International Symposium on Signal Processing and its Applications (ISSPA), Kuala Lumpur, Malaysia, 13 – 16 August, 2001. Organized by the Dept. of Microelectronics and Computer Engineering, UTM, Malaysia and Signal Processing Research Centre, QUT, Australia

SCALABLE AUDIO CODING EMPLOYING SORTED SINUSOIDAL PARAMETERS

M. Raad, I.S. Burnett and A. Mertins
School of Electrical, Computer and Telecommunications Engineering.
University of Wollongong, Northfields Ave Wollongong NSW 2522, Australia.
mr10@uow.edu.au

ABSTRACT

This paper describes the use of sorted sinusoidal parameters to produce a fixed rate, scalable, wideband audio coder. The sorting technique relies on the perceptual significance of the sinusoidal parameters. Sinusoidal coding permits the representation of a given signal through the summation of sinusoids. The parameters of the sinusoids (the amplitudes, phases and frequencies) are transmitted to allow the signal reconstruction. In the proposed scheme, the sinusoidal parameters are sorted according to energy content and perceptual significance. The most significant parameters are transmitted first allowing the use of only a small set of the parameters for signal reconstruction. The proposed scheme incurs a low delay and uses a 20ms frame length. The results presented show the advantages gained for scalable audio coding by sorting the parameters.

1. INTRODUCTION

Audio coding aims to reduce the bandwidth required for storage and transmission of a digitized audio signal. Nominally, single channel CD quality audio requires a transmission rate of 706 kb/s [6]. The reduction of this bit rate has been the impetus behind the introduction of standards such as MPEG-1, 2 and 4 as well as products such as the Dolby AC-2 and AC-3 systems for digital audio compression and transmission [6].

This paper focuses on reducing the bit rate required for the transmission of audio by using a different approach to the existing schemes, as well as allowing for better quality audio as more bandwidth is made available. Sinusoidal coding of the audio signal is combined with a sorting technique to allow better quality audio to be produced at low bit rates. The sorting emphasizes the perceptually significant signal frequency components.

This paper primarily aims at offering audio enhancements to wireless applications using fixed rate transmission, with higher rates offering higher quality. Thus the sinusoidal coding technique utilises a short, fixed, frame length, setting it aside from typical sinusoidal coders that use a variable frame length [1] [2]. The coder described also utilizes the entire 22.05 kHz bandwidth of the audio signal, unlike the existing HILN parametric coder [10].

2. SINUSOIDAL AUDIO CODING

Sinusoidal coding has been used to develop both speech [1] and audio (music) coders [2]. The following sections describe the general principles behind sinusoidal coding and the modifications made to the general model for the work of this paper.

2.1 The general model

The principle behind sinusoidal coding is the Fourier representation of a signal whereby a given signal is represented by its sinusoidal components such that:

$$s[n]=\sum_{\ell=1}^{L} A_\ell \cos(\omega_\ell n+\phi_\ell) \quad n=0,\ldots,N-1 \qquad (1)$$

where A_ℓ, ω_ℓ and ϕ_ℓ are the amplitudes, frequencies and phases of the sinusoidal components respectively and s is a selected frame, of length N, of the original audio signal. In the proceeding sections, the amplitudes, frequencies and phases will be regularly referred to as the parameters of the sinusoidal model.

There are two popular techniques for deriving the sinusoidal parameters: direct use of the Discrete Short Time Fourier Transform (DSTFT) [1] or the use of Analysis-By-Synthesis (ABS) [2]. Using the Fourier transform method, the sinusoidal parameters are determined from the DSTFT parameters such that:

$$A_\ell=\sqrt{(a_\ell^2+b_\ell^2)},\ \phi_\ell=\arctan\left(\frac{-b_\ell}{a_\ell}\right) \text{ and } \omega_\ell=\frac{2\pi\ell}{N} \qquad (2)$$

where a_ℓ and b_ℓ are the DSTFT coefficients and N is the total number of transform coefficients.

To allow the DSTFT of the signal to be performed, the signal is divided into frames by the use of a satisfactory window. Typically, window functions such as the Hamming or Hanning windows are used [1][2][3] and the frames are overlapped by up to half the given frame length. The length of the window is normally kept at 2.5 times the length of the pitch period of the signal. The pitch period is regularly estimated and updated, resulting in a variable length frame coder.

The reconstruction of the signal is performed by employing the overlap-add technique to avoid discontinuities at frame boundaries. That is, the synthesised signal is given by:

$$\hat{s}[n]=W_s[n]\hat{s}^{k-1}[n]+W_s[n-T]\hat{s}^k[n-T] \qquad (3)$$

where $\hat{s}^k$ is the synthesised k^{th} frame, W_s is the reconstruction window and T is the overlap of the consecutive frames (in samples) [1]. It has been shown that some performance gains may be obtained by ensuring that the analysis and synthesis windows are identical [3].

2.2 Model modifications

The general model described in Section 2.1 was modified slightly for the work described in this paper. Firstly, a short, fixed frame is used which is more fitting for a coder to be used for continuous transmission purposes. In choosing the frame length of the coder, a lead was taken from the GSM speech coder which utilises a 20ms frame length [7]. The use of a 20ms frame length means that the audio coder proposed is a short frame length coder. Frame selection is carried out by the use of overlapping windows. At a sampling frequency of 44.1 kHz (the sampling frequency of the CD) a 20ms frame corresponds to approximately 880 samples per frame.

0-7803-6703-0/01/$10.00©2001 IEEE

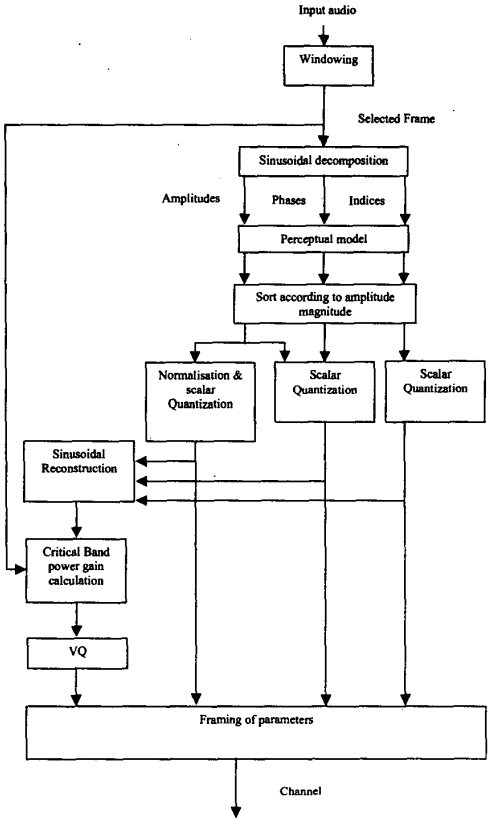

Figure 1 (a) Proposed encoder architecture.

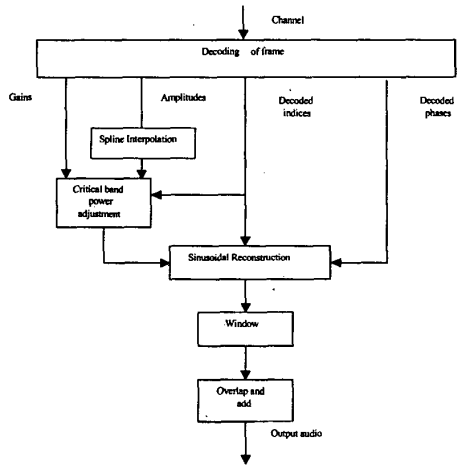

Figure 1(b) The proposed decoder

The window used in this case was obtained from [9] and is given by:

$$W[n] = \sin\left((n+0.5)\frac{\pi}{N}\right) \quad 0 \le n \le N-1 \qquad (4)$$

W is a perfect reconstruction window of length N and so is used for both analysis and synthesis. This window is also used because of its sharp main lobe and low side lobe frequency response. The narrow main lobe ensures minimal "smoothing" effects in the frequency domain and the low side lobes reduce the potential of "frequency leakage". The use of a perfect reconstruction window is consistent with such coders as AC-3 and MPEG-4 [6].

3. PROPOSED ARCHITECTURE

Figure 1 illustrates the proposed audio coder and decoder. The proposed architecture is built around the sorting of the amplitudes according to energy content. We thus consider the sorting technique first in this discussion.

3.1 Sorting the parameters

The amplitudes are sorted in order of decreasing magnitude, and the same sorting is used for the phases. This sorting allows the coder to concentrate on the parameters that contribute most to the reconstruction of the original signal. The process also has the added advantage of producing a monotonic relationship between the ordered amplitude magnitudes. This relationship permits the modeling of the amplitudes by the use of either a monotonic decreasing function or by the use of interpolation. A further advantage is that, having arranged the amplitudes in a manner that determines the amplitudes' relative (and perceptual) importance, scalable audio coding with a smooth increase (or decrease) in quality is a possibility.

Using the sorting scheme described, it was found, through informal listening tests, that only fifty sets of parameters (i.e. amplitudes, phases and indices) need be used (out of a complete 441 in the frame selected) to achieve good quality synthesised audio [5].

Simply sorting the amplitudes in terms of energy content does not take advantage of the perceptual redundancies in the signal. To improve transmission rates while maintaining quality a perceptual model is employed to remove the perceptually insignificant components from the model before the sorting of the amplitudes. In the following subsection we explore the use of simultaneous masking as a perceptual model for this selection process.

3.2 The perceptual model

Sorting the sinusoids according to energy content to determine the relative importance of each sinusoid is an effective technique when the signal to be coded is highly tonal. However, this technique will not perform as well for non-tonal signals as the energy content of non-tonal signals is distributed over a wider range of frequencies than tonal signals.

It is known from auditory theory that when numerous sounds reach the ear simultaneously, a number of them may be masked [6]. In this work, the masking effect is used to determine which frequency components are required (according to the original signal's frequency representation). In particular, we seek to eliminate a number of high-energy components in the same critical frequency band. This allows either a reduction in the number of components that need transmission or an increase in

sound quality using the same number of components. The technique used to calculate the simultaneous masking curve for each frame is the same as that presented by Johnston [8].

The application of the masking model differs from the technique used in, for example, MPEG-4 and AC-3 where the model determines a limit for the quantization noise [6]. Here, the perceptual model is being used in the selection of the frequency components to be transmitted first.

3.3 Frequency domain gains

As mentioned in the preceding two subsections, only a small number of sinusoids need be used from the complete set of sinusoids to produce the synthesised audio. A consequence of using so few sinusoids from the original set is the reduction of overall energy content in the signal. To counter significant energy loss, twenty-five frequency domain gains are used, each corresponding to a critical frequency band as given in [6]. These gains adjust the average energy of the synthesised signal in each critical band to ensure the energy in each band is approximately the same for the synthesised and original signal.

In practice, it was found that more weight should be applied to the low frequency gains than the high frequency gains. Without this weighting high frequency "scratches" may occur in the synthesised audio when the quantized gains are used. These scratches are undesirable and the weighting is carried out by applying a power law as described in [5].

4. QUANTIZATION

The sorting of the sinusoids according to energy content allows preferential transmission of the parameters as well as the exploitation of the relationship between consecutive amplitudes. Scalar quantization is used for the quantization of most of the parameters in this work. This may seem wasteful, but, as will be clarified, scalar quantization is necessary for some parameters and sufficiently efficient for the quantization of other parameters.

4.1 Quantization of the amplitudes

The monotonic relationship between successive sorted amplitudes is exploited by modeling the amplitudes with the use of a spline interpolator. This means that only a limited number of the amplitudes are transmitted, the rest are inferred at the decoder by spline interpolation of the transmitted amplitudes. Figure 2 shows actual sorted amplitudes and the spline model of the amplitudes. The number of amplitudes to be transmitted at any given rate has been determined on the basis of similar results to those presented in [5]. In the coder presented here at least one amplitude is inferred for every amplitude transmitted. The transmitted amplitudes are scalar quantized at 8 bits each.

4.2 Quantizing the phases

In sinusoidal speech coders, the phase tends to be modeled rather than quantized, as in [1]. A different approach taken by [2] was to maintain the time domain envelope of the signal which implicitly contains phase information. In our experiments we found the maintenance of phase information to be important to overall quality, a result confirmed by [1]. In the previous, variable rate, version of this coder, the phase was quantized using weighted scalar quantization, where the amplitudes were used as relative weights [5]. That is, phases associated with the large amplitudes are quantized more accurately than phases associated with the smaller amplitudes.

