ASAC - Analysis/Synthesis Audio Codec for Very Low Bit Rates

Bernd Edler, Heiko Purnhagen, Charalampos Ferekidis

Institut für Theoretische Nachrichtentechnik und Informationsverarbeitung

University of Hannover, Germany

Abstract

An Analysis/Synthesis Audio Codec (ASAC) is presented, which allows the coding of audio signals at very low bit rates for applications like mobile communication or multimedia database access via modem and analog telephone lines. Compression with bit rates between 6 kbit/s and 24 kbit/s is addressed. Furthermore the implementation of special effects like independent pitch change and speed change in the decoder is described.

1 Introduction

Many applications for very low bit rate storage and transmission of digital audio signals mainly focus on speech signals. Therefore a lot of optimizations for speech codecs operating at bit rates between 6 and 16 kbit/s were carried out during the past years. However there is an increasing interest in extensions which also allow the transmission of music signals in areas like mobile communications or multimedia database access via modem and analog telephone lines [1]. The presented Analysis/Synthesis Audio Codec is therefore designed to represent music signals with bit rates between 6 kbit/s for telephone bandwidth (8 kHz sampling) and 24 kbit/s for wideband signals (48 kHz sampling) [2], [3].

The current implementation of the Analysis/Synthesis Audio Codec is based on the model of single spectral lines. However, the system can be extended by using more complex synthesis schemes like FM-Synthesis for the generation of harmonic spectra.

The major differences of this codec with respect to other systems which are also based on sinusoidal representations [4] result from the iterative analysis/synthesis approach for the parameter extraction. First this allows in each step to obtain the components of the intermediate residual error which are most significant for the subjective quality. Furthermore a high resolution frequency estimation technique is applied which can generate a parametric description for sinusoidal components which either have a constant frequency or a linearly changing frequency. By using amplitude envelope parameters, rapid amplitude changes within frames can also be taken into account.

In Section 2 a detailed description of the analysis/synthesis encoder is given, including the main components *Pre-Analysis*, *Analysis/Synthesis Loop*, *Psychoacoustic Model*, and *Quantization and Coding* unit. This encoder is able to generate two bitstreams, a *basic bitstream* which does not contain phase information and an additional *enhancement bitstream* mainly consisting of coded phase parameters which are needed for applications in which the final residual error is coded at a higher bit rate in order to obtain a scalable representation for high quality audio signals. The decoder with two modes for decoding with or without phase parameters is described in Section 3, where also the implementation of additional playback features like pitch and time scale modification is presented. The results obtained with the described Analysis/Synthesis Audio Codec are discussed in Section 4 and conclusions are drawn in Section 5.

2 Analysis/Synthesis Audio Coder

The basic principle of the Analysis/Synthesis Audio Codec is to analyze the input signal in order to extract parameters which can be transmitted instead of signal or subband signal samples. The input signal is processed in frames with a constant length and for each frame, a set of parameters is extracted, quantized, coded, and transmitted. At the decoder side, an audio signal is synthesized from the received parameters. The block diagram of the analysis/synthesis encoder is shown in Figure 1.

Since different parameter sets and different synthesis techniques can be applied, the input signal of the encoder has to be split up in an appropriate way. This is performed by the *Separation* unit. Depending on the most appropriate synthesis technique, a parameter set for each part of the input signal is derived in the *Model Based Parameter Estimation* unit. The two units *Separation* and *Model Based Parameter Estimation* can be regarded as the analysis stage which produces a parametric description of the input signal. The separation of the input signal is enhanced by feeding back the signals which are generated in the *Synthesis* unit from all the parameters of previously separated signal components. The *Separation* and the *Model Based Parameter Estimation* additionally receive data from a synthesis model independent *Pre-Analysis*. Prior to transmission, the parameters are fed through the *Quantization and Coding* unit, which is controlled by a *Psychoacoustic Model*. This *Psychoacoustic Model* processes the input signal in order to derive information about the relevancy of synthesis parameters. In addition, the synthesized signal is fed into the *Psychoacoustic Model*, which thus is allowed to assist the *Model Based Parameter Estimation*.

In the following, a detailed description of the different blocks is given for the current implementation, which is based on the synthesis model „individual spectral lines“, where each line is specified by the parameters frequency, amplitude, and phase.

2.1 Pre-Analysis

To improve modeling of fast signal transients, a *Pre-Analysis* is performed prior to the *Analysis/Synthesis Loop*. In the current implementation, the *Pre-Analysis* performs two tasks. First it determines for each frame whether a long or a short analysis window is used in the *Analysis/Synthesis Loop*. Thereafter the amplitude envelope of the signal in the current frame is analyzed and a set of parameters describing an appropriate envelope is calculated. The *Analysis/Synthesis Loop* later determines for each line whether or not this amplitude envelope is used.

The decision between the long and the short analysis window is based on the ratio between the maximum amplitudes within the frame and within the intervals in which the long window function exceeds the frame, i.e. ½ frame length before the frame start and ½ frame length after the frame end. If the ratio of the maximum amplitudes outside the frame and inside the frame exceeds a given threshold, the short window function is used. Otherwise, the long window is used. The long window is a Hanning window twice as long as a frame while the short window is a rectangular-like window with smoothed transitions at the beginning and end. Due to the smoothed transitions, the short window is slightly longer than a frame.

The block diagram of the envelope parameter generation is shown in Figure 2. To calculate the original envelope of a frame, the input is fed through a bandpass filter which attenuates very high and very low frequency components. Thereafter a Hilbert transformer is used to reconstruct the imaginary part of the signal so that the momentary amplitude of the now complex signal can be calculated. The envelope is then normalized with respect to its maximum amplitude. The bandpass filter and the Hilbert transformer are implemented as FIR filters. The length of the FIR filters depends on the sampling frequency and the FIR coefficients are derived from a prototype lowpass and Hilbert transformer impulse response

by appropriate windowing. For a sampling frequency of 8 kHz the frequency responses of these FIR filters are shown in Figure 3. The bandpass filter is necessary to attenuate the signal components at frequencies at which the Hilbert transformer cannot operate properly. Otherwise e.g. a low frequency signal would disturb the envelope parameter generation.

