ZERO-REDUNDANCY ERROR PROTECTION FOR CELP SPEECH CODECS

Norbert Görtz
Institute for Network and System Theory
University of Kiel, Germany
Tel.: +49 431 77572 406, Fax: +49 431 77572 403
E-Mail: ng@techfak.uni-kiel.de

ABSTRACT

In this paper the possibilities of channel-error protection for transmission of CELP-coded speech over highly disturbed channels without additional bits for error-control are discussed. Algorithms are given which do not require explicit channel models and work without additional delay and almost no additional complexity. Time-based and mutual dependencies of the speech codec parameters are exploited for channel-error detection and parameter extrapolation at the decoder. The algorithms are optimized by informal listening tests rather than by maximization of a mathematically tractable measure.

1. INTRODUCTION

In the implementations of presently used speech codecs, i.e. the full-rate GSM-speech-codec, the extrapolation of parameters that were corrupted by bit-errors in the corresponding channel indices, is based on the correlation in time of the parameters. Often the extrapolation is implemented less selectively as bad-*frame*-handling instead of bad-*parameter*-handling, i.e. if an error has been detected the whole frame is replaced by the last uncorrupted set of indices. The new zero-redundant error detection techniques described in this paper make use of parameter dependencies that have partly not been reported yet and allow for a detection of single corrupted parameters. This way, uncorrupted parameters within a corrupted frame are not "thrown away" by some frame-replacing technique. The algorithms stated here are well suited for situations with strongly time-varying channels, since no explicit information about the channel is required, and for the improvement of existing codec implementations, because modifications are only required at the speech decoder.

The paper is organized as follows: In section 2 a codec developed for enhanced speech transmission in the GSM-system [1] is briefly described. It is used as reference codec for the following investigations. After that, time-based and mutual dependencies of its parameters are stated in section 3 and they are exploited for parameter extrapolation and channel-error detection at the decoder in sections 4 and 5. The algorithms are optimized by informal listening-tests rather than by maximization of a mathematically tractable measure because usual measures like SNR do not reflect the subjective speech quality well. Finally the performance of the proposed algorithms is discussed in section 6.

2. SPEECH CODEC

The CELP encoder processes frames of 160 samples, which are divided into 4 subframes. Figure 1 shows the block diagram of the speech encoder. The total number of bits per frame is 214, resulting in a bit rate of 10.7kbps.

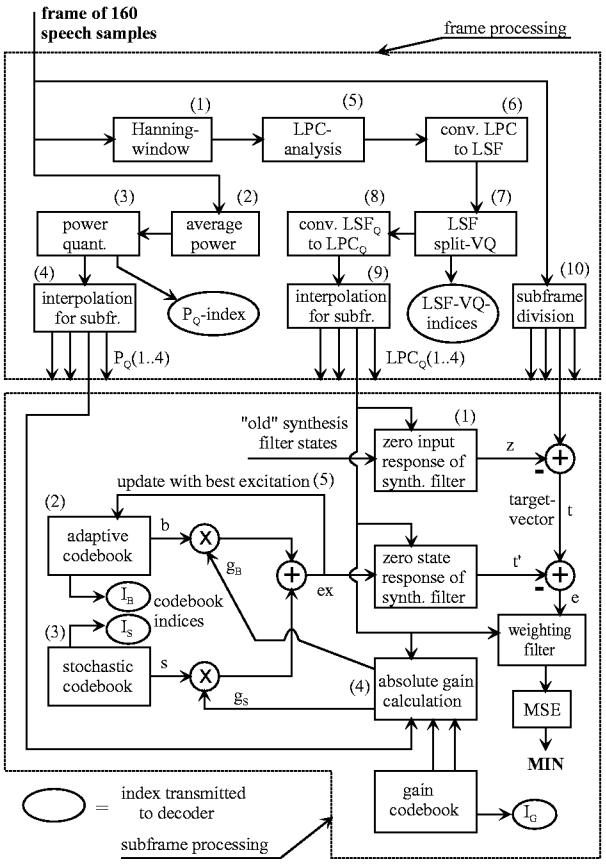

Figure 1: Speech Encoder

2.1. Frame Processing

First, the average power of the signal is calculated. It is logarithmically quantized by a 5-bit table, similar to [2]. Then a 10^{th}-order LPC analysis is carried out. The LPC coefficients are converted to Line Spectrum Frequencies (LSF) using the algorithm in [3]. They are split-vector-quantized according to the method in [4], resulting in 3 codebook indices for the 10 LSF, with 9+8+8=25 bits.

2.2. Subframe Processing

The search for the best components of the excitation vector "ex", that is the input signal of the synthesis filter which uses the LPC-coefficients from frame processing, and the corresponding gains is performed sequentially with the adaptive-excitation vector b first. While

searching for the best vectors b and s out of the codebooks the optimal scaling factors are used. When the best adaptive- and stochastic-excitation vectors have been found, the best gain codevector for those excitation vectors is searched "closed-loop" in the gain codebook.

The adaptive codebook is a buffer of the "best" excitation vectors of previous subframes. The idea is to find a past excitation vector that is similar to the one needed in the current subframe to synthesize the target vector t. The number of samples back in the past where the adaptive excitation starts is called "lag". The codebook is searched "closed-loop" over a lag-range of 20..141. The best lag is coded by 8 bits. Up to 5 non-integer values (fractional lags) between two integer samples are possible.

The stochastic excitation consists of ten +1/-1 pulses which are systematically placed in the excitation vector and coded by 30 bits, similar to the ACELP approach [5].

The excitation signals are scaled by two gains which are jointly quantized by an 8 bit codebook trained by the LBG-algorithm [6] similar to the method in [2]. The components of the codebook are P0, the power of the adaptive excitation divided by the sum of the powers of the excitation signals, and the factor GS, which compensates for the error in the estimation of the excitation signal power by the sum of the powers of the components, neglecting their correlation.

2.3. Decoder and Postfiltering

As in most "analysis-by-synthesis" codecs, the operations to be performed in the decoder (except post-processing) are similar to those already performed in the corresponding encoder stages. The postfilter is employed to increase the speech quality in terms of human perception. It includes long-term and short-term filtering similar to [7].

3. PARAMETER-DEPENDENCIES

A large set of speech data (100000 frames) was coded, and the quantized parameters of the speech codec, i.e. the average power, the LSF-coefficients, the lags, and the gains, were used to calculate the probability distributions of the differences of consecutive parameters with a distance of D frames/subframes. Some results are plotted in figure 2. Figure 2 reveals that the log. average power logP, the

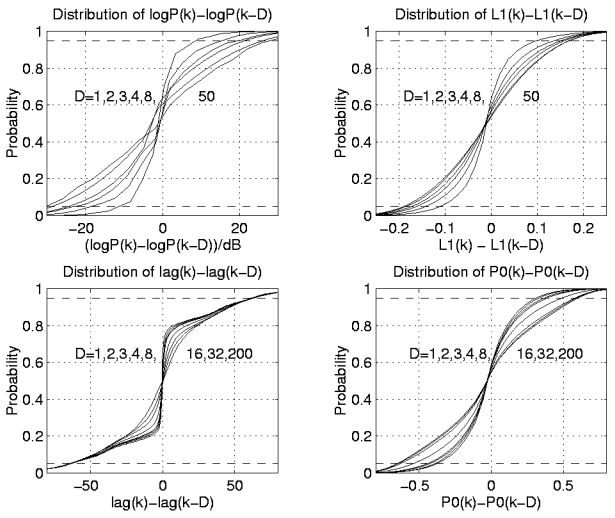

Figure 2: Probability distributions of differences of the parameters with a distance of D frames/subframes

LSF-coefficients (only the first LSF-coefficient is plotted), the lag and the normalized power P0 of the adaptive excitation have a considerable correlation in time, i.e. the differences of consecutive parameters tend to be smaller if their distance D is small. For $D > 4$ frames/16 subframes (80msec) the parameters are almost independent, only the lag has some correlation that is further discussed below. The reason for the independence can be derived from a time-domain plot of a speech signal: Most of the phonemes, that are the "areas of similarity" of the signal, don't last longer than 80ms, so there can be a correlation in time only for those parameters that describe properties of the speaker.

Beyond the correlations in time, the differences of consecutive lags are correlated with P0: If P0 is close to 1, the speech signal is often voiced and the lag-differences are small with higher probability than given by figure 2. In addition to that, the lag values in voiced subframes correspond to the basic pitch period or integer multiples or parts of it. Therefore the differences of consecutive lags are even more limited if lag-values that are corrected to the basic pitch period are considered. Figure 3 shows the plots of the probability distributions of differences of the lags for several intervals of P0 and with period correction in voiced subframes. In each plot, the curve labeled with "I" corresponds to the case of independent lag values in the difference. In voiced subframes the lag-value keeps some distance to the independent case even for distances of 200 subframes (1sec) because the speaker, i.e. the range of pitch frequencies, remains the same in the speech samples of the data bank for $3sec = 600$ subframes or more. There is quite a strong difference between the plots in fig-

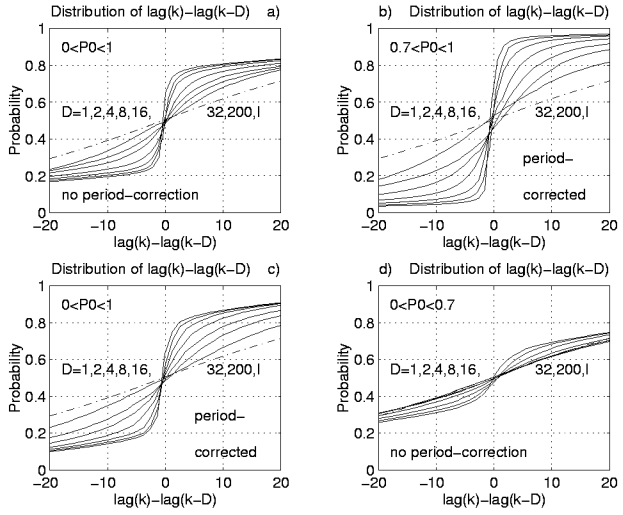

Figure 3: Probability distributions of differences of lags with a distance of D subframes for several ranges of P0

ure 3 b) and d), i.e. the lag values are strongly correlated in voiced subframes and they are almost independent in unvoiced subframes.

Also the LSF-coefficients *within* a frame are correlated: The differences of LSFs with neighbouring indices only have one sign and their absolute values are limited by the signal statistics.

The parameter dependencies can be exploited for parameter-extrapolation and error detection if bit errors occur on the channel. In the following, parameter-extrapolation is treated first because it is necessary for

the optimization of the zero-redundant error detections in section 5.

4. BAD-PARAMETER HANDLING

If a parameter corruption has (somehow) been detected, bad-parameter handling is carried out in the speech decoder by replacing the corrupted parameter of the current frame by the last uncorrupted parameter from a previous frame. This basic idea is justified by the correlations in time measured in section 3. For situations with consecutive parameter-corruptions, the last uncorrupted parameters are modified before use: The power, for example, is decreased with the number of consecutive corrupted indices to muffle the extrapolated signal and thereby to avoid annoying distortions in the output signal. The LPC-poles are radially shifted towards the origin of the Z-plane, so peaks in the spectrum will more and more be flattened if several errors occur in a sequence . This is implemented by

$$a_{n,D} = a_{n,old}\,\gamma^{n(D-1)},\ n = 1, 2, .., p \qquad (1)$$

with LPC-order p. D is the distance of the corrupted set of LPC-coefficients to the last uncorrupted set $a_{n,old}$. The factor $\gamma = 0.98$ was found by informal listening-tests and $a_{n,D}$ are the extrapolated LPC-coefficients for the frame.

In the subframes, the extrapolation of corrupted parameters can be performed by exploiting uncorrupted parameters in future subframes within the frame, without adding additional delay to decoding. For the gains, the following formulas are used for linear interpolation between the last uncorrupted and the future gain-parameter within the current frame:

$$GS_{log}(m) = \frac{1}{D+q-m}\{(q-m)\ GS_{log,old} + D\ GS_{log}(q)\} \qquad (2)$$

and

$$P0(m) = \frac{1}{D+q-m}\{(q-m)\ P0_{old} + D\ P0(q)\} \qquad (3)$$

In the formulas, D is the subframe distance between the corrupted and the last uncorrupted parameter, the parameters indexed "old" are the last uncorrupted ones, q is the subframe index of the future subframe with the uncorrupted parameter, and m is the current subframe-index with the corrupted parameter, i.e. $q > m$ and $q, m = 1, 2, 3, 4$. If there is no uncorrupted future parameter within the frame the last uncorrupted parameter is used for extrapolation.

For the lags the last uncorrupted value is used, if its distance to the corrupted lag is smaller than the distance to a future uncorrupted value or if there is no future uncorrupted value within the frame. Otherwise the future uncorrupted lag-value is used for extrapolation.

5. ZERO-REDUNDANT ERROR DETECTION

In figure 2 dashed lines were printed into each of the subplots, corresponding to parameter intervals on the x-axes with the probability of 0.9. The intervals can be found by the intersections of these dashed lines with the probability distributions for each D. The resulting intervals are not exceeded by uncorrupted parameters with a probability of 0.9, i.e. *if* they are exceeded this is caused with a probability of 0.9 by a channel error. So, parameter corruptions by bit-errors can be detected by checking, whether the intervals are exceeded. Since the probability of the interval was 0.9, there will be wrong error detections with a probability of 0.1 that initiate parameter-extrapolation without necessity.

The choice of the intervals (or the corresponding probability) was carried out by informal listening tests. An optimum had to be found between too many incorrect error detections caused by small intervals (low probabilities) and too many undetected true channel errors caused by large intervals (high probabilities), both resulting in poor speech quality.

For the optimization the indices of the speech codec were corrupted by random and burst bit errors generated by a simple Gilbert-Elliott channel model. While an interval for a parameter was optimized, the other parameters were checked by a comparison of the bits of the corrupted and uncorrupted indices (ideal error detection). If an error was detected by the zero-redundant or ideal detection the corresponding parameter was replaced by the algorithm described in section 4. The corruption of the channel indices was carried out for bit-error rates between 0% and 9%. This range covers most of the practical channel situations in mobile communications.

Figure 4 shows an example of the optimization for the log-value logP of average power for random bit errors. In

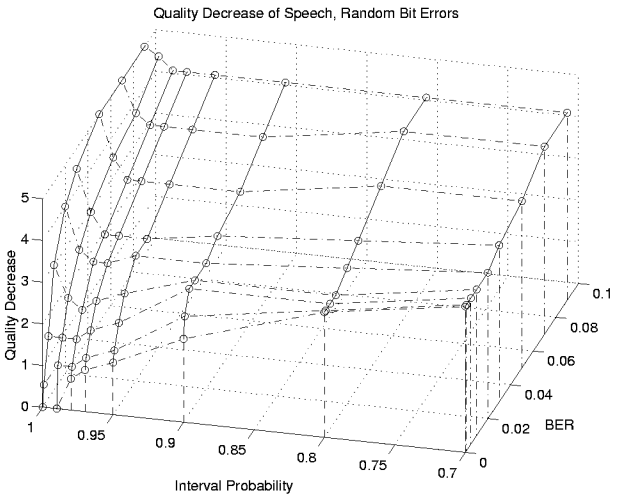

Figure 4: Optimization of the interval probability for the average power, channel corrupted by random bit errors

the plot, the quality decrease of decoded speech caused by channel errors and wrong error detections compared to the "clear-channel" quality of the codec is assigned to the z-axis. The values 0..5 are allocated to the subjectively perceived speech-quality decrease by table 1. In figure 4 there is a minimum for the quality decrease at an interval-probability of $W_{logP} = 0.98$ for bit-error rates (BER) between 0.5% and 3%, i.e. the range of bit-error rates that is of special interest for mobile communications. Below this probability too many errors are "detected" although no bit error was in the index of the average power, and above $W_{logP} = 0.98$ many errors remain undetected that cause severe degradations in speech-quality like "clicks" and "plops". Unfortunately, the "clear-channel"-quality is decreased by the zero-redundant error detection since 2% of the parameter differences are declared as errors.

For the other parameters the optimization of the error

Quality decrease	Valuation
inaudible	0
hardly audible	1
little annoying	2
annoying	3
very annoying	4
catastrophic	5

Table 1: Evaluation of the quality decrease of decoded speech from corrupted parameters compared to the "clear-channel" quality of the codec

detection intervals were carried out similarly. The optimal interval probabilities are 0.98 for the LSF-coefficients and the gain-parameters. For the lags the best probability is 0.80.

The results for burst bit errors are quite similar, but in general the quality decrease is lower, because bit errors concentrate on single parameters that are replaced anyway if an error has already occured in an "important" bit. The optimal interval probability is the same as for random error channels since the algorithm is only based on parameter-statistics and not on any information about the channel.

6. PERFORMANCE

The plots of the quality decrease for optimized error detections for all parameters are shown in figure 5.

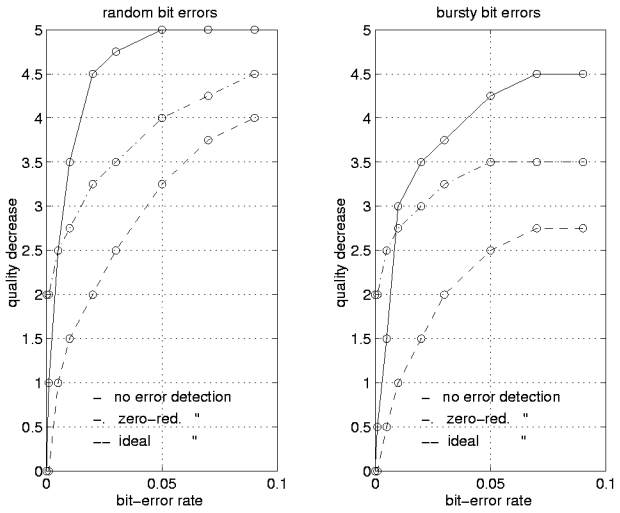

Figure 5: Quality decrease of speech for decoding without error detection (and bad-parameter handling) zero-redundant error detection and ideal error detection

Decoding without error detection results in a catastrophic quality decrease for bit-error rates above 5% for random bit errors. In case of burst errors the quality is better, and above a bit-error rate of 5% the quality is not decreased further because more and more bits of indices are corrupted that are extrapolated anyway because of already present bit errors. This is due to of the simple channel-model (Gilbert-Elliott) that was used. The main observation that random errors cause worse speech quality compared to burst errors is not affected by that.

The zero-redundant error detection strongly reduces the "clicks" and "plops" in the decoded speech signal. The quality is better than for the case without error detection above bit-error rates of 1%. The "clear-channel" quality is significantly reduced by the zero-redundant error detection but it remains acceptable. The strongly annoying distortions in the decoded speech signal caused by the bit errors are removed. Overall, the zero-redundant error detections lead to better speech quality, if errors are likely to occur on the channel compared to decoding without any error detection.

The comparison of the zero-redundant detection with the ideal error detection reveals that the quality decrease by channel errors can be significantly reduced further by improved error detections.

Since the investigations above showed that the optimizations are independent of the channel, the optimal intervals can also be used for a more realistic channel resulting from the output of a rate-1/2 convolutional code on the GSM full-rate traffic channel and the well known GSM error patterns. The results for the random-error channel at the same bit-error rate are worse compared to the realistic channel which is of "burst-error-type".

In an application the loss of "clear-channel" performance might not be acceptable. In this case a combination of the zero-redundant techniques with established redundant error detections can solve the problem.

7. CONCLUSIONS

The strongly annoying effects that occur without error detection have mostly been removed by the new bad-parameter detection and handling without additional redundancy. The clear-channel speech quality is affected by wrong error detections, but it remains acceptable. If combined with some redundant error detection applied to several or all of the indices, the additional bit rate for redundant error detection can be kept small while parameter-corruptions can still be located more accurately. The new algorithms don't require large amounts of memory since only small tables with the parameter statistics have to be stored, and they also don't entail additional complexity since they can be implemented by simple comparisons.

8. REFERENCES

[1] Norbert Görtz, "Vorschlag für einen GSM-Vollraten-Sprach- und Kanalcodec mit verbesserter Sprachqualität", Proceedings of "Aachener Kolloquium Signaltheorie 1997", March 1997 (in German)

[2] Electronic Industries Association, Cellular Systems Dual-Mode Subscriber Equipment - Network Equipment Compatibility Specification, Dec. 1989

[3] F.K. Soong, B.H. Juang, "Line Spectrum Pair (LSP) And Speech Data Compression", Proc. ICASSP, pp. 1.10.1-1.10.4, 1984

[4] K.K. Paliwal and B.S. Atal, "Efficient Vector Quantization of LPC Parameters at 24 Bits/Frame", IEEE Trans. on Speech and Audio Processing, Vol. 1, No. 1, pp. 3-14, January 1993

[5] J-P. Adoul, P. Mabilleau, M. Delprat and S. Morisette "Fast CELP coding based on algebraic codes", Proc. ICASSP, pp. 1957-1960, 1987

[6] Y. Linde, A. Buzo, R.M. Gray, "An Algorithm for Vector Quantizer Design", IEEE Trans. on Comm., Vol. COM-28, No. 1,pp. 84-95 Jan. 1980

[7] ITU-T Rec. G.728, "Coding Of Speech At 16kbit/s Using Low-Delay Code Excited Linear Prediction"