A NARROWBAND LOW BIT RATE SINUSOIDAL AUDIO AND SPEECH CODER

Gerard Hotho and Rob Sluijter

Philips Research Laboratories
Prof. Holstlaan 4 (WO 2), 5656 AA Eindhoven, The Netherlands
gerard.hotho@philips.com

ABSTRACT

Speech and audio coders use different coding strategies. As a result, broadband (22.05 kHz bandwidth) audio coders typically have a good quality for both audio and speech signals and a high bit rate, whilst narrowband (4 kHz bandwidth) speech coders have a low bit rate but a distorted quality for audio signals. Our aim is to develop a single narrowband low bit rate audio and speech coder that provides speech quality on par with current state-of-the-art speech coders and good audio quality. To this end, the coder uses a mix of typical speech and audio coding strategies. A prototype coder was implemented and compared to the GSM-EFR standard speech coder. For speech signals, the attained quality approaches that of the GSM-EFR coder, whilst for music it performs clearly better. These results raise the expectation that more transparent coders are feasible for future mobile telephony.

1. INTRODUCTION

We developed a narrowband sinusoidal audio and speech coder operating at a bit rate of 12.2 kbit/s. The work is partly based on ideas and results from broadband sinusoidal coders [1,2]. In these coders, the input signal is typically split into three types of objects: transients, sinusoids and noise. Transients describe those parts of a signal that are non-stationary within a segment and are typically associated with a fast increase in power. Sinusoids are used to model the tonal parts of a signal, and noise is employed to model the stochastic parts. The stochastic parts can also be modelled with sinusoids, however, and it is hypothesised that due to the narrowband aspect of our signal this is a cost-effective approach. Hence, the number of objects for the narrowband coder is limited to two: transients and sinusoids. Transients are modelled by sinusoids as well, albeit by an increased number, and their starting positions are used in the coder analysis.

Section 2 describes both the encoder and the decoder. Section 3 gives the results of a listening test and in Section 4 our conclusions are presented.

2. CODER

The coder comprises an encoder and a decoder. The encoder produces parameters that represent a certain time interval of the input signal. This time interval is called a frame. For each frame, these parameters are estimated from segments of the input signal. These segments can extend outside the frame. For the jth frame, the sinusoidal signal model approximates a segment $x_j[n]$ of the input signal $x[n]$ by $\tilde{x}_{L_j,j}[n]$ which consists of L_j sinusoids according to

$$\tilde{x}_{L_j,j}[n] = \sum_{i=1}^{L_j} A_{i,j} \cos\left(\omega_{i,j} n + \varphi_{i,j}\right), \qquad n \in 0,1,\ldots,N_j - 1, \quad (1)$$

where the angular frequencies $\omega_{i,j}$, the amplitudes $A_{i,j}$ and the phases $\varphi_{i,j}$ are the sinusoidal parameters and N_j is the length of the segment. The sinusoidal parameters are estimated in a parameter estimator. These parameters are then both quantised and encoded in a parameter encoder. The decoder uses the output parameters of that parameter encoder to produce an output signal for each frame. Successive output signals are then combined to form the decoder output signal, which is a perceptual approximation of the encoder input signal.

The coder aims at two important objectives: a low bit rate and a good quality for speech signals. Bit rate reduction is primarily obtained by decreasing the update rate of the coder whenever the signal is stationary and by using Adaptive Differential PCM (ADPCM) coding of the sinusoidal parameters. Improvement of the speech quality is mainly achieved by coding the unwrapped phase instead of the frequency and by using time warping [3] in both parameter estimation and linking.

2.1 Parameter estimation

The strategy of the parameter estimation is depicted in Fig. 1. Transient detector TD determines the starting positions of transients x_t in the input signal $x[n]$ on the basis of energy increase. The parameter estimation is first explained for the regular case, i.e. when no transient is present in a frame. Data segmenting unit SD_1 segments the signal $x[n]$ for each frame. It produces a set of segments, x_s, of 60 and 100 ms, whose centres coincide with the current frame centre. The segments are used to determine both the pitch π and the warping parameter w [3]. The warping parameter is a measure for the rate of change of the frequencies, and therefore of the pitch, within these segments. Data segmentation unit SD_2 determines a set of segments, x_{2s}, of length 20, 40 and 60 ms, whose centres coincide with the current frame centre. The parameters of the sinusoidal model, indicated by p, are estimated in SP on the basis of these segments. For estimating amplitudes and phases, the 20 ms segment is used. For estimating the frequencies, either the 40 or the 60 ms segment is selected, depending on the pitch. The choice of segment length is such that at least 2 pitch periods are present in the segment. When the warping parameter w is different from 0, the segment for frequency estimation is warped, i.e. its pitch variation is removed in order to improve the frequency estimation [3]. Warping of segments introduces aliasing at high frequencies. To prevent aliasing of relevant signal information, SD_2 takes the segments from $x_2[n]$, which is the input signal $x[n]$ upsampled by a factor 2. Upsampling is performed by block US. Side information, denoted by s, that includes the transient positions, the update rate and the warping parameter, is also composed in SP. It is used by the parameter encoder.

The analysis update interval is 10 ms, except in the vicinity of a transient. When the start of a transient is detected in between 10 and

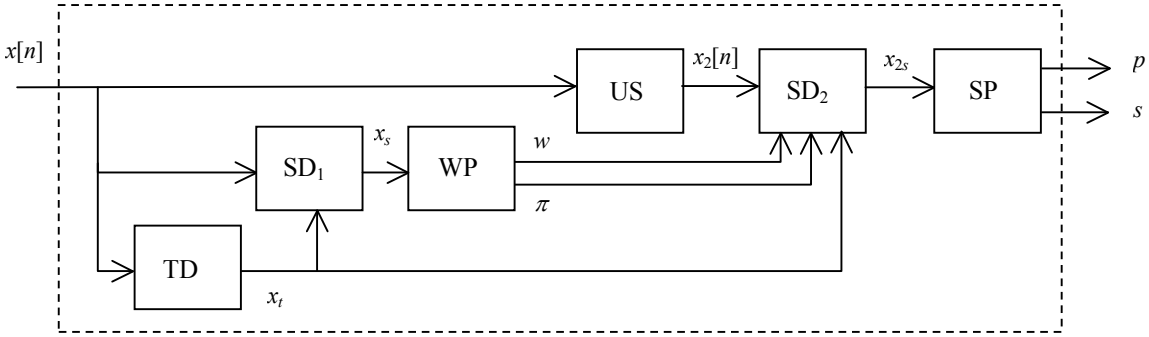

Fig. 1 Parameter estimation module. For details is referred to the text.

20 ms ahead of the current frame centre, the update interval is changed and the analysis windows of the current and next frame are adapted. For the current frame, the 20 ms segment is extended up to the transient. This segment is indicated by s_1. In the next frame a 20 ms segment, denoted by s_2, is taken that begins at the location of the transient. The transient is always positioned at the beginning of a segment, because if it were to occur somewhere in the middle of a segment, the decoder would treat the estimated sinusoids erroneously as if they were present during the entire segment, thus causing perceptually annoying pre-echoes. Both s_1 and s_2 are used to estimate all sinusoidal parameters p. For data segmentation unit SD_1, transient positions x_t only influence the segment centres, such that they coincide with the segment centres of SD_2.

2.1.1. Frequency estimation

The frequency estimation starts with windowing. All regular segments are windowed by Hamming windows. In the case of a transient, only half Hamming windows are applied and the region around the transient remains unwindowed. The top plot of Fig. 2 shows an example of the successive windows $w_j[n]$ in such a case. Subsequently, zeroes are appended to the windowed segment up to 2048 samples and the power spectrum is computed using a Discrete Fourier Transform (DFT). Finally, L_j frequencies (up to 50) are determined by detecting local maxima in the power spectrum, with priority according to magnitude.

In the case of a transient, the two segments around x_t will be approximated more accurately to compensate for the spectral effects of the rectangular parts of the windows, the extension of the first segment and to enable adequate modelling of fast decays of the signal in the second segment. Fast decays typically occur in the second segment, for instance in the sound of clattering castanets. Enhanced accuracy for these segments is obtained by allocating another set of Λ_j (up to 50) frequencies to the residual segment

$$r_{L_j,j}[n] = x_j[n] - \tilde{x}_{L_j,j}[n], \tag{2}$$

in a way similar to the allocation of the first L_j frequencies. In the above equation $\tilde{x}_{L_j,j}[n]$ is given by Eq. 1 and $x_j[n]$ denotes the segment that is selected for frequency estimation.

2.1.2. Amplitude and phase estimation

The amplitudes $A_{i,j}$ and phases $\varphi_{i,j}$ are *not* taken from the power spectrum at the detected frequencies $\omega_{i,j}$, but they are computed so as to minimise

$$E_{i,j} = \sum_{n=0}^{N_j-1} \left\{ r_{i-1,j}[n] - A_{i,j} \cos\left(\omega_{i,j} n + \varphi_{i,j}\right) \right\}^2 w_j[n], \tag{3}$$

with

$$r_{i-1,j}[n] = x_j[n] - \tilde{x}_{i-1,j}[n], \tag{4}$$

where i=1,2,...L_j, $x_j[n]$ denotes the selected segment for amplitude and phase computation and $w_j[n]$ is the applied window. In the case of a transient, $w_j[n]$ is equal to the window that is used for the frequency estimation. In a regular case, $w_j[n]$ is a Hamming window. In the case of a transient when a second set of Λ_j frequencies is determined, as described in the previous section, i proceeds in the second iteration according to $i=L_j+1, L_j+2,..., L_j+\Lambda_j$. Prior to each iteration the frequencies $\omega_{i,j}$ are ordered according to their magnitudes from the DFT power spectrum, with the largest first. This approach gives rise to the relations

$$A_{i,j} = \frac{2\left|X\left(\omega_{i,j}\right)\right|}{\sum_{n=0}^{N_j-1} w_j[n]}, \quad \varphi_{i,j} = \arg\left(X\left(\omega_{i,j}\right)\right), \tag{5}$$

where $X(\omega_{i,j})$ is given by the DFT

$$X\left(\omega_{i,j}\right) = \sum_{n=0}^{N_j-1} r_{i-1,j}[n] w_j[n] \exp\left(-j\omega_{i,j} n\right). \tag{6}$$

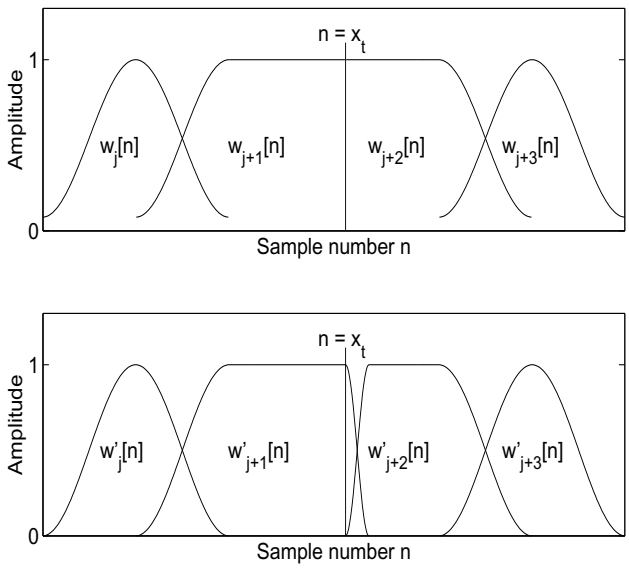

Fig. 2 Example of windows around transient position x_t. Top plot: encoder windows. Bottom plot: decoder windows.

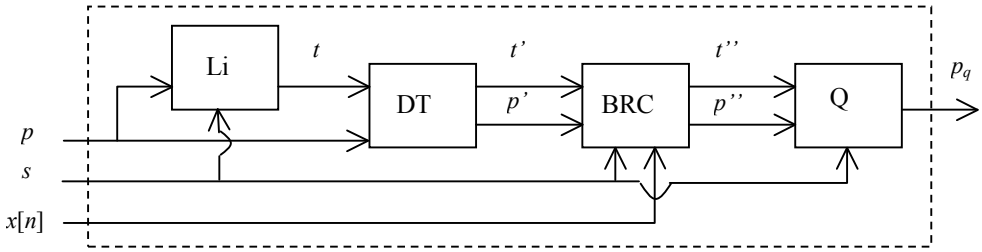

Fig. 3 Parameter encoding producing quantised parameters. It consists of a block that links sinusoids of consecutive frames into tracks (Li), a block that discards short tracks (DT), a bit rate control (BRC) and a parameter encoder and quantiser (Q).

2.1.3. Variable update rate

A post-processing analysis routine is used to decrease the update rate, and therefore the bit rate, of the coder in the case of a stationary input signal. Stationarity of the input signal is computed on the basis of the linking between two consecutive frames (for linking see Section 2.2.1.). If the stationarity of the input signal exceeds a fixed threshold, the two associated frames are combined and the sinusoidal parameters are recomputed. Segments of 40 and 60 ms are used to compute the frequencies as described in Section 2.1.1. A segment of 30 ms is used to compute both amplitudes and phases as described in Section 2.1.2. These parameters replace the parameters that were computed for the two frames. Frames can be combined only once on the basis of stationarity. The update interval between two stationary frames amounts to 20 ms.

2.2. Parameter encoding

The successive steps in the parameter encoding process are shown in Fig. 3. Input are the estimated sinusoidal parameters p, input signal $x[n]$ and side information s. Output of the process are the quantised and encoded sinusoidal parameters p_q.

2.2.1. Linking

The first step of parameter encoding consists of linking, in which continuing tones in the signal are tracked by connecting frequencies of consecutive segments [1,2]. In this process, candidate connections are assessed on the basis of frequency differences on an ERB scale, amplitude ratios and, for low-frequencies, matching of the phases. The frequency difference is computed by forward predicting the frequency of the first frame of a connection and backward predicting the frequency of the second frame of that same connection, to the position halfway between the centres of the two frames using the pertinent warping parameters. The resulting tracks t are used by all other steps of the parameter encoder.

2.2.2. Discarding short tracks

Discarding of short tracks is a two-step process. In the first step, tracks of length 1 and 2 are discarded, whose amplitudes do not exceed a certain threshold value. The second step removes those sinusoids that belong to a track of length 1 or 2, which comply with two criteria: they do not have the highest amplitude within their critical band and their energy is below the average energy of all sinusoids in their frame. The entire process removes about 83% and 78% of the tracks of length 1 and 2 respectively, without deterioration of the sound quality. Because most tracks are short and short tracks are relatively expensive to encode, as described in Section 2.2.4, a considerable reduction of the bit rate is obtained.

2.2.3. Bit rate control

Bit rate control is a mechanism to ensure that a pre-set bit rate is not surpassed. It is based on the psycho-acoustical model described in [4]. The model labels all track elements with a measure of perceptual relevance D. This measure is computed on the basis of the DFT of the segments that were selected for phase and amplitude computation. In the bit rate control mechanism, a certain threshold T is chosen and track elements for which D is less than T are discarded, if that reduces the bit rate. Different threshold values are searched to meet the bit rate requirement.

2.2.4. Quantisation

Prior to actual quantisation of the frequency, amplitude and phase parameters, the frequencies are mapped onto an ERB scale and the amplitudes onto a dB scale. In each frame both births and continuations of tracks can occur. The births of tracks are encoded separately from the continuations in a way as described in [2]. Along the frequency axis, the lowest frequency birth is uniformly quantised and entropy coded using a fixed table, and higher frequency births are quantised and coded differentially. The amplitude parameter associated with the lowest frequency is also quantised uniformly and entropy coded using a fixed table. The other amplitudes are ordered according to the frequency ordering and also differentially quantised. On average a birth costs about 6.5 bits for the frequency and about 3.5 bits for the amplitude. The continuations of a track are encoded differentially using ADPCM with a fixed 2nd-order predictor and a backward adaptive quantiser. It was found that 1-bit ADPCM is sufficient to encode the amplitudes of a track. Instead of encoding the frequency of a track and retrieving the phase from it in the decoder [1,2], the unwrapped phase is encoded with 2-bit ADPCM and the frequency is derived from it [5]. The major advantage of the last approach is that it preserves the waveform of the signal. This can be especially advantageous when encoding low-pitched voiced speech. Both the frequency- and phase-birth of each track are needed to initialise the 2nd-order predictor. The phase birth has a uniform distribution, and is therefore uniformly quantised. This is done using 3 bits.

2.3. Decoder

The decoder produces a sequence for frame j according to:

$$\hat{x}_j[n] = \sum_{i=0}^{L'_j-1} A'_{i,j} \cos\left(\omega'_{i,j} n + \varphi'_{i,j}\right), \quad n \in 0,1,\ldots,M_j - 1, \qquad (7)$$

where M_j is the length of the segment that is composed of the quantised frequencies $\omega'_{i,j}$, amplitudes $A'_{i,j}$ and phases $\varphi'_{i,j}$. L'_j stands for the number of sinusoids that are received in the jth frame. The ampli-

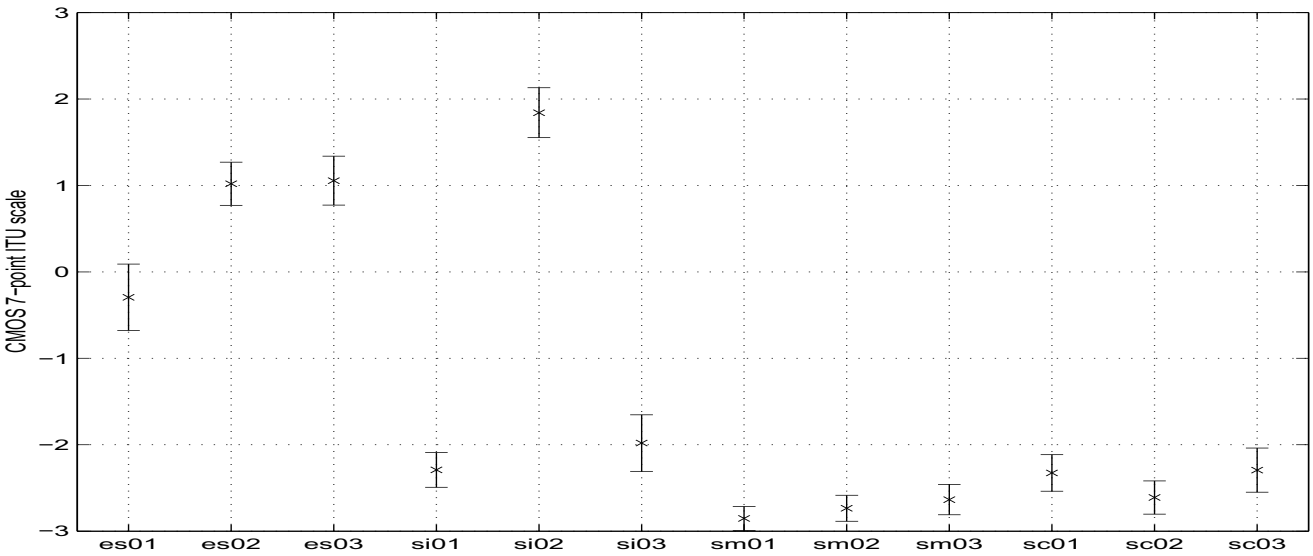

Fig. 4 Test results from the listening experiment. The crosses represent the mean CMOS scores for the MPEG excerpts. The bars indicate the 95% confidence interval. Negative values show a preference for the sinusoidal coder, positive values for the GSM-EFR coder.

tude and the unwrapped phase are computed by the ADPCM decoder. The frequency parameter is obtained by low-pass filtering the derivative w.r.t. time of the unwrapped phase [5].

The sequences of consecutive frames are windowed and combined using overlap add. For a normal and a stationary frame, windows of 20 and 30 ms are used respectively. Both windows start with a 10 ms long left half of a Hanning window and end with a 10 ms long right half of a Hanning window. Any sample in between is set to 1. In the case of a transient, however, different windows are applied. They are shown in the lower plot of Fig. 2. Unlike the window $w_{j+1}[n]$, the window $w'_{j+1}[n]$ does not immediately go to zero after the transient position x_t. A small overlap of 2.5 ms with the next window is created, to prevent discontinuities in the output signal. The windowed sequences of successive frames are aligned such that the right half Hanning window of the current frame and the left half Hanning window of the next frame add up to 1. Using overlap add, the sequences are combined to form the output signal.

3. LISTENING TEST

Our coder is compared to the GSM-EFR standard speech coder [6] at the same bit rate. The Comparison Mean Opinion Score (CMOS) test is used with the 7-point ITU comparison scale (attributes: 'A is much better than B', better, slightly better, equal, slightly worse, worse, much worse, with associated scores from +3 down to –3 respectively). MPEG-4 test signals [7] were used, consisting of three speech (es01, es02, es03) and nine music excerpts. The test involved 16 listeners. The test results are shown in Fig. 4. For music signals, the proposed coder is on average 'better' than the GSM-EFR coder (average CMOS score –1.99). Only si02 (castanets) is an exception to this average behaviour. For the speech fragments, the proposed coder performs on average in between 'equal' and 'slightly worse' than the GSM-EFR coder (average CMOS score 0.59).

4. CONCLUSIONS

A narrowband low bit rate sinusoidal audio and speech encoder and decoder have been described. A listening test shows that the proposed coder performs significantly better for most music signals than the GSM-EFR coder. For speech, our coder performs slightly worse than the GSM-EFR coder. These results raise the expectation that more transparent coders are feasible for mobile telephony. Future work should concentrate on the improvement of the coder for both speech and burst-like signals (castanets). Also the coder delay, which is larger for our coder than for the GSM-EFR coder, needs to be addressed.

5. REFERENCES

[1] B. Edler et al., "ASAC – Analysis/Synthesis Audio Coder for very low bit rates", 100th AES Convention, Copenhagen Denmark, preprint 4179 (F-6), May 11-14 1996.

[2] A.C. den Brinker, E.G.P. Schuijers and A.W.J. Oomen, "Parametric coding for high-quality audio", 112th AES convention, Munich Germany, Conv. paper 5554, May 10-13 2002.

[3] R.J. Sluijter and A.J.E.M. Janssen, "A time warper for speech signals", Proc. IEEE Workshop on Speech Coding, Parvoo Finland, June 20-23 1999, pp. 56-58.

[4] S. van de Par et al., "A new psychoacoustical masking model for audio coding applications" IEEE ICASSP, Orlando USA May 2002, pp. II-1805-1808.

[5] A.C. den Brinker et al., "Phase transmission in a sinusoidal audio and speech coder", 115th AES Convention, New York, Conv. paper 5983, 10-13 October 2003.

[6] "Enhanced Full Rate (EFR) Speech Transcoding", ETSI-GSM Recommendation 06.60.

[7] Audio Subgroup, "Call for proposals for new tools for audio coding", ISO/IEC JTC1/SC29/WG11 MPEG 2001/N3793 Jan. 2001.