The envelope model used in this codec consist of an attack and a decay phase. Three parameters are used to describe the shape of the envelope:

- t_{max}: time of maximum amplitude, specifying the end of the attack phase which is also the beginning of the decay phase
- r_{atk}: attack rate, specifying the angle of the attack slope
- r_{dec}: decay rate, specifying the angle of the decay slope.

The time parameter is encoded in relation to the frame length. Both rates are converted into angles to simplify encoding of steep slopes as well as a constant maximum amplitude before or after t_{max}. If a steep decay slope reaches zero amplitude before the end of the frame, the remaining part of the envelope is set to zero. In the same way, a steep attack slope causes the envelope to be zero before the attack phase itself begins. In Figure 4, the original envelope of one frame of the signal „castanets“ is shown together with the envelope generated using the parameters t_{max}, r_{atk}, r_{dec} found by the envelope parameter estimation.

When determining the three envelope parameters, first of all t_{max} is found. The parameter t_{max} indicates the point at which the original envelope reaches its maximum for the first time. Next the RMS (root mean square) amplitude of the original envelope is calculated. To determine r_{atk} and r_{dec} linear regression including is employed. The attack rate r_{atk} is the gradient of that line going through the maximum at t_{max} that fits best the original envelope prior to the maximum. To improve performance a weighting function is used. It gives no weight to those parts of the original envelope that are below the RMS amplitude while the weight continuously increases with increasing amplitude. Also the weight increases with increasing distance to t_{max} on the time axis. The decay rate r_{dec} is the gradient of that line that fits best the original envelope after the maximum. The same weighting function is used as for the attack, but the line is not any longer forced to go trough the maximum at t_{max}. Nevertheless only the gradient of the line is used as r_{dec} while the vertical position of the line is not utilized.

If the attack as well as the decay rate are below a threshold (which means that there are no fast changes in the signal amplitude), the gradient of a line that fits best the original envelope for the whole frame is calculated. The weighting function now only depends on the amplitude, not on the time difference relative to t_{max}. If the gradient is positive, it is used as r_{atk} and t_{max} is set to the frame end so that a continuously increasing amplitude is modeled by the envelope. If the gradient is negative, it is used as r_{dec} and t_{max} is set to the frame beginning.

After the three parameters are determined, an envelope is generated using these parameters. As mentioned above, the *Analysis/Synthesis Loop* determines for each spectral line whether this generated envelope or a constant amplitude should be used.

2.2 Analysis/Synthesis Loop for individual spectral lines

For each frame, the encoder uses an *Analysis/Synthesis Loop* (Figure 5) to calculate the parameters of the spectral lines iteratively in order of their „significance“. A high resolution frequency estimation technique is employed for each line. To determine the most significant line in each step of the iteration, a *Psychoacoustic Model* is used. This model calculates the masked threshold for a synthesized signal containing those single spectral lines that already have been extracted from the current frame. The *Model Based Parameter Estimation* for the *i*-th line in the loop consists of the following steps:

- calculation of the difference between absolute FFT spectra of input and synthesized signals

- selection of the „most significant“ FFT line with center frequency $f_{i,m}$
- high resolution frequency estimation in the surrounding of $f_{i,m}$
- amplitude and phase estimation, selection of envelope information

The FFT line to be processed is determined by calculating the difference between input spectrum $|X(f)|$ and synthesized spectrum $|S_i(f)|$ and searching the maximum ratio of the square of this difference and the masked threshold $|M_i(f)|$ calculated from the signal $s_i(n)$ synthesized from the previously determined spectral lines.

Based on the center frequency $f_{i,m}$ of the selected FFT line a frequency estimation derived from [5] is performed in order to obtain frequency parameters of higher accuracy than the FFT resolution. The original frequency estimation based on a linear regression for phase values has been extended to a second order regression which allows the estimation of frequencies changing linearly within frames (Figure 6). For this frequency estimation, first the spectrum of the residual error signal $e_i(n)$ is shifted by $-f_{i,m}$ by multiplication with a complex harmonic:

$$v(n) = e_i(n)\, e^{-j2\pi n f_{i,m}/f_s}$$
$$\Rightarrow V(f) = E_i(f - f_{i,m}),$$

where f_s denotes the sampling frequency. The complex output values $v(n)$ of this operation are fed through a lowpass filter and sampling rate reduction with factor K before the calculation of phase values for the resulting samples $w(k)$. In order to obtain phase values which can exceed the interval $[-\pi...\pi]$, for each pair of two consecutive samples first the phase differences are calculated

$$\Delta\varphi(k) = \arg\left(w(k)\, w^*(k-1)\right) \quad \text{for } k \geq 1 .$$

These differences can be assumed to be unique, if the reduced sampling rate is larger than two times the stopband edge frequency of the lowpass filter. To the phase values obtained by accumulation

$$\varphi(k) = \sum_{i=1}^{k} \Delta\varphi(i) ,$$

a second order regression is applied which gives a parabolic approximation $\hat{\varphi}(t)$ (Figure 7a). Differentiation

$$\Delta\hat{f}(t) = \frac{1}{2\pi}\frac{d}{dt}\hat{\varphi}(t)$$

gives a frequency offset which can be added to $f_{i,m}$ to obtain an estimate for a linearly changing frequency (Figure 7b):

$$\hat{f}(t) = f_{i,m} + \Delta\hat{f}(t) \ .$$

Substitution of t by the start time t_s and the end time t_e of the frame gives the offsets $\Delta f_{i,e}$ and $\Delta f_{i,s}$ and the corresponding frequency parameters $f_{i,e}$ and $f_{i,s}$:

$$\begin{aligned} \Delta f_{i,s} &= \Delta\hat{f}(t_s), \quad f_{i,s} = f_{i,m} + \Delta f_{i,s} \\ \Delta f_{i,e} &= \Delta\hat{f}(t_e), \quad f_{i,e} = f_{i,m} + \Delta f_{i,e} \end{aligned} .$$

Based on $f_{i,s}$ and $f_{i,e}$ the amplitude and phase of the spectral line are calculated. This is realized by calculating a complex correlation coefficient of the residual error signal and a complex harmonic signal with a frequency linearly changing from $f_{i,s}$ to $f_{i,e}$ within the frame. The

absolute value of the correlation coefficient gives the amplitude parameter a_i and the argument gives the phase parameter φ_i. If the *Pre-Analysis* has generated envelope parameters, a second set of parameters $a_{i,env}$ and $\varphi_{i,env}$ is calculated by correlation of the residual error signal and the complex harmonic signal which is additionally multiplied with the according envelope.

The start frequency $f_{i,s}$ and the end frequency $f_{i,e}$ are used in the following synthesis stage in order to minimize the residual error in the case of signal components with frequencies varying within the frame. Furthermore these parameters are used to decide whether lines are regarded as to be continuing from one frame to the next. However, only the arithmetic mean

$$f_i = \frac{f_{i,s} + f_{i,e}}{2} ,$$

which is equal to the estimated frequency in the middle of the frame is quantized and coded as described in Section 2.4.

The *Synthesis* unit generates a sinusoidal signal according to the parameters $f_{i,s}$, $f_{i,e}$, a_i and φ_i. If envelope parameters are available, a second sinusoidal signal is synthesized from the parameters $f_{i,s}$, $f_{i,e}$, $a_{i,env}$ and $\varphi_{i,env}$ and multiplied with the envelope function delivered by the *Pre-Analysis*.

In the *Separation* unit a new residual error signal is generated by subtracting the synthesized signal. If a second signal synthesized with envelope is available, a second residual error signal is calculated. In this case the parameter set leading to the lowest residual error variance is selected and the corresponding residual error signal is used in the further analysis steps.

2.3 Psychoacoustic Model

The *Psychoacoustic Model* used in the analysis/synthesis encoder calculates a masked threshold from a given input signal. It has been derived from the model for high quality audio coding as described in [6]. On one hand it has been simplified in a way that pre- and post-masking effects are not taken into account, and on the other hand it has been generalized in order to handle different FFT resolutions and bandwidths according to the needs for different sampling rates of the input signals. This is implemented by mapping the complex frequency components obtained from an FFT-based spectral analysis to partitions with a constant width on the Bark scale. After calculating the masked threshold for these partitions they are mapped back to the resolution of the FFT in order to allow a comparison of masked thresholds and FFT-spectra.

In the current implementation, the *Psychoacoustic Model* is used in the *Analysis/Synthesis Loop* in order to calculate masked thresholds for the signal synthesized from previously determined parameters (Figure 5). This procedure allows to find signal components, which are not yet covered or masked by the synthesized signal, and thus allows the encoder to find the „most significant“ component of the residual error.

2.4 Quantization and Coding

In the *Quantization and Coding* unit the parameters are processed in the order as they are obtained from the *Analysis/Synthesis Loop*, since this order corresponds to their „significance. This unit is able to generate two bitstreams, a *basic bitstream* which allows the generation of the basic quality audio signal, and an *enhancement bitstream* which can be used in applications where a difference signal between input and decoder output is needed, e.g. for scalable codecs. The *basic bitstream* mainly contains the frequency, amplitude and envelope

parameters, while the *enhancement bitstream* contains the phase parameters and information for finer quantization of frequency and envelope parameters.

For each frame of the input signal, a constant number of bits according to the desired bit rate is transmitted. The first bit in each frame is the *envelope bit*, which indicates whether envelope parameters are used or not. If this bit is set, the three envelope parameters t_{max}, r_{atk}, r_{dec} follow, and for each transmitted line an additional *line envelope bit* is transmitted, which indicates, whether a constant amplitude or the envelope has to be used for the synthesis of the corresponding line.

Since the human auditory system is not very sensitive to phase changes only the frequency and amplitude information of the spectral lines is coded and transmitted in the *basic bitstream* for obtaining a signal with the basic audio quality. But in this case it is necessary to provide information for the decoder which enables it to generate a signal free of phase discontinuities at frame boundaries. Therefore the first processing stage detects lines which continue from one frame to another. If a line is to be continued from the previous frame, only the frequency and amplitude changes are quantized and transmitted instead of the absolute frequency and amplitude values. For this purpose the frequency and amplitude parameters of the *i*-th line in the current frame *m* are compared with those of the *k*-th line in the previous frame *m*-1 for all possible combinations of *i* and *k*. The line continuation is used, if the relative frequency change

$$q_f(i,k) = \frac{\left| f_{i,s}(m) - f_{k,e}(m-1) \right|}{f_{i,s}(m)}$$

does not exceed a given threshold $q_{f,max}$ and if the ratio of amplitudes

$$q_a(i,k) = \begin{cases} a_i(m)/a_k(m-1) & \text{if } a_i(m) \geq a_k(m-1) \\ a_k(m-1)/a_i(m) & \text{if } a_i(m) < a_k(m-1) \end{cases}$$

lies within the interval $[1...q_{a,max}]$. If there is more than one possibility to continue a line from the previous frame, that line in the previous frame is selected, for which the following similarity criterion reaches its maximum:

$$Q = \frac{q_{f,max} - q_f(i,k)}{q_{f,max}} \cdot \frac{q_{a,\max} - q_a(i,k)}{(q_{a,\max} - 1)\, q_a(i,k)}$$

For each line, a *continuation bit* is transmitted in the bitstream, which indicates whether the line is continued from the previous frame or not. If this bit is set, the relative frequency change is transmitted after uniform quantization and the amplitude ratio is transmitted after quantization with a logarithmic characteristic. If the continuation bit is not set, the absolute frequency and amplitude values are both quantized with logarithmic characteristic.

Since the transmission of absolute frequency and amplitude values requires more bits per line than for the relative values, the number of lines transmitted per frame varies, if the bit rate for the *basic bitstream* is kept constant over the frames. For a frame length of 32 ms and a bit rate of 6 kbit/s with the selected quantizer stepsizes each frame contains the parameters of 10 to 17 lines.

Since the *basic bitstream* does not contain phase information, it is not useful to calculate a residual error signal by subtracting the corresponding decoder output signal from the input signal. In order to enable scalability modes, in which the residual signal is transmitted in a bitstream of higher bit rate, an additional *enhancement bitstream* is generated. It is constructed the following way:

- if the envelope parameters t_{max}, r_{atk}, r_{dec} are transmitted in the *basic bitstream*, additional bits for finer quantization of these parameters are transmitted
- if a line is starting, i.e. not continued from the previous frame, and its frequency exceeds a given threshold, additional bits for finer quantization of the absolute frequency are transmitted
- for each line the phase parameter is transmitted after uniform quantization

The number of bits per frame in the *enhancement bitstream* can vary, this has to be taken into account in the calculation of available bits for the coding of the residual error.

Since the Analysis/Synthesis Audio Codec allows to encode spectral lines that can be continued from the previous frame more efficiently than new lines having no predecessor, a higher bit rate would be required for rapidly changing signals than for slowly changing signals. For many signals, a rapid change - like for example the beginning of a new tone in a music signal - is followed by a longer period with only slow changes.

To improve the encoding of such signals, a dynamic buffer for the bitstream is introduced. When the required bit rate is low due to a slowly changing signal, bits are saved in a bit reservoir. If a rapidly changing signal requires more bits per frame than available at the fixed bit rate, the bits saved in the bit reservoir are used.

The access to the buffer is controlled by the number of lines to be transmitted. If this number exceeds a given upper threshold the buffer is filled with the remaining bits for the current frame. If the number of lines which can be transmitted with the bits of the current frame is below a lower threshold, additional bits are taken out of the buffer, if available. This way the sound quality for attacks in the audio signal is improved.

3 Decoder

To decode the bitstream generated by the analysis/synthesis encoder, two different decoders are available. The *basic decoder* uses only the *basic bitstream* containing the quantized frequency and amplitude parameters of the spectral lines. Phase relationships are not preserved but the *basic decoder* generates an output signal free of phase discontinuities. The *enhanced decoder* uses both the basic bitstream and the *enhancement bitstream* in order to produce an output signal which can be used for applications providing functionalities like bit rate scalability. Both implementations are described in the following.

3.1 Basic Decoder

For each frame this decoder first checks the *envelope bit* and if it is set, reconstructs the 3 envelope parameters, from which the envelope function is derived. After that the spectral lines are processed, until the number of remaining bits for the frame is less than the minimum number of bits needed for the line parameters. If the bitstream buffer is used, the number of processed lines is determined by the buffer control mechanism. The lines are processed as described in the following.

First the *continuation bit* and, if the *envelope bit* is set, the *line envelope bit* are read. Depending on the content of the *continuation bit*, the line parameters are processed:

- If the *continuation bit* is set, the relative frequency change and the amplitude change factor are decoded. The synthesis for this line is carried out by generating a sine wave with a linear transition from the frequency in the previous frame to the new frequency and with a continuous transition from the amplitude in the previous frame to the new amplitude. The phase at the beginning of the frame is taken from a buffer, in which the end phase in the previous frame is stored.
- If the *continuation bit* is not set, the absolute frequency and the absolute amplitude are decoded. A sine wave is synthesized, which has a constant frequency and which is faded in

from zero to the desired amplitude according to a given line start window function. In the same way, a line not continued from the previous frame is faded out with a given line end window function. In the current implementation, the line start and line end windows have the length of one frame. The line start window equals the first half of a Hanning window while the line end window equals the second half of a Hanning window.

In both cases the line is additionally multiplied with the envelope function if the line envelope bit is set. Furthermore the values of frequency, amplitude and phase at the end of the frame are stored in a buffer for continuation or fade-out in the next frame. All transition times (frequency and amplitude continuation, fade-in, fade-out) usually are equal to the frame length, but in the case of lines which use the envelope, the transition times are reduced by a given factor.

Due to the phase continuation of this decoder implementation, the speed of the decoded signal can be changed by simply changing the frame length for the synthesis without any other modifications. The relation of the encoder frame length and the selected decoder frame length directly corresponds to a speed change factor.

In a similar way, the pitch of the decoded signal can be varied without causing phase discontinuities. The pitch change is performed by simply multiplying each frequency parameter with a pitch change factor before it is used in the synthesis.

3.2 Enhanced Decoder

This decoder generates an output signal in which the spectral lines have phases corresponding to the phase of the input signal at the specified frequencies. Furthermore the envelope is reconstructed more accurately and the frequencies at the beginnings of new lines have finer resolution than those of the *Basic Decoder*. This is achieved by including the *enhancement bitstream* in the decoding process.

At the beginning of the frame this decoder first checks the *envelope bit* in the *basic bitstream* and if it is set, reconstructs the 3 envelope parameters from the corresponding bits in the *basic bitstream* and the additional bits for the envelope parameters in the *enhancement bitstream*. With the help of these parameters the envelope function is derived. The number of spectral lines which are processed after that is determined in the same way as by the *Basic Decoder*. The lines are processed as described in the following.

First the *continuation bit* and, if the *envelope bit* is set, the *line envelope bit* are read from the *basic bitstream*. Depending on the content of the *continuation bit*, the line parameters are read in:

- If the *continuation bit* is set, the relative frequency change and the amplitude change factor are decoded from the bits in the *basic bitstream*, and the phase is decoded from the corresponding bits in the *enhancement bitstream*.
- If the *continuation bit* is not set, the absolute frequency and the absolute amplitude are decoded from the bits in the *basic bitstream*. If the frequency bits in the *basic bitstream* indicate a frequency exceeding a given threshold, additional bits for a finer resolution of the absolute frequency are taken from the *enhancement bitstream*. After that the phase is decoded from the corresponding bits in the *enhancement bitstream*.

A sine wave is synthesized, which has a constant frequency and amplitude and the decoded phase at the beginning of the frame. The synthesized signal is additionally multiplied with the envelope function if the *line envelope bit* is set. The synthesized signal is then windowed using a Hanning window with a length of two frames.

Finally an overlap-add is used to combine the first half of the windowed synthesized signal of the current frame with the second half of the windowed synthesized signal of the previous frame. Thus the output signal is obtained. The values of frequency and amplitude at the end of the frame are stored in a buffer for continuation of spectral lines in the next frame.

4 Results

In order to evaluate the performance of the codec, the results of the *Analysis/Synthesis Loop*, which is the core of the whole system, have to be examined. Figure 8 shows for one frame of the input signal „male speech“ the short-time spectrum of the input signal, the spectrum of the signal synthesized with 13 spectral lines and the masked threshold calculated from the synthesized signal, which is used for the selection of the next significant spectral line. These results show that the parameters for the selected components properly represent the corresponding spectral components of the input signal, but that the number of spectral lines is not sufficient to represent the whole input signal without audible artifacts. For this type of input signals, an extension towards combined coding of spectral lines with frequencies which are multiples of one common fundamental frequency (i.e. harmonic structure) is very important.

In order to evaluate the frame-to-frame behavior of the system, the determined frequencies are shown for excerpts of the two input signals „vocal“ (Figure 9) and „haydn“ (Figure 10). Additionally the line continuation of the *Quantization and Coding* unit is visualized by connected lines for the corresponding frames. The first signal consists of a female voice singing a cappella. The first part (2s ... 2.35s) contains unvoiced portions, which are approximated by spectral lines which don’t form a harmonic structure and which are not continued from frame to frame. The second part contains voiced sound with a fast pitch change at about 2.75s. The estimation of non-constant frequencies is very important for properly reproducing this kind of input signals. The second signal contains a solo trumpet and background orchestra, i.e. many instruments with different fundamental frequencies. For this type of input signals the independent analysis for the individual spectral lines is essential.

Recently a previous implementation of the presented Analysis/Synthesis Audio Codec [2] with a less sophisticated envelope parameter generation, with frequency estimation only for spectral lines with constant frequencies and without the dynamic bitstream buffer took part in a subjective test within the MPEG-4 standardization activities of ISO/IEC [7]. Compared with the other proposals which were mainly speech codecs at a bit rate of 6 and 16 kbit/s the ASAC showed advantages over speech codecs for input signals mainly consisting of tonal components which are not necessarily harmonics of one fundamental frequency. In the case of speech signals the specialized speech codecs still had clear advantages, since they are optimally adapted to the signal source, including e.g. special provisions for unvoiced portions. For single instruments, the quality obtained from ASAC performed comparable to the best available speech codecs. A comparison of the improved implementation [3] described in this paper with the implementation according to [2] by experts showed clear improvements for some of the input signals [8].

5 Conclusions

The concept of the Analysis/Synthesis Audio Codec was presented and a first implementation based on the synthesis of individual spectral lines was described in detail. An analysis of the results obtained with this implementation showed its advantages and disadvantages in comparison with specialized speech codecs. Therefore the future work will focus on extensions to synthesis methods which allow the generation of harmonic spectra and of noise-type signals in order to improve the quality for speech and single instrument signals.

6 References

[1] ISO/IEC JTC1/SC29/WG11, „MPEG-4 Testing and Evaluation Procedures Document“, ISO/IEC JTC1/SC29/WG11/N999, Tokyo, July 1995.

[2] B. Edler, „Technical Description of the MPEG-4 Audio Coding Proposal from University of Hannover and Deutsche Bundespost Telekom“, ISO/IEC JTC1/SC29/WG11 MPEG95/M0414, October 1995.

[3] B. Edler, H. Purnhagen, „Technical Description of the MPEG-4 Audio Coding Proposal from University of Hannover and Deutsche Telekom AG“, ISO/IEC JTC1/SC29/WG11 MPEG96/M0632, January 1996.

[4] R. J. McAulay, T. F. Quatieri, „Speech Analysis/Synthesis Based on a Sinusoidal Representation“, IEEE Trans. on ASSP, Vol. 34, No. 4, August 1986.

[5] S. Kay, „A Fast and Accurate Single Frequency Estimator“, IEEE Trans. on ASSP, Vol. 37, No. 12, December 1989.

[6] F. Baumgarte, C. Ferekidis, and H. Fuchs, „A Nonlinear Psychoacoustic Model Applied to the ISO MPEG Layer 3 Coder“, Preprint 4087, 99th AES Convention, New York, October 1995.

[7] ISO/IEC JTC1/SC29/WG11, „MPEG-4 Audio Test Results (MOS Tests)“, ISO/IEC JTC1/SC29/WG11/N1144, München, January 1996.

[8] ISO/IEC JTC1/SC29/WG11, „Report of the Ad Hoc Group on the Evaluation of New Audio Submissions to MPEG-4“, ISO/IEC JTC1/SC29/WG11/MPEG96/M0680, München, January 1996.

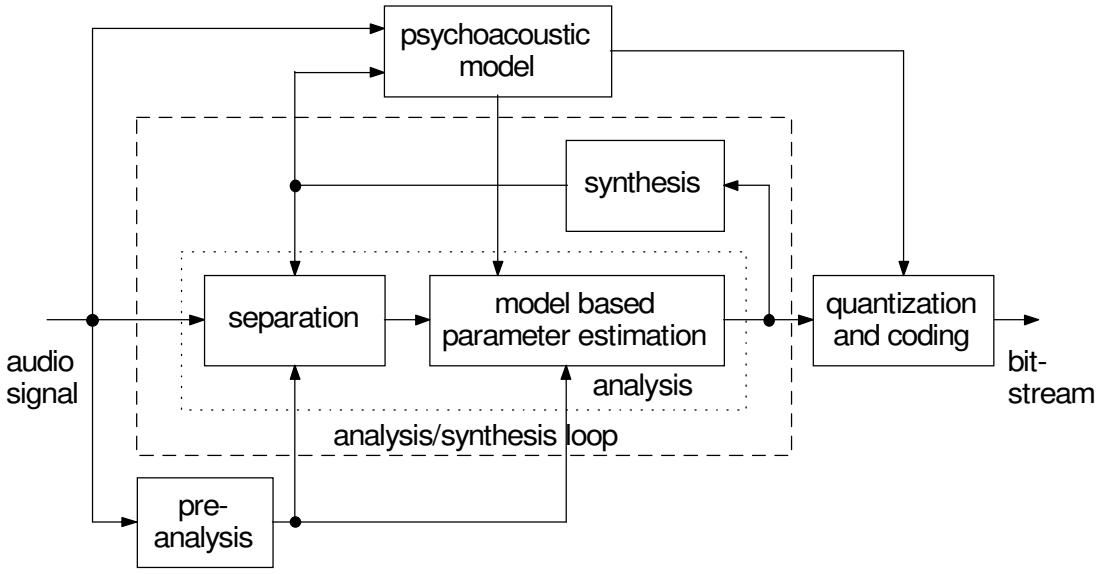

Figure 1: Block diagram of the Analysis/Synthesis Audio Codec.

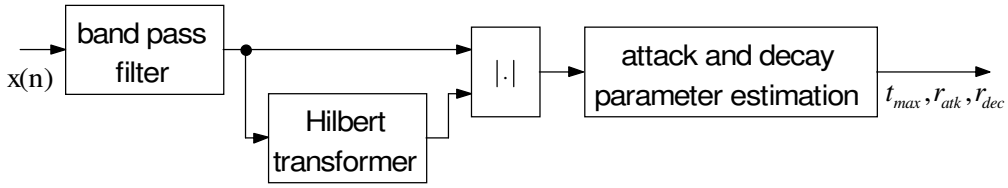

Figure 2: Block diagram for envelope parameter estimation.

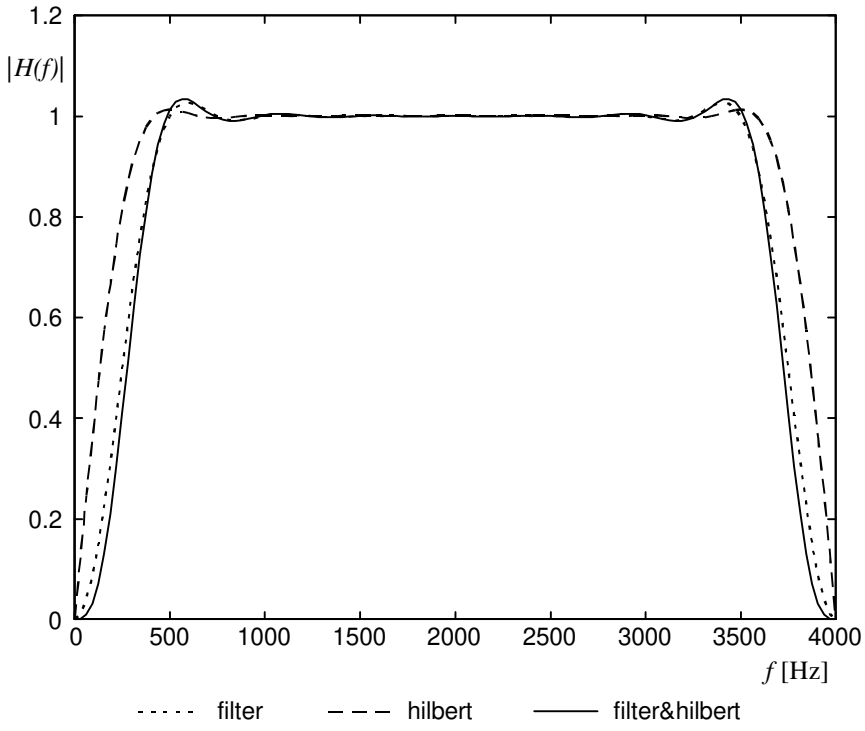

Figure 3: Bandpass filter and Hilbert transformer frequency response (8 kHz sampling).

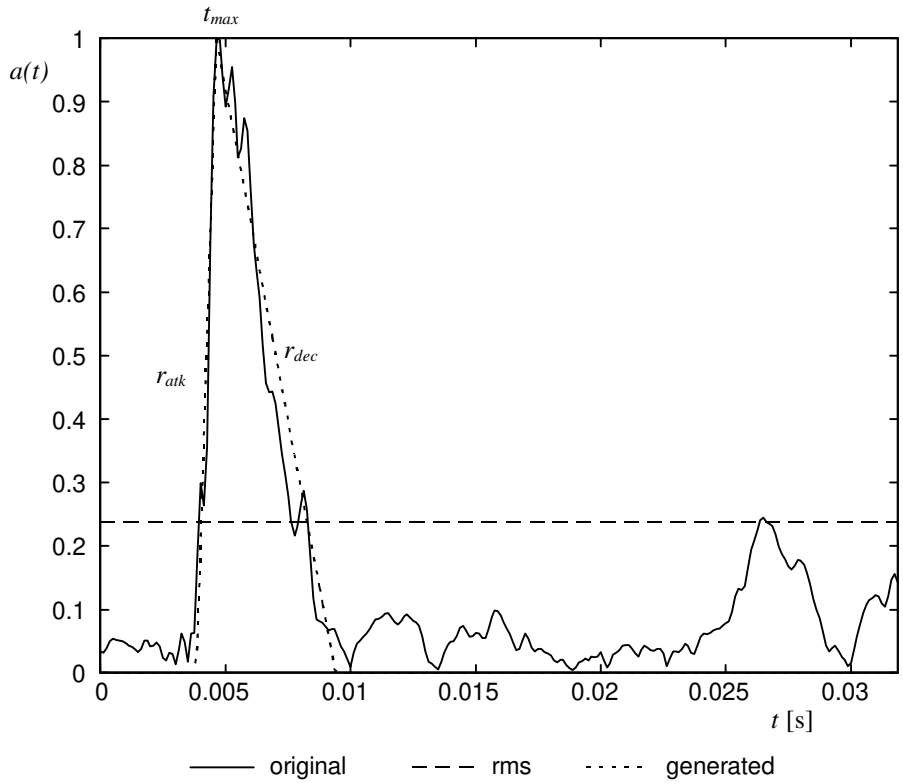

Figure 4: Envelope *a(t)* generated by the Pre-Analysis for one frame of the signal „castanets“.

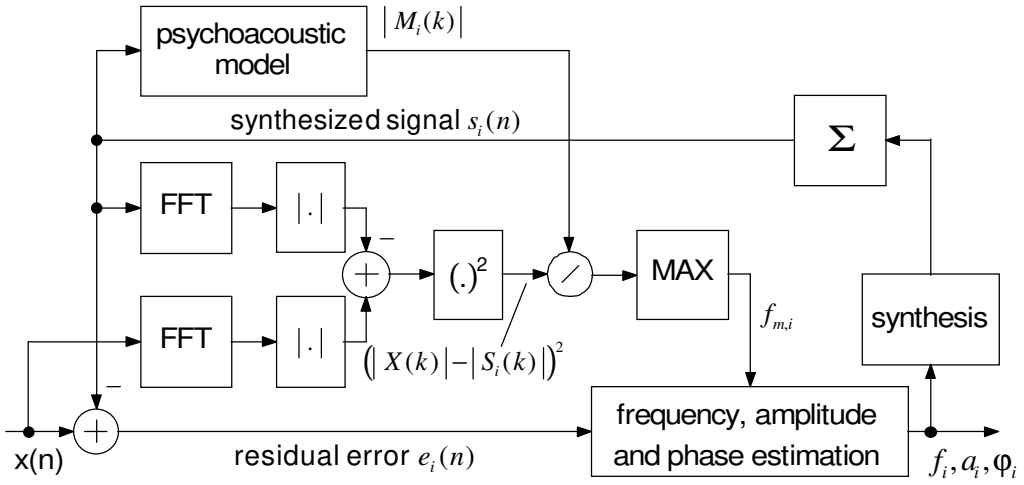

Figure 5: Analysis/Synthesis Loop based on the synthesis method „single spectral lines“.

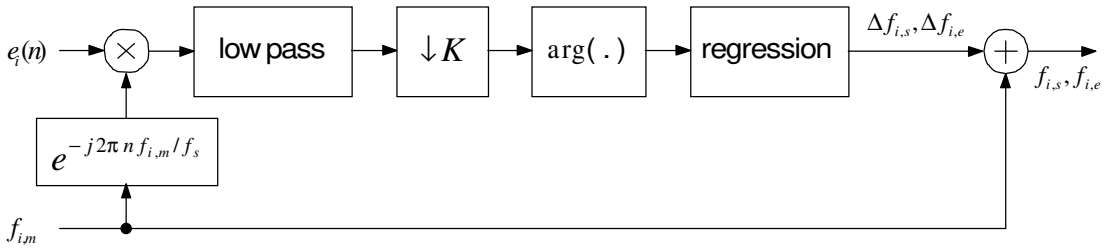

Figure 6: High accuracy estimation of linearly changing frequencies.

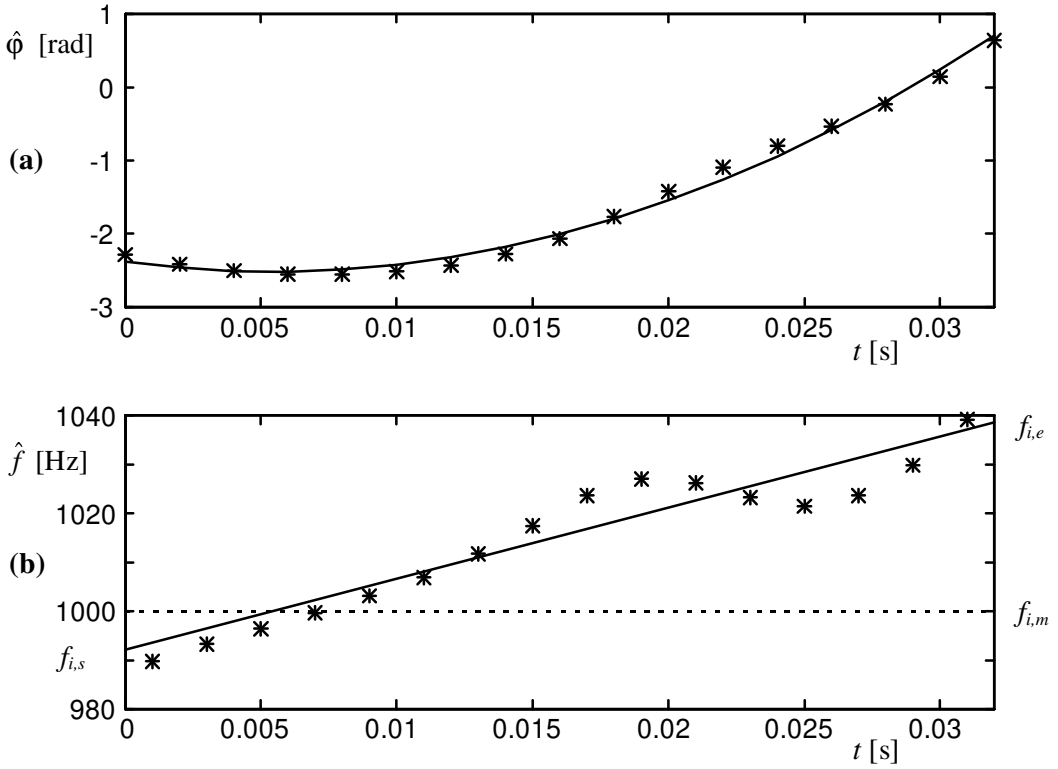

Figure 7: Second order phase regression (a) for estimation of linearly changing frequencies (b).

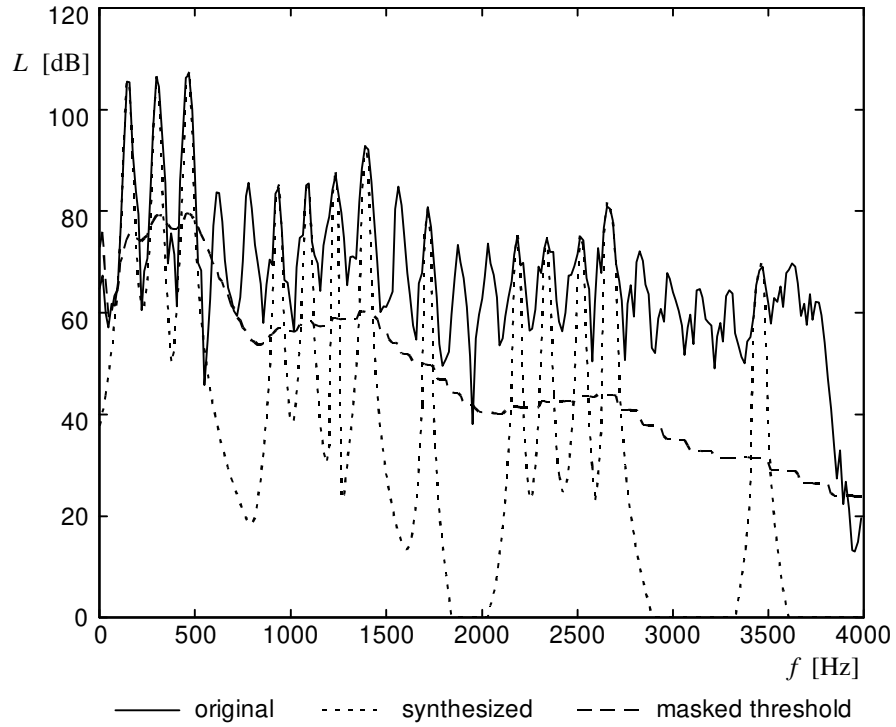

Figure 8: Spectra of the original signal and synthesized signal and the masked threshold calculated from the synthesized signal for one frame of the signal „male speech“.

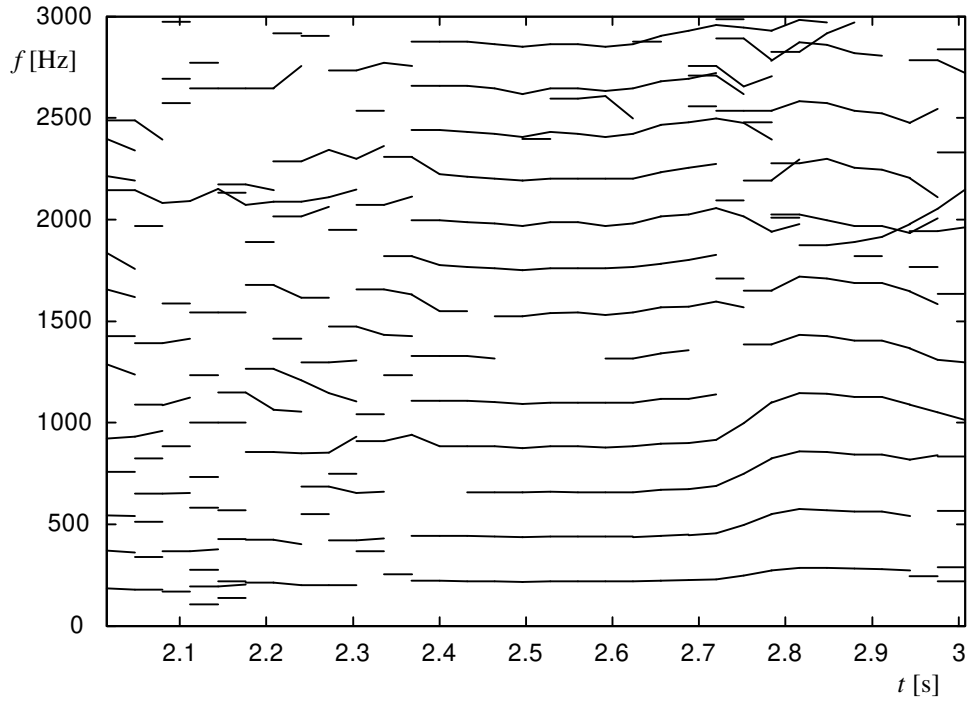

Figure 9: Spectral line pattern for the signal „vocal“.

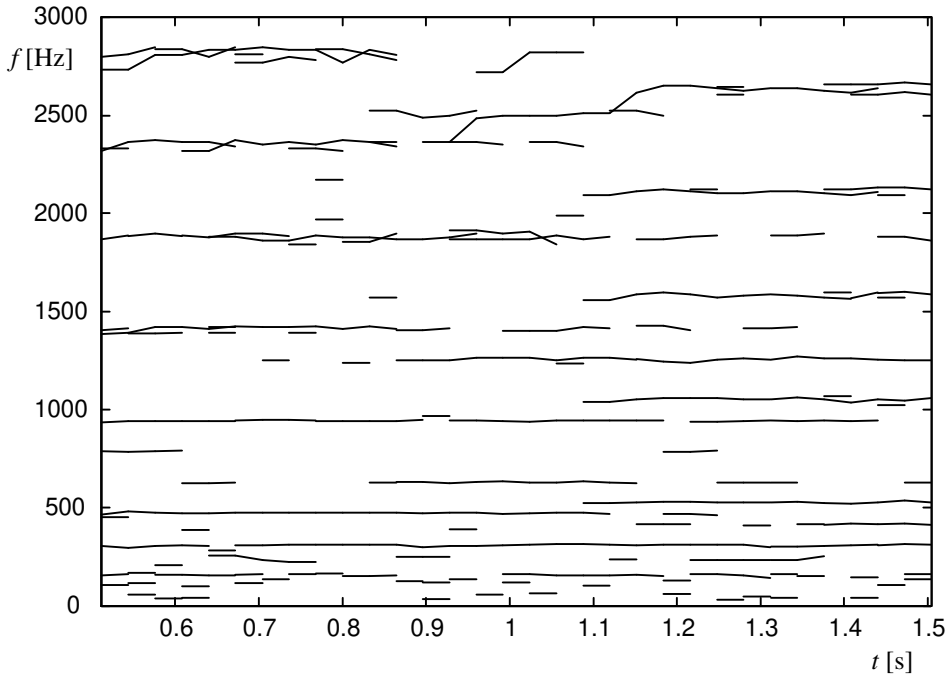

Figure 10: Spectral line pattern for the signal „haydn“.