FROM LOSSY TO LOSSLESS AUDIO CODING USING SPIHT

Mohammed Raad and Alfred Mertins

School of Electrical, Computer and Telecommunications Engineering
University of Wollongong, NSW 2500, Australia
mr10@uow.edu.au

ABSTRACT

This paper discusses the design and implementation of a scalable audio compression scheme that scales up from lossy to lossless compression. Scalable audio compression has been of interest in the audio compression community for some time, with the most obvious attempt at obtaining a solution coming in the form of the MPEG-4 standard [1]. At the same time the increase in bit rates in both mobile communications [2] and the internet's broadband technology means that audio compression algorithms with higher bit rates than currently used, such as MPEG's mp3 [1], can be employed to obtain higher quality. However, the new increased data rates are not necessarily constant, this is especially the case when considering the internet. As such, scalable schemes that can scale to lossless compression have become rather interesting from an application point of view. The scheme presented in this paper achieves lossless compression that is comparable with the state of the art whilst maintaining a scalable embedded bitstream.

1. INTRODUCTION

Lossless compression of audio aims to reduce the bandwidth or memory required to transmit or store the original audio signal. That is, the error between the original Pulse Code Modulated (PCM) signal and the compressed version is zero. The majority of digital audio material in use today is quantized using 16 bits per sample and obtained at a sampling frequency of 44.1 kHz or 48 kHz. The former is the CD standard for digital audio, while 48 kHz are used in audio studios. However, other sampling rates may be used in certain cases and a different quantization scheme utilized.

Currently, lossless audio coding has been approached from a signal model perspective [3],[4],[5]. The signal is typically modeled using a linear predictor, which may either be FIR or IIR [4] . The aim of using a linear predictor is to decorrelate the audio samples in the time domain and to reduce the signal energy that must be coded [3]. The coefficients of the linear predictor are coded as well as the excitation of the predictor, which is typically coded using an entropy code. The combination of the linear predictor with a variable length entropy code leads to a perfect reconstruction of the audio signal. The compression ratio of such coders typically depends on the nature of the audio signal being coded. Values reported range between 1.4 and 5.3 [3].

Another approach to lossless compression of audio signals involves the use of transform coding as presented in [6]. This approach is very similar in nature to the linear prediction approach as it utilizes a transform coder to produce a lossy compressed version of the original signal and an entropy code to compress the generated error signal between the lossy compressed signal and the original one. The transform coder decorrelates the audio samples and hence the transform coder operates on the same basic principles of decorrelation and entropy coding as the linear prediction based lossless coders [3]. The compression ratios reported in [6] again varied with the nature of the input audio signal and ranged between 2.2 and 3.2.

Similarly, scalable audio compression has been approached from a signal model point of view. Recent scalable coding schemes, such as the one described in [7], use a composite signal model. The model is built through the combination of Sinusoids, Transients and Noise (STN). The STN model of an audio signal is described in detail in [7] and [8]. The scalability obtained in [7] is mainly large step scalability, with more granular scalability made possible through the use of an adequately designed entropy code. The system in [7] is scalable between 6 kbps and 80 kbps, however, as different frame lengths are used to model the different signal components more adequately the scheme is seen mainly as an 'off-line' tool in [7].

Considering the advances in the bandwidth availability for cellular telephone and internet users, it is clear that a compression scheme that combines both scalability and lossless compression is of interest and potential use. For example, MPEG have started a process of standardization for such a scheme [9]. In this paper, we present a scalable audio coder that allows very fine granular scalability as well as competitive compression at the lossless stage. The compression scheme is built around transform coding of audio. Particularly, the Set Partitioning In Hierarchical Trees (SPIHT) algorithm [10] is used to allow scalability as well as perfect reconstruction. Transform coding takes advantage of the more harmonic structure of an audio signal. It also allows fine grain scalability, which is more difficult to obtain in parametric coders, such as those that rely on linear prediction. Similarly, the use of SPIHT allows the coder to quantize the transform coefficients in such a manner that only the input audio segment's statistics are required, avoiding the necessity to design dedicated entropy code books.

This paper is organized as follows. Section 2 describes the different components of the proposed scalable-to-lossless scheme. Section 2.1 gives a brief outline of the SPIHT algorithm and Section 2.2 presents a study that was conducted to illustrate how SPIHT may be applied to achieve lossless audio compression. In Section 2.3 the actual scalable-to-lossless scheme is presented along with some general results. Section 2.4 looks at determining the optimal rate for the lossy component, and Section 2.5 describes sensory pleasantness, its contributing factors and how these factors may be used to objectively analyze an audio compression scheme. In that same section results are presented to show how the

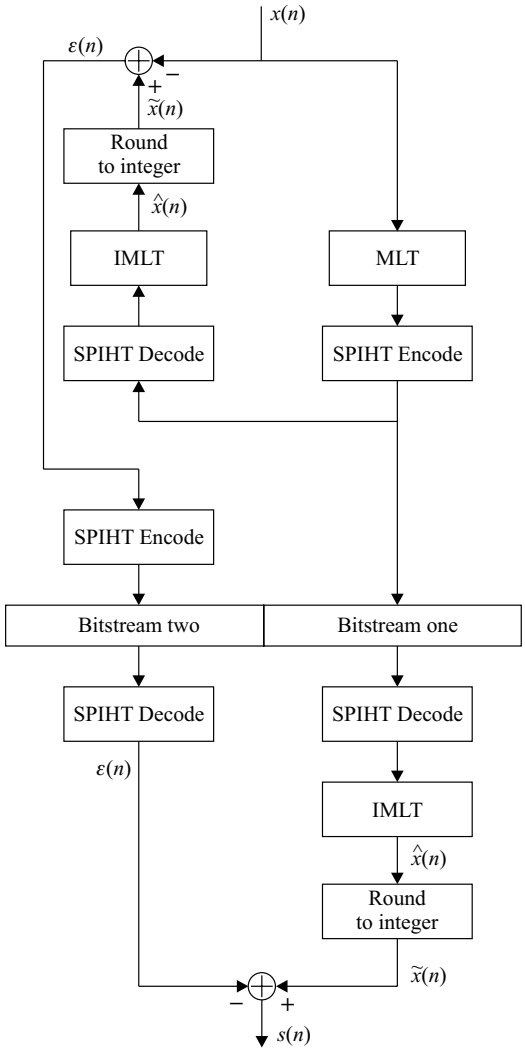

Figure 1: *The scalable-to-lossless scheme based on SPIHT.*

presented scalable-to-lossless scheme affects the sensory pleasantness factors of the test material. Section 3 presents the lossless and scalable-to-lossless results obtained, and Section 4 provides a brief conclusion.

2. SPIHT AND LOSSLESS AUDIO COMPRESSION

The structure of the coder proposed in this paper is depicted in Figure 1. It consists of the combination of the lossy coder of [11], which is based on the Modulated Lapped Transform (MLT) and SPIHT, and a lossless coder for transmitting the error made by the lossy part. The lossy part is given by the right half of the structure in Figure 1, and the error coding (if present) takes place in the left half. Note that both parts of the coder are based on the SPIHT algorithm. In this section we mainly focus on the lossy part of the structure, referred to as MLT-SPIHT.

The input signal is transformed using the MLT where floating point calculations are used. The transform coefficients are encoded using SPIHT, and the bitstream is transmitted to the decoder. We will refer to this bitstream as bit stream one. Bitstream one is decoded at the encoder and the synthesized audio is subtracted from the original audio to obtain the output error. Here integer operations are used, so that the error output is integer and, as has been discussed, usually has a dynamic range that is less than that of the original integer signal. The time-domain error signal is then encoded into bitstream two, using a second SPIHT encoder. At the decoder, both bitstreams are received as one global bitstream, with bitