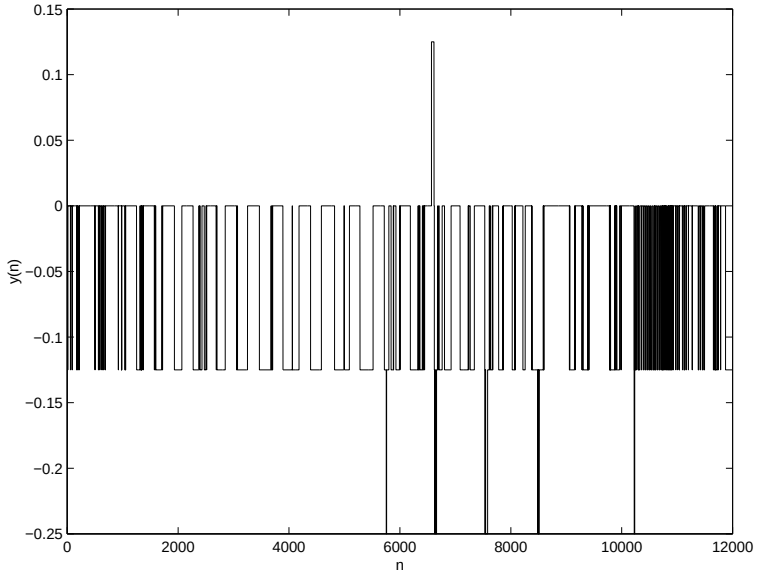

Figure 9: Audio input quantized with $R = 4$

information than the less important bits. A nice property of ERT is the fact that no decoding is required at the receiver if the cleartext information arrives undisturbed. This feature allows for fast processing in case of error free environments.

In case of errors the amount of redundancy being assigned to the different frame types decides wether a specific priority level can be recovered or not. If enough error free packets arrive, all of the bits belonging to a certain priority level can be recovered. If this threshold is not reached, no recovery is possible via decoding. Nevertheless, some cleartext information might have gotten through so that there is still a chance that some usable information has arrived, even though the recovery mechanism doesn't have enough packets to recover the whole message.

3.2 Analysis of the ERT scheme

The simplified ERT scheme will be analyzed more deeply in the following. Consider the following definitions:

N_i = size in kBytes of the elements of priority level i.

N = size in kBytes off all n ERT-packets (encoded).

n_i = number of packets required to reconstruct the elements of priority level i.

$x_i = \frac{n_i}{n}$, where x_i represents the fraction of packets that are necessary to recover all elements of priority level i. The x_i will also be reffered to as the ERT parameters which shall be chosen optimally to get the best possible audio quality.

With these definitions one can easily see that

$$N = \sum_i \frac{N_i}{x_i}$$

$$N = (\sum_i N_i) \cdot (1 + r)$$

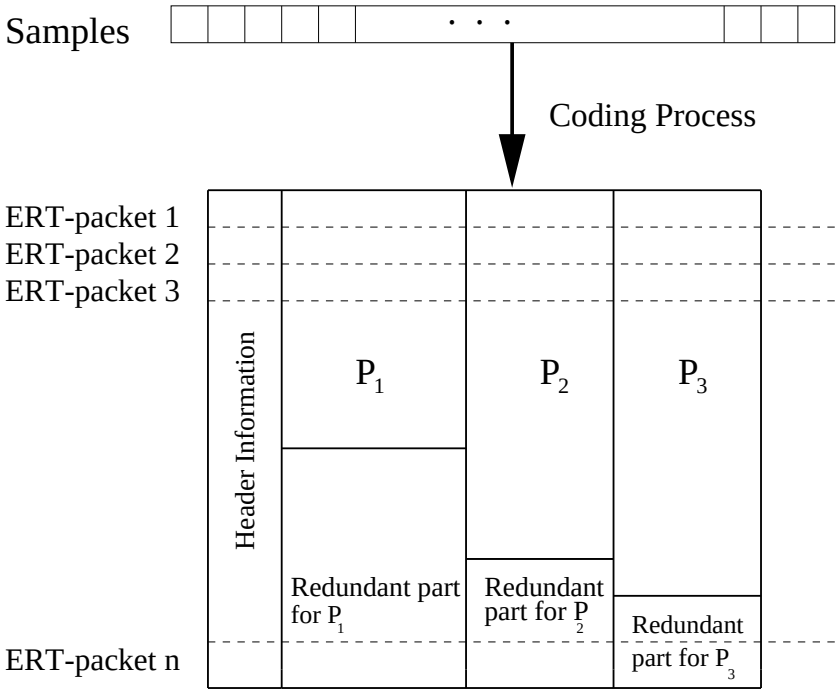

Figure 10: The coding process in ERT

r denotes the overall redundancy factor that is spend in the encoding process. The x_i are dependent on r.

One of ERT's basic design principles is that the x_i are specified by the user or the application. In addition to the redundancy which is going to be spent for the encoding process the probability of loosing packets during transmission also has influence on the choice of the parameters x_i. This probability is, among other things, dependent on the type of network, the load of the network, the packet size, number of packets to be transmitted, environmental conditions, the burstiness of the traffic and correlations. It's not so easy to compute packet loss statistics for different connections. Let us assume that the packet loss statistics belonging to a certain connection is available for our application (Figure 11).

4 An ERT scheme for PCM audio data

Figure 12 shows a possible redundancy distribution for an audio input sampled at $f_s = 44.100$ kHz, $R = 16$. For real time applications, a typical group of samples contains 882 samples, representing 20 ms.

Priorities are expressed by fraction of packets needed to recover the original information. P_1 might be encoded in a way via the ERT encoding scheme that it can be recovered from any 60% of the total number of packets sent, P_2 from any 80% and P_3 from any 95% of the packets. Figure 13 displays this multilevel encoding idea. As we can see, the total overhead is unevenly distributed over the data. Although the original message is 75% the length of the total encoding, 60%, 80% and 95% of the encoding is sufficient to recover P_1, P_2, P_3.

Let us recall, that x_i is the fraction out of n packets required to reconstruct P_i. If we

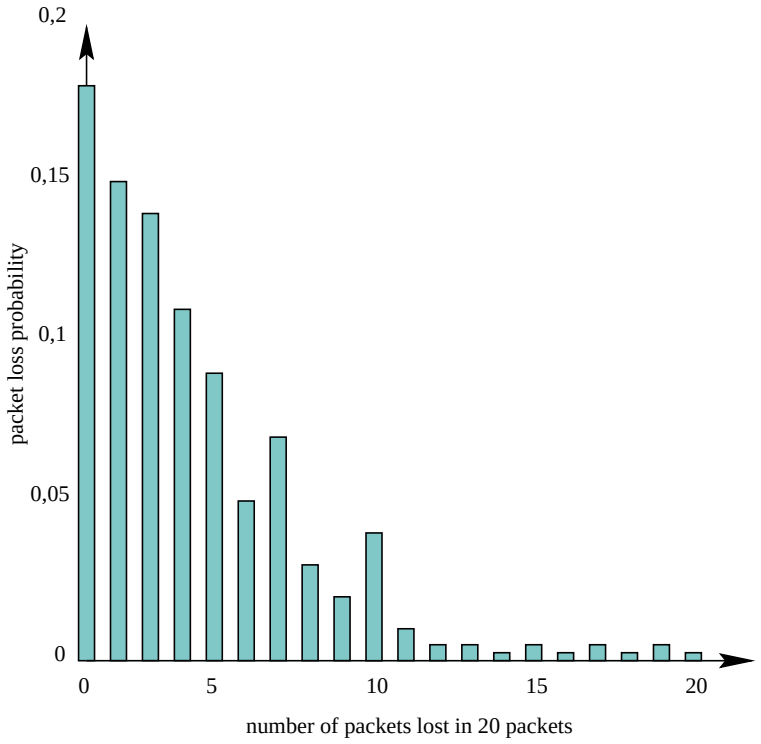

Figure 11: Packet loss probability

apply the redundancy r to the audio stream in a uniform manner we can see that there is only one value $x = x_1 = x_2 = x_3 = \frac{1}{1+r}$ which determines the recovery threshold. If x is the required fraction of packets to be received, $1-x$ is the fraction which is allowed to be lost. As illustrated in figure 14, all information will be recoverable if not more than $n \cdot (1-x) = \frac{n \cdot r}{1+r}$ packets are lost.A packet loss less than this number will result in complete recovery and good audio quality (16 bit resolution). If more than $\frac{n \cdot r}{1+r}$ packets are lost, no recovery is possible and thus the audio quality will be unacceptable. The results of a multilevel ERT encoding scheme are qualitatively explained in figure 15. If more than $n \cdot (1-x_3)$ packets are lost, P_3 is not recoverable any more. P_1 and P_2 are still recoverably, thus the audio quality drops to 12 bit resolution. The ERT encoding scheme generates a transition band between 16 bit and 8 bit resolution with less chance of an unacceptable resolution (< 8 bit). A disadvantage is that the threshold were the 16 bit solution starts is lower compared to the single level case. An

Priority level	Size in Bytes	Occurrence in group of samples	Total Size	Fraction x_i needed to recover	Encoding
P_1	8	882	7056	60%	11760
P_2	4	882	3528	80%	4410
P_3	4	882	3528	95%	3714
Total			14112		19884

Figure 12: Possible redundancy distribution

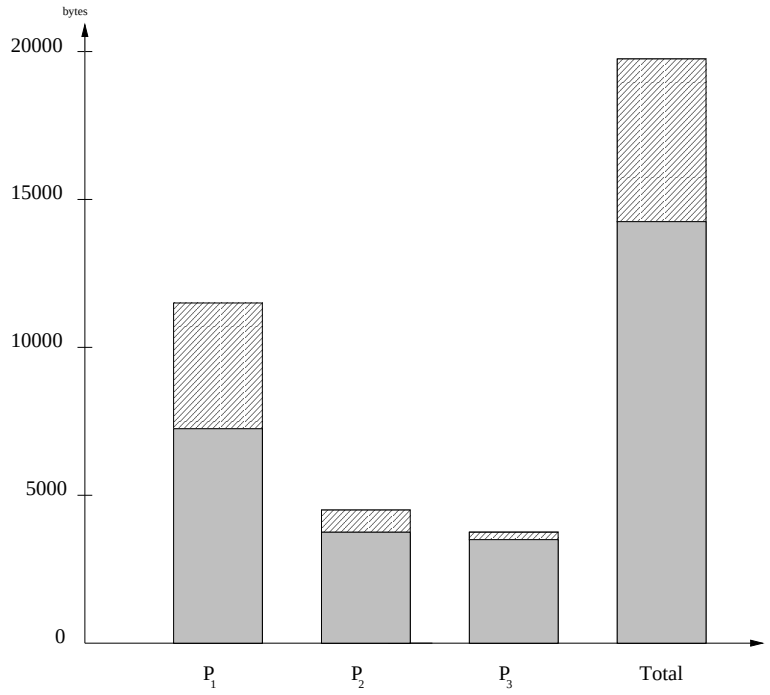

Figure 13: Multilevel redundancy distribution

important advantage of the multilevel ERT encoding scheme is that the redundancy added in the encoding process will be smaller than in the single level case and thus the bandwidth needed to transmit the encoded message will be smaller.

A typical audio signal which is recovered by the receiver is shown in figure 16. We can easily see where losses affected the signal during transit and thus the changing between 16 bit and 4 bit resolution.

5 Conclusion and further research

In this work I have presented a useful method to transmit uniform quantized PCM audio data to different users across a lossy network. I have shown how to prioritize data in a way that a graceful degradation of quality is possible in case of packet losses caused by the networking environment and how the quality of the audio signal increases with each additional bit received by the user. The SNR increases by 6 dB. Furthermore I have introduced the ERT encoding scheme which is very useful for sending messages over lossy media and have shown how to use it for the audio application. I have discribed the advantage of a multilevel redundancy scheme for real time applications and thus it makes sense to prioritize data. At the moment I run experiments on nonuniform quantized audio data. I will try to use the results on uniform quantized PCM data for nonuniform quantized data, especially for logarithmic quantized data. In speech coding, logarithmic quantizers are used, for example Alaw or ulaw.

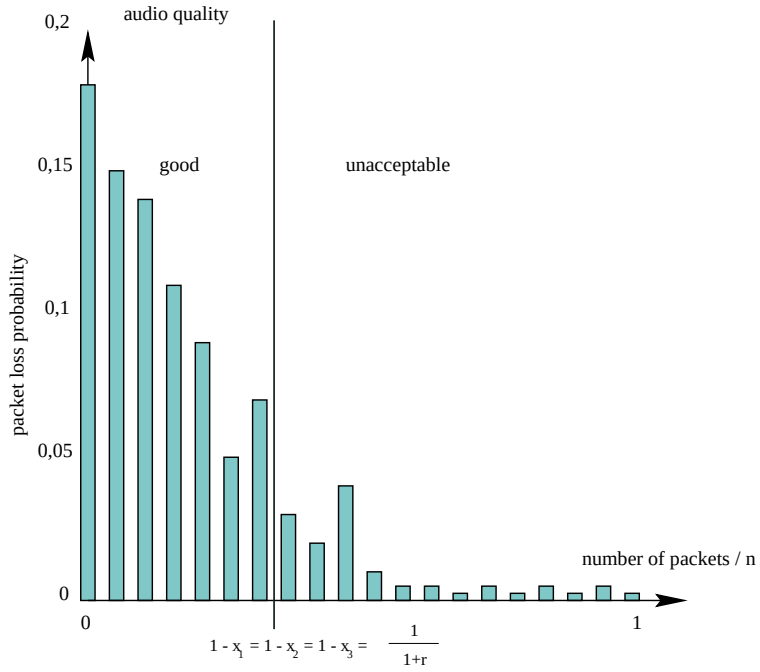

Figure 14: ERT with singlelevel redundancy

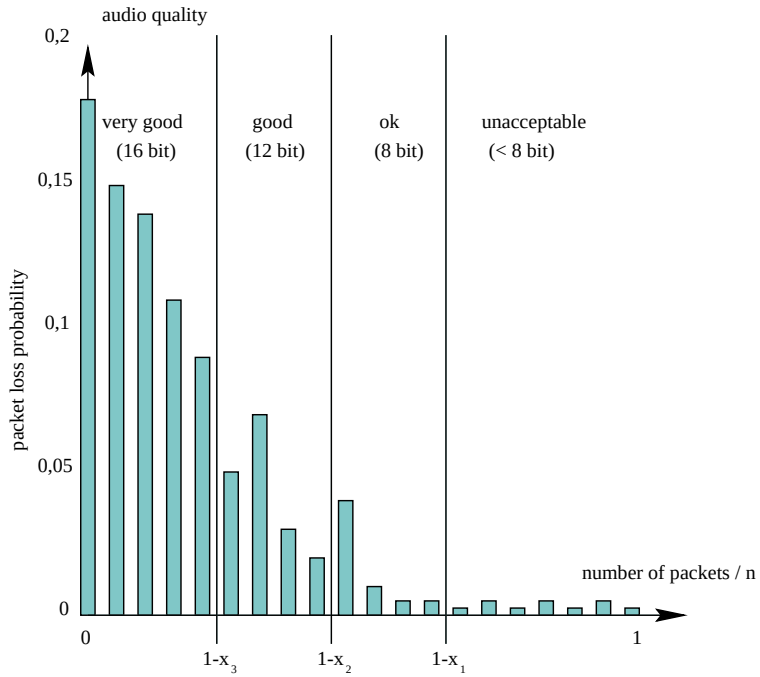

Figure 15: ERT with multilevel redundancy

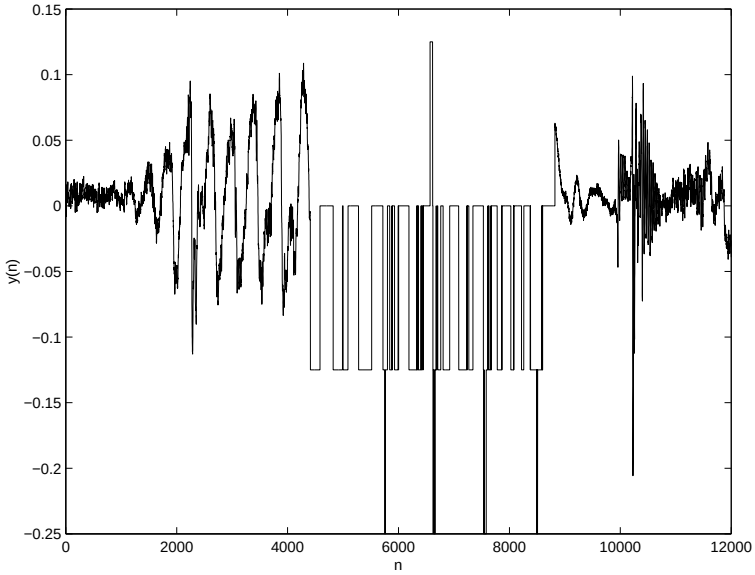

Figure 16: Audio signal corrupted by packet losses

Acknowledgements

I would like to thank Heiko Wagner for his experiments on quantizing .wav files and testing the presented algorithm 2. I also thank Frank Kurth for helpful comments and interesting discussions.

References

[Aa 79] Aaron, M., *The Digital (R)Evolution*, Proc. IEEE Communications Magazine (1979), pp. 21–22.

[ABEL 94] Albanese, A., Blömer, J., Edmonds, J., Luby, M., *Priority Encoding Transmission*, Technical Report TR-94-039, International Computer Science Institute, Berkeley, 1994.

[BKK+ 95] Blömer, J., Kalfane, M., Karp, R., Karpinski, M., Luby, M., Zuckerman, D., *An XOR-Based Erasure-Resilient Coding Scheme*, Technical Report TR-95-048, International Computer Science Institute, Berkeley, 1995.

[G 96] Günzel, C., *Fehlerresistente Übertragungssysteme für multimediale Anwendungen*, Diplomarbeit, Institut für Informatik der Universität Bonn, 1996.

[GR 98] Günzel, C., Riemenschneider, F., *Robust Real-Time Videotransmission over Lossy Channels*, Research Report 85206-CS, Institut für Informatik der Universität Bonn, 1998.

[GRW 97] Günzel, C., Riemenschneider, F., Wirtgen, J., *Parallel Real-Time Videotransmission over Lossy Channels*, Proc. 1^{st} Workshop on Cluster - Computing (1997), pp. 231–237.

[JN 84] Jayant, N., Noll, P., *Digital Coding of Waveforms, Principles and Applications to Speech and Video*, Prentice-Hall Inc., New Jersey, 1984.

[Le 94] Leicher, C., *Hierarchical Encoding of MPEG Sequences Using Priority Encoding Transmission (PET)*, Technical Report TR-94-058, International Computer Science Institute, Berkeley, 1994.

[OS 89] Oppenheimer, A., Schafer, R., *Discrete Time Signal Processing*, Prentice-Hall Inc., New Jersey, 1989.

[Pa 93] Pan, D., *Digital Audio Compression*, Proc. 5^{th} Digital Technical Journal (1993), pp. 1–14.