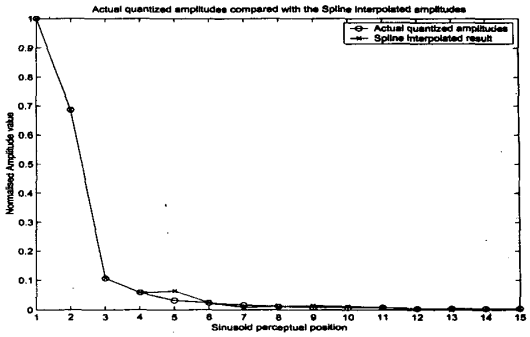

Figure 2 The use of spline interpolation (The transmitted amplitudes correspond to positions [1 2 3 4 6 8 11 15]).

The scalable version of the proposed coder used in this work utilised a uniform scalar quantizer for the phase to allow a simpler implementation of the coder. Each phase component is now coded using 5 bits. The number of bits used per phase was determined experimentally.

4.3 Quantizing the indices

The indices indicate the spectral position of the sinusoids before sorting. As the sinusoids have been re-arranged, the indices are required by the decoder for correct synthesis. To avoid distortion, the indices must be coded losslessly. In [5] the use of a Huffman code to quantize the indices led to a variable rate coder. For this work the aim was the use of a scalable coder at fixed bit rates, hence a fixed length loss-less code is used to represent each index. A total of 441 sinusoids are used; thus 9 bits per index are required.

4.4 Quantizing the gains

The vast majority of the critical band energy matching gains were found to be distributed between one and two. A ten bit codebook was trained to quantize the gains with each gain vector having twenty-five elements. The use of this codebook in combination with the weighting of the gains produced insignificant distortion on the synthesised audio.

4.5 Total number of bits

As a summary of the previous sub-sections it can be seen that the total number of bits per sinusoid is 22 (8 for the amplitude, 5 for the phase and 9 for the index). There are also 12 bits per frame used for the normalization of the amplitudes and 10 bits per frame used to quantize the frequency domain gains. Thus at a bit rate of 32 kb/s (for example) a total of thirteen sinusoids would be used, without spline interpolation. The use of spline interpolation allows this number to be extended to 15.

5. RESULTS

The proposed coder was used to code all of the SQAM files obtained from [4] at 16 kb/s, 32 kb/s, 42 kb/s, 64 kb/s and 128 kb/s. Informal listening tests showed that the improvement in quality as the bit rates were increased was easily appreciated by the listener. It is also significant to report that although the quality of the audio degraded significantly at 16 kb/s, all of the speech files included in the testing were found to be intelligible with slightly annoying distortion.

SIGNAL DESCRIPTION	16 KB/S	32 KB/S	42 KB/S	64 KB/S	128 KB/S	MASKIING SEGSNR
Bass	6.53	10.41	12.27	14.70	15.89	22.3
Glockenspiel	15.26	15.37	15.37	15.37	15.38	22.68
Harpsichord	4.68	6.28	6.83	7.57	8.44	10.53
Qaurtet	7.81	11.65	13.22	15.15	16.51	26.52
Eng F speech	8.40	10.28	10.82	11.36	12.11	20.45
Eng M speech	8.05	10.22	10.79	11.35	11.87	18.02
Horn	15.91	17.87	17.94	17.94	17.94	26.74

Table 1 SegSNR (dB) results at various bit rates and before coding (Masking SegSNR)

The music files coded at 16 kb/s varied between annoying and very annoying with the exception of highly tonal music such as that obtained from the Horn. The source of the most perceptually annoying distortion seems to be audibility of individual sinusoids when very few are used for the synthesis. As the bit rates used for the coding were increased the quality improved accordingly. Good quality was obtained for most files at the 32 and 42 kb/s rates. Table 1 lists selected Segmental Signal to Noise Ratio (SegSNR) results in dB, whilst Figure 3 plots some of those results.

Two points to note are the "Masking SegSNR" column in Table 1 and the general consistent shape of the curves in Figure 2. The "Masking SegSNR" column in Table 1 shows the SegSNR obtained using the complete set of sinusoids available after the masked components had been removed (before any quantization). At such SegSNR results, the synthesised audio was found to be indistinguishable from the original in informal listening tests. The values presented in the "Masking SegSNR" column are meant as a point of comparison and as an indicator of the relative quality of the synthesised audio. These objective results were found to be in agreement with informal listening test results.

The general shape of the curves in Figure 3 indicates the existence of a knee point in the relationship between quality obtained and bits used at around the 42 kb/s mark. The existence of such a knee point seems to indicate that this coder should be operated at around the 42 kb/s mark, however, the perceptual quality of the reconstructed audio was found to keep on improving as the bit rate was increased beyond 42 kb/s.

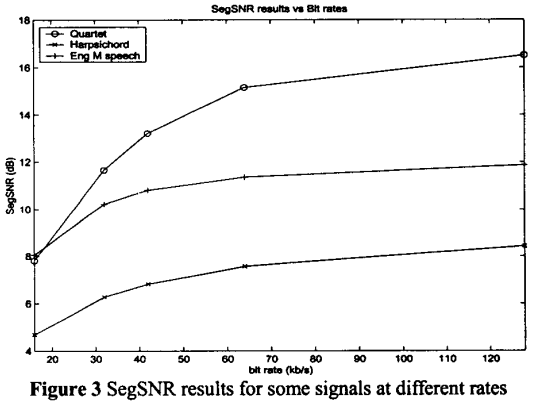

Figure 3 SegSNR results for some signals at different rates

6. CONCLUSION

A fixed rate sinusoidal coder employing a new technique for transmitting the amplitudes has been presented. The sinusoids are rearranged according to each sinusoid's energy contribution to the reconstruction of the original signal. A perceptual model is used to enhance the selection process and ensure the selection of perceptually significant information. The sorting scheme allows for scalable coding as the most important information is transmitted first, thus relating perceptual quality to bits used. The coder presented uses a short frame length and minimal algorithmic delay in the coding of the audio signal.

7. ACKNOWLEDGEMENTS

M. Raad is in receipt of an Australian Postgraduate Award (Industry) and a Motorola (Australia) Partnerships in Research Grant.

8. REFERENCES

[1] McAulay, R.J. Quatieri, T.F. "Sinusoidal coding" Chapter4 in "Speech coding and synthesis" edited by Kleijn, W.B. and Paliwal, K.K. Netherlands: Elsevier publishing, 1995.

[2] George, E.B. and Smith, J.T. "Analysis-by-Synthesis/Overlap-add sinusoidal coding applied to the analysis and synthesis of musical tones" J. Audio Eng. Soc., Vol. 40, No. 6, pp. 497-516. June 1992.

[3] Vos, K. Vafin, R. Heusdens, R and Kleijn, W.B. "High-quality consistent analysis-synthesis in sinusoidal coding", 17th AES international conference on High Quality Audio coding, pre-print version.

[4]http://www.tnt.uni-hannover.de/project/mpeg/audio/#mpeg4

[5] Raad, M. and Burnett, I.S. "Audio coding using sorted sinusoidal parameters" to be published in IEEE international symposium on circuits and systems, May 2001.

[6] Gibson, J.D. Berger, T. Lookabaugh, T. Lindbergh, D. Baker, R.L. "Digital compression for multimedia". USA: Morgan Kaufmann Publishers, Inc. 1998.

[7] Eberspacker, J. Vogel, H-J. "GSM switching, services and protocols" Great Britain: John Wiley & Sons ltd, 1999.

[8] Johnston, J.D. "Transform coding of audio signals using perceptual noise criteria" IEEE Journal on selected areas in communications vol. 6, No. 2, pp. 314-323 February 1988.

[9] Malvar, H.S. "Signal Processing with Lapped Transforms". USA: Artech House, 1992.

[10] Purnhagen, H. Meine, N and Edler, B. "Speeding up HILN – MPEG-4 Parametric Audio encoding with reduced complexity" AES 109th convention, LA Sept 2000. Preprint no: 5177.