Audio Engineering Society
Convention Paper
Presented at the 110th Convention
2001 May 12–15 Amsterdam, The Netherlands

Error Protection and Concealment for HILN MPEG-4 Parametric Audio Coding

Heiko Purnhagen, Bernd Edler, Nikolaus Meine
Laboratorium für Informationstechnologie
University of Hannover
Schneiderberg 32, 30167 Hannover, Germany
Phone: +49-511-762-5033, Fax: +49-511-762-5052
{purnhage,edler,meine}@tnt.uni-hannover.de

ABSTRACT

The HILN (Harmonic and Individual Lines plus Noise) MPEG-4 parametric audio coding tool allows efficient representation of general audio signals at very low bit rates. Therefore possible applications include transmission over IP or wireless channels which are both characterised by specific transmission error models. On the other hand, since parametric audio coding is a relatively new technique compared to transform coding and CELP speech coding, there have been only very limited investigations on HILN's behaviour in error prone environments. In this paper we present an analysis of error sensitivities and approaches to error protection and concealment.

1 INTRODUCTION

The MPEG-4 parametric audio coding tool "Harmonic and Individual Lines plus Noise" (HILN) permits coding of general audio signals at very low bit rates using a parametric representation [1, 2, 3]. The basic idea is to decompose the input signal into components which are described by appropriate source models and represented by model parameters. This approach allows to utilise more advanced source modelling than spectral decomposition as commonly used in transform-based coders [4]. Similar to speech coding [5], where specialised source models resemble the human vocal tract, advanced source models can be advantageous in particular for very low bit rate coding.

Three different types of signal components are utilised in MPEG-4 HILN: The first component type is a harmonic tone described by its fundamental frequency and the spectral envelope of the amplitudes of its partials. The second type is a single sinusoid (i.e. an "individual line") characterised by its individual frequency and amplitude. The third type is noise specified by amplitude and spectral shape. The spectral shape of a harmonic tone or noise component is characterised by reflection coefficients as known from Linear Predictive Coding (LPC).

The audio signal is modelled as sequence of overlapping frames, while additional parameters permit to refine the temporal envelope of transient components.

The HILN encoder analyses the audio signal in order to decompose the input signal into components and to estimate the corresponding parameters. The HILN decoder synthesises all components and superimposes them to obtain the decoded audio signal. For the tonal components normally no phase is transmitted, but in the case of sinusoidal tracks spanning multiple frames phase continuity is preserved by the synthesis procedure. In HILN a maximum of one harmonic tone and one noise component can be used, while the maximum number of individual sinusoids is restricted by the bit rate and by criteria limiting the decoder complexity.

Transmission of an adequate number of parameters at bit rates down to 4 kbit/s requires very efficient parameter coding which also takes into account perception phenomena. However some of the involved techniques need special consideration when HILN is operated in an error prone environment. One example is a sinusoidal track where the start frequency and amplitude is transmitted in the first frame. For all subsequent frames until the end of the track differential parameter coding is applied which can lead to high temporal error propagation. Furthermore most of the parameters are entropy coded using variable length code words so that a bit error can cause incorrect decoding of the remaining parameters within a frame. This indicates that the efficiency of this representation needs to be traded against error robustness.

After an overview of the MPEG-4 HILN coder and its bit stream format (Section 2) approaches for improving the error protection and concealment are presented. One way to reduce error propagation is to modify the HILN encoder in a way that it restricts the maximum duration of differential encoding for signal components (Section 3). Unequal Error Protection (UEP) is enabled by assigning the bit stream elements to different "Error Sensitivity Categories" (ESC), which are supported by the Error Protection tool (EP tool) of the MPEG-4 Audio standard (Section 4). Even more extensive use of UEP is possible by HILN's bit rate scalable operation mode which additionally enables the efficient use of channels with priorisation capabilities (Section 5). In order to optimise the behaviour in the case of residual errors, strategies for error concealment are presented in Section 6. Finally first results using simulation of different error characteristics and corresponding encoder/decoder configurations are presented in Section 7.

2 HILN OVERVIEW AND BIT STREAM FORMAT

The parametric audio coding tools "Harmonic and Individual Lines plus Noise" (HILN) [3, 6] as defined in Version 2 of the MPEG-4 Audio standard [2] permit coding of general audio signals at bit rates of about 4 kbit/s and above using a parametric representation of the audio signal [7, 8, 9, 10]. The HILN encoding and decoding process and the bit stream format is reviewed in the following Subsections.

It should be noted that only the bit stream format and the decoding process are defined in the normative part of the MPEG standard, while an example for a possible encoding process is given in an informative annex. This permits to optimise the encoding process and/or adapt it to special requirements – like low encoder complexity or transmission over error-prone channels – even after finalisation of the standard.

2.1 HILN Parametric Audio Encoder

Fig. 1 shows the block diagram of a typical HILN parametric audio encoder as used during the development of the MPEG standard. The encoding is performed on overlapping frames of input samples. A typical frame length (hop size) for signals sampled at 16 kHz is 32 ms. Each frame of the input signal is decomposed into different components and the model parameters for the components' source models are estimated:

- An *individual sinusoid* is described by its frequency and amplitude.
- A *harmonic tone* is described by its fundamental frequency, amplitude, and the spectral envelope of its partials.
- A *noise* signal is described by its amplitude and spectral envelope.

Sinusoidal components that "live" for more than one frame are handled as sinusoidal trajectories to ensure phase continuity at frame boundaries. Although the phase parameters of sinusoidal components are estimated as well, they are usually not conveyed in the bit stream, thus exploiting the low phase sensitivity of the human ear. The modelling of transient signals is improved by optional parameters describing the component's temporal envelope. To represent the spectral envelope of the harmonic tone and the noise signal, LPC modelling as well known from the Linear Predictive Coding of speech signals is employed. The frequency response of an all-pole LPC synthesis filter is used as spectral envelope, and the number of LPC parameters used to describe the spectral envelope is adapted to provide the desired level of spectral detail.

As shown in Fig. 1, the signal decomposition and parameter estimation can be implemented as a three step process:

- First all sinusoidal components are extracted from the current frame of the input signal.
- If several sinusoids share a common fundamental frequency, they are grouped as a single *harmonic tone* to permit efficient coding, while all remaining sinusoids are handled as *individual sinusoids*.
- After extraction of all sinusoidal components, the magnitude spectrum of residual signal is used to find the parameters of the *noise component*.

Due to the very low target bit rates of typically 6 to 16 kbit/s, only the parameters for a small number of components can be transmitted. Therefore a perception model is employed to select those components that are most important for the perceptual quality of the signal.

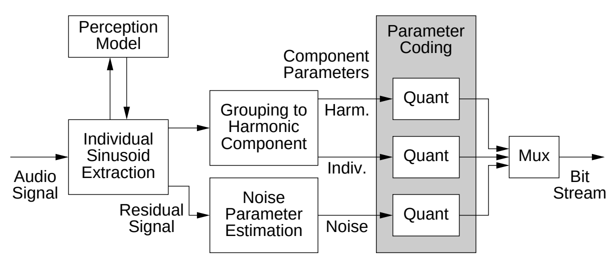

Fig. 1: Block diagram of an HILN parametric audio encoder.

2.2 HILN Bit Stream Format

The estimated parameters of the components selected in the encoder are quantised, coded, and multiplexed to form a bit stream. Non-uniform quantisation is used for the parameters to take into account perceptual properties like the "just noticeable differences" of amplitude and frequency. In HILN amplitudes of all components are quantised on a logarithmic scale with 1.5 dB step size. Frequencies of individual sinusoids are quantised on a Bark scale with 1/32 Bark step size (i.e. $\approx$ 3 Hz below 500 Hz and ≈ 10 cent above 500 Hz), while a logarithmic scale with ≈ 5 cent step size is used for the fundamental frequency of a harmonic tone. The LPC reflection coefficients describing the spectral shape of the harmonic and noise components are converted into Logarithmic Area Ratios (LAR) and then quantised with a step size of ≈ 0.1 (harmonic) or ≈ 0.3 (noise).

To permit efficient coding of the parameters of components that are continued from the previous frame, predictive coding is employed so that only frequency and amplitude changes are transmitted. For the LARs, this inter-frame prediction uses a predictor coefficient of 0.75 or 0.5 and takes into account the mean LARs of "default" low-pass spectra as shown in Fig. 2.

For new components, on the other hand, the amplitudes are coded relative to the maximum amplitude in the current frame, which is transmitted as "global gain" for each frame. In addition, usually a coarser quantisation with 3 dB step size is employed for the start amplitude of new individual sinusoids.

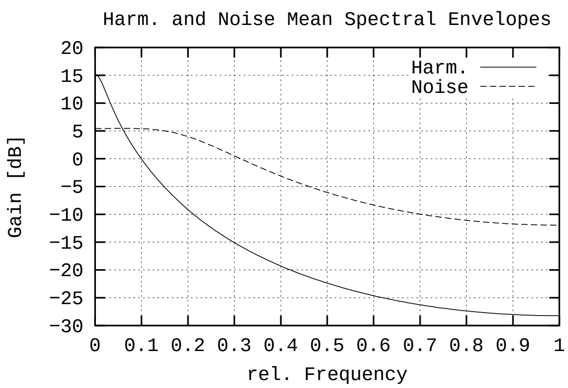

Fig. 2: "Default" low-pass spectra for mean LARs of harmonic and noise components.

For each frame, the total number of individual sinusoids and the presence of harmonic and noise components is signalled in the bit stream. If a harmonic tone is present, the number of partials is coded as well. Also the number of LARs for the spectral shape of harmonic and noise, i.e. the order of the all-pole spectral model, is transmitted. Continuation of components from the previous frame is signalled by so-called "continue flags." If parameters for the optional temporal envelope are transmitted, there is also an "envelope flag" for each component.

Advanced entropy coding schemes utilising various variable length codes (VLC) are applied to exploit the non-uniform distribution of the quantised parameters. To improve the coding efficiency for the parameters of new individual sinusoids, a special technique called "Sub-Division Coding" (SDC) is used [3].

2.3 HILN Parametric Audio Decoder

The block diagram of the HILN parametric audio decoder is shown in Fig. 3. First the parameters of the components are decoded and then the component signals are re-synthesised according to the transmitted parameters. By combining these signals, the output signal of the HILN decoder is obtained. Overlapping synthesis windows are used to obtain smooth transitions between frames. For sinusoidal trajectories, i.e. sinusoids continued from the previous frame, linear interpolation of amplitude and frequency parameters is used in the synthesis to provide phase continuity. Usually the start phase of a new ("born") sinusoid is not conveyed in the bit stream, so that a random start phase is used instead. To enable smooth transitions between sinusoids that are coded as individual line in one frame and as partial of a harmonic tone in the adjacent frame, they are "connected" in the decoder if their frequencies and amplitudes are similar enough.

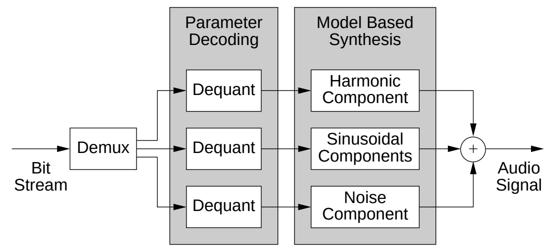

Fig. 3: Block diagram of the HILN parametric audio decoder.

A very interesting property of this parametric coding scheme arises from the fact that the signal is described in terms of frequency and amplitude parameters. This signal representation permits speed and pitch change functionality by simple parameter modification in the decoder without additional complexity.

3 ENCODING FOR ERROR-PRONE CHANNELS

The differential parameter coding for components that are continued over many frames can lead to serious temporal error propagation. If for example the frequency parameter at the beginning of a long sinusoidal track is corrupted, the resulting frequency error would remain until the end of the track.

One approach to reduce this problem is to force the encoder to restrict the maximum duration of the use of differential coding. A correspondingly modified encoder can limit the maximum number of consecutive frames with the "continue flag" set for a given component. Hence the encoder is forced to "re-start" the component again in the next frame using absolute coding of parameters. This requires the encoder to keep track of the current duration individually for each component. The maximum duration can be specified independently for the different component types, i.e. individual sinusoids, harmonic tone, and noise component.

An alternative approach is to "re-start" all components periodically every Nth frame, similar to the "intra-frames" (I-frame) as known from video coding. Such an "I-frame" can also be used as a random access point, thus improving the "tune-in" behaviour in case of continuous streaming.

The periodic "re-start" of sinusoidal components would lead to phase discontinuities for long trajectories if, as usual, no start phase information is conveyed and a random start phase is used. To avoid this problem and enable smooth synthesis of "re-started" sinusoids, the decoder can be configured to "connect" individual sinusoids in adjacent frames for the synthesis if their frequencies and amplitudes are similar enough. This is an extension of the "connection" mechanism described in Subsection 2.3.

4 UNEQUAL ERROR PROTECTION

The MPEG-4 audio standard addresses not only compression but also provides means for robust transmission over error-prone channels. This is accomplished by mechanisms enabling Unequal Error Protection (UEP) according to the individual error sensitivity of the different bit stream elements. In addition, the source coding schemes were designed to provide good error resilience (i.e. fair behaviour in case of transmission errors) and to allow for efficient error concealment in the decoder.

In order to enable the use of UEP for HILN encoded data, the standard describes the assignment of coded parameters to "Error Sensitivity Categories" (ESC). This assignment is explained in the next Subsection and takes into account the two aspects, that the different ESCs should contain parameters of similar importance and that each ESC packet can be parsed individually and thus provides a re-synchronisation point.

Subsection 4.2 describes the "Error Protection Tool" (EP tool) defined in MPEG-4 [2]. This configurable channel coding tool allows to individually select the degree of error protection for the ESCs using different code rates and provides flexible "Cyclic Redundancy Check" (CRC) for error detection.

4.1 Definition of Error Sensitivity Categories

For HILN, a total of 5 ESCs are defined, starting with the first category, named ESC0, which holds the bit stream elements with highest error sensitivity. A summary of the assignment of bit stream elements (i.e. coded representation of parameters) to ESCs is shown in Table 1 on page 5.

ESC0 contains the main control data, e.g. flags signalling the presence of the different component types and the total number of individual lines. If present, frequency and amplitude parameters of new harmonic and noise components are also included in ESC0.

ESC1 contains the parameters describing the more important part of the spectral envelope parameters for harmonic and noise components and, if present, amplitude and frequency changes of harmonic and noise components continuing from the previous frame. At its end this category contains the flags indicating which individual lines are continuing from the previous frame. The number of flags is equal to the number of individual lines in the previous frame, which would prevent further bit stream parsing in cases where this number is unknown.

ESC2 contains for each individual line a flag activating the temporal envelope, if present. Furthermore the frequency and amplitude parameters of all new individual lines are transmitted here. Their number is equal to the difference of the total number specified in ESC0 and the number of continuing lines signalled by the continue flags in ESC1.

ESC3 contains the temporal envelope parameters, the less important spectral envelope parameters of harmonic and noise components, and the frequency and amplitude changes of all continuing individual lines. At the end ESC3 contains the frequency stretching parameter for harmonic tones.

ESC4 is only needed in rare cases, since the transmission of start phase parameters and the temporal envelope for the noise component which would be included here is usually not activated.

For designing the error protection for the ESCs an analysis of their importance with respect to signal quality is necessary.

If an ESC0 packet gets lost, no signal component of the affected frame can be decoded and synthesised. Furthermore, the number of continue flags for individual lines in the following frame is unknown, so that only parameters for new harmonic and noise components can be decoded. Even one more frame is required before the decoding of new individual lines can be re-started.

Since the continue flags for individual lines are contained in ESC1, no individual line parameters can be decoded in the case that one of its packets gets lost. Furthermore harmonic and noise components cannot be continued correctly. However, in contrast to ESC0, individual lines starting in the following frame can be decoded.

Loss of an ESC2 packet disables correct decoding of parameters for new individual lines. The resulting error will propagate for as many frames as the lines are continued using differential encoding. In addition temporal envelopes cannot be applied correctly in the current frame.

The main impact of a lost ESC3 packet is that the shape of the temporal envelope is not available and that individual lines cannot be continued correctly. Due to the limited range of possible frequency and amplitude changes the impact on following frames is relatively small.

Another aspect of the ESC definition is the distribution of the overall bit rate amongst the ESCs. Since almost all parameters are encoded with variable length or even can be completely disabled, strong variations of the bit numbers are usual. Therefore the minimum, maximum, and average ESC packet lengths are of interest. Results of a statistical analysis for two bit rates 6 and 16 kbit/s and 39 items are shown in Tables 2 and 3 (from [11]). It should be noted that the encoder was allowed to use a bit reservoir of a size corresponding to 64 ms.

ESC	Min	Max	Avg	%tot
0	14	41	18.2	9.7%
1	0	74	32.1	17.2%
2	0	229	70.2	37.5%
3	0	146	66.7	35.6%
4	0	0	0.0	0.0%
all	14	290	187.1	100.0%

Table 2: ESC packet size [bit] for 39 items coded at 6 kbit/s with a frame length of 32 ms (20801 frames, 665.632 s).

ESC	Min	Max	Avg	%tot
0	15	42	18.9	3.9%
1	0	112	59.8	12.5%
2	0	699	235.8	49.2%
3	0	348	164.4	34.3%
4	0	0	0.0	0.0%
all	15	782	478.9	100.0%

Table 3: ESC packet size [bit] for 39 items coded at 16 kbit/s with a frame length of 32 ms (20792 frames, 665.344 s).

4.2 The MPEG-4 Error Protection Tool

Depending on the nature of the transmission channel, various error models are well established. They range from simple random bit error models for noisy channels over models for channels with bursty errors common in mobile radio communication to models for packet loss on networks using IP or similar protocols.

To address channels with random or bursty error characteristics, the MPEG-4 standard defines a flexible EP tool [2, 12] which covers both Forward Error Correction (FEC) codes and Cyclic Redundancy Check (CRC) codes for error detection. To accommodate a wide range of transmission channel conditions, it permits flexible configuration of the error correction and/or error detection capabilities for each class of bits corresponding to an ESC. Thus the redundancy required to provide the desired error protection can be minimised.

Most of the EP tool configuration is transmitted “out-of-band,” e.g. in a bit stream header. However, for some source coding schemes (like HILN), the structure and size of the error sensitivity classes can vary from frame to frame. This is accommodated by transmitting the required configuration “in-band,” i.e. in the bit stream frames.

The block diagram of the complete error protection encoder is depicted in Figure 4. First a CRC of up to 32 bits is added to the data bits of each class to permit error detection. Then a Systematic Rate-Compatible Punctured Convolutional (SRCPC) code is applied to provide forward error correction. The puncturing procedure allows fine adjustment of the code rate, i.e. the redundancy added by the code. Alternatively a Shortened Reed-Solomon (SRS) code can be used. The CRC and SRCPC/SRS coded data bits of the different error protection classes as well as the “in-band” part of the EP tool configuration is then processed by an interleaver to improve the robustness for bursty errors. Finally the interleaved data is protected with another SRS code. The SRCPC and SRS codes utilised here are very similar to those described in the Annex of H.223 [13].

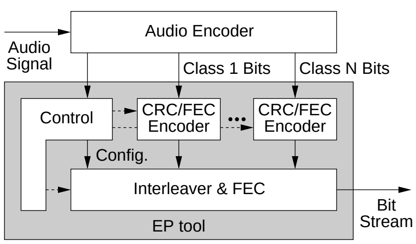

Fig. 4: Block diagram of an Encoder with Unequal Error Protection (UEP) using the EP tool (from [2]).

It is not required to actually utilise the error resilience information in the decoder. In the simplest case, the redundancy added by the EP tool is removed and de-interleaving is applied. However, to make full use of an error protected bit stream, a more complex decoder is required, which can e.g. include a Viterbi decoder for the SRCPC code.

5 BIT RATE SCALABILITY

While the definition of ESCs already gives the possibility to protect the parameters according to their importance, the loss of parameters still can lead to clearly audible distortions. This is mainly caused by interdependencies between different ESCs. Therefore MPEG-4 HILN can be operated in a bit rate scalable mode, also known as embedded coding, where the bit stream is hierachically divided into a base layer and one or multiple extension layers [14, 15]. The hierarchical structure ensures that lower layers starting with the base layer can be decoded independent of the presence of the higher layers. Thus UEP can be implemented by using strong error protection for the base layer and weak or no protection for the extension layers.

Furthermore, in contrast to the ESCs, the different layers can be transmitted as completely separate bit streams. This enables the efficient use of channels with priorisation capabilities (“Quality of Service,” QoS), since even the complete loss of higher layers does not lead to explicit distortions.

The design of the scalable HILN system takes into account the fact that a maximum of only one harmonic component and one noise component is possible per frame. Therefore the number of bits for transmitting the corresponding parameters exposes only small dependencies on the total bit rate. This is also reflected in the ESC packet sizes, as can be seen in Tables 2 and 3. An increase of bit rate mainly leads to an increased number of individual lines being transmitted.

Therefore bit rate scalability for HILN is implemented by dividing the parameters for individual lines according to their importance for the audio quality into subsets for the different scalability layers. If present, the parameters for harmonic and noise components are always included

Component	**ESC0**	**ESC1**	**ESC2**	**ESC3**	**ESC4**
Frame Configuration	num. indiv. lines harm. flag noise flag env. flag phase flag max. ampl.			env. shape (H & I)	
Harmonic (H)	cont. flag env. flag start ampl. (new) start freq. (new)	spectral env. (a) Δ ampl. (cont.) Δ freq. (cont.)		spectral env. (b) freq. stretching	start phase (new)
Individual Lines (I)		cont. flags	env. flags start ampl. (new) start freq. (new)	Δ ampl. (cont.) Δ freq. (cont.)	start phase (new)
Noise (N)	cont. flag env. flag start ampl. (new)	spectral env. (a) Δ ampl. (cont.)		spectral env. (b)	env. shape (N)

Table 1: Assignment of component parameters to Error Sensitivity Categories (ESC).

in the base layer. The perceptually most important individual line parameters are also assigned to the base layer, while the least important subset is transmitted in the highest layer. However, in order to achieve real scalability with the lowest possible loss of coding efficiency, special care has to be taken about line continuation.

In order to avoid error propagation from a higher to a lower layer, line continuations in this direction are not allowed. This restriction can result in a slight decrease of coding efficiency. In addition to that, the only overhead compared to the non-scalable case is the transmission of the number of individual lines and the continue flags in each extension layer. Continue flags are only needed for those lines which are not continued within the lower layers.

An additional difference in the behaviour of scalable HILN when compared to non-scalable HILN at corresponding bit rates can be caused by differences in the bit assignments for harmonic and noise components. In the non-scalable case it can be optimised individually, while in the base layer of a scalable system a compromise must be made between optimum assignments for the lowest and the highest bit rate.

Nevertheless, subjective evaluations carried out during MPEG-4 standardisation showed the efficiency of this approach to scalability. Scalable HILN operating with one base layer of 6 kbit/s and one extension layer of 10 kbit/s was compared to non-scalable HILN operating at 6 and 16 kbit/s respectively. In this test no significant difference was observed between the two operating modes at both bit rates as shown in Fig. 5.

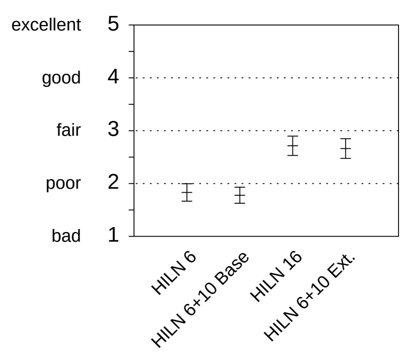

Fig. 5: MPEG-4 Version 2 verification test results for HILN at 6 and 16 kbit/s showing overall mean grading and 95% confidence interval for 7 items and 16 listeners (from [16]).

It should be noted that, different from most other scalable coding schemes, the audio bandwidth of the decoded signal does not depend on the number of scalability layers available to the HILN decoder.

6 ERROR CONCEALMENT

In cases of residual errors after error protection, it is important to have a decoder with well defined behaviour depending on the kind of parameters that got corrupted or lost. Although no specific behaviour is requested by the MPEG-4 standard, we propose the following mechanisms for getting the best possible audio quality using as many of the correctly received parameters as possible. While the following description is organised by the type of affected parameters, it corresponds to the behaviour for loss of an ESC packet containing the respective parameters.

First, validity flags should be introduced for all signal components. They are necessary to avoid uncontrolled decoding of paramters depending on other parameter values. Furthermore they can be used for muting the corresponding synthesiser. They need to be carried from frame to frame to keep track of validity in case of continuing components. For individual lines reordering is necessary to track the position of corresponding lines in the actual transmission order.

If the parameter for the maximum amplitude of all components is corrupted, all components should be flagged invalid. The same holds for the harmonic or noise component, if one of the corresponding presence indicator or continue flags is corrupted.

A new harmonic component is also to be regarded invalid in the case of a corrupted start amplitude or start frequency. If however no valid spectral envelope information is available we propose to use default LPC parameters corresponding to an averaged low-pass characteristic, as shown in Fig. 2. For continued harmonic components a good default behaviour for replacing corrupted differentially encoded parameters is to keep the frequency constant, to reduce the amplitude by an attenuation factor (e.g. -6 dB), and to let the spectral envelope converge towards that of the averaged lowpass characteristic. An alternative for the spectral envelope would be to keep it unchanged. With respect to amplitudes and spectral envelopes, noise components can be treated the same way as harmonic components.

If the number of individual lines is corrupted, no line parameters can be decoded in the current frame. Additionally in the next frame all individual lines must be flagged invalid, because the number of continue flags is still unknown. New lines with corrupted start amplitudes or start frequencies are also to be regarded invalid. The default behaviour for continued lines in case of corrupted parameters is to keep the frequency constant and reduce the amplitude.

For corrupted envelope parameters medium attack and decay rates can be used in conjunction with the peak set to the frame centre. If corrupted start phases are received, the decoder should behave as it would without phase transmission.

Due to the fact that the HILN synthesis procedure according to the standard implements fade-in for all new components and fade-out for all discontinued components, a decoder behaving as described above responds to residual errors as graceful as possible.

7 RESULTS

First evaluations of HILN's capabilities to cope with error prone channels focused on the influence of the encoder modifications described in Section 3 on the resulting audio quality. As a typical setup a frame length of 32 ms and a limitation of the duration of all component types to a maximum of 10 frames was chosen. This limits the maximum temporal error propagation to approximately 1/3 s. Informal subjective assessment did not reveal any audible degradation compared to the original encoder.

In a next step the sensitivity to random bit errors without any error protection was investigated. The modifications of the decoder compared to a bare standard implementation consisted of basic plausibility checks. In cases where parameters exceeded the specified limits concealment techniques as described in Section 6 were activated. As expected, bit error rates of 10^{-4} can already lead to clearly audible artifacts. However the kind of artifacts was quite different than those known from other coding schemes.

The actual significance of the different ESCs was evaluated by simulation of packet loss with individually controllable error rates. In cases of errors, appropriate concealment according to Section 6 was activated.

The outcome of these investigations was that the practical significance matched the assumptions quite well: Loss of ESC0 packets revealed the highest impact on audio quality, while even with 100% loss of ESC3 packets tolerable quality was maintained.

A configuration of the MPEG-4 EP tool for HILN bit streams was designed to add the required in-band configuration data and CRC for error detection. An analysis of this setup in the presence of random bit errors showed that the error detection and re-synchronisation in conjunction with the described concealment technique already provided clear improvements over the unprotected mode described above. However the required overhead for a system without error protection capabilities appeared to be relatively high, i.e. close to 50% for a bit rate of 6 kbit/s. The reason for this was found in the relatively small ESC packet lengths in conjunction with their variability which required a lot of in-band configuration data and corresponding header protection in the EP tool.

Compared to this, an additional FEC with code rates adjusted to the different ESCs was found to be very effective. Experiments with simulated random bit errors showed the strength of the combination of this protection with concealment for residual errors.

Final investigations focused on the bit rate scalable mode in order to examine the behaviour under conditions where the base layer of 6 kbit/s is transmitted without errors, but one extension layer of 10 kbit/s is exposed to a relatively high packet loss rate. Informal subjective evaluation showed that the transitions from the full to the reduced rate and back worked seamlessly. Even if they occurred frequently, they did not become annoying, especially because the audio bandwidth was not modulated with the error pattern.

8 CONCLUSIONS

In this paper a first implementation of an MPEG-4 HILN system dealing with transmission errors was presented. It provides an encoder capable of generating bit streams with restricted temporal error propagations. In addition, the standardised interface to tools for unequal error protection and the bit rate scalable operation mode were described. Optional error concealment techniques were proposed, which are specially designed for this type of parametric audio coder.

Tests under different error conditions showed that especially for the very low bit rates specific tuning of error detection and/or correction tools is necessary. Further investigations might consider the combination of multiple subsequent frames, if not prevented by transmission delay constraints.

Finally, bit rate scalability proved to be a very powerful way to transmit HILN encoded data over channels with priorisation capabilities under conditions with strongly varying channel bandwidth.

9 REFERENCES

[1] R. Koenen, *Overview of the MPEG-4 Standard*, ISO/IEC JTC1/SC29/WG11 N3156, Dec. 1999. `http://www.cselt.it/mpeg/standards/mpeg-4/mpeg-4.htm`

[2] ISO/IEC 14496-3/AMD1:2000, *Coding of audio-visual objects – Part 3: Audio (MPEG-4 Audio Version 2)*, ISO/IEC International Standard, 2000.

[3] H. Purnhagen and N. Meine, "HILN - The MPEG-4 Parametric Audio Coding Tools," *Proc. IEEE ISCAS 2000*, Geneva, May 2000.

[4] K. Brandenburg and M. Bosi, "Overview of MPEG Audio: Current and Future Standards for Low Bit Rate Audio Coding," *J. Audio Eng. Soc.*, Vol. 45, No. 1/2, pp. 4–21, Jan./Feb. 1997.

[5] B. Edler, "Speech Coding in MPEG-4," *Int. J. of Speech Technology*, Vol. 2, No. 4, pp. 289–303, May 1999.

[6] H. Purnhagen, N. Meine, and B. Edler, "Speeding up HILN – MPEG-4 Parametric Audio Encoding with Reduced Complexity," *AES 109th Convention*, Preprint 5177, Los Angeles, Sep. 2000.

[7] B. Edler and H. Purnhagen, "Parametric Audio Coding," *Proc. International Conference on Signal Processing (ICSP2000)*, Beijing, August 2000.

[8] H. Purnhagen, "Advances in Parametric Audio Coding," *Proc. IEEE WASPAA*, Sep. 1999.

[9] R. McAulay and T. Quatieri, "Speech Analysis/Synthesis Based on a Sinusoidal Representation," *IEEE Trans. ASSP*, Vol. 34, No. 4, pp. 744–754, Aug. 1986.

[10] M. Goodwin, *Adaptive Signal Models: Theory, Algorithms, and Audio Applications* , PhD thesis, University of California, Berkeley, 1997.

[11] H. Purnhagen, B. Edler, and N. Meine, *Study of HILN in 14496-3 FPDAM1 and proposed revisions*, ISO/IEC JTC1/SC29/WG11 M5529, Dec. 1999.

[12] H. Purnhagen, "An Overview of MPEG-4 Audio Version 2," *Proc. AES 17th International Conference*, Florence, Sep. 1999.

[13] ITU-T, *Recommendation H.223 - Multiplexing protocol for low bit rate multimedia communication*, International Telecommunication Union, Mar. 1996.

[14] B. Feiten, R. Schwalbe, and F. Feige, "Dynamically Scalable Audio Internet Transmission," *AES 104th Convention*, Preprint 4686, Amsterdam, May 1998.

[15] T. Verma, *A Perceptually Based Audio Signal Model with Application to Scalable Audio Compression*, PhD thesis, Stanford University, 2000.

[16] ISO/IEC, *Report on the MPEG-4 Audio Version 2 Verification Test*, ISO/IEC JTC1/SC29/WG11 N3075, Dec. 1999. `http://www.tnt.uni-hannover.de/project/mpeg/audio/public/w3075.pdf`

Further references and related links can be found at:
`http://www.tnt.uni-hannover.de/~purnhage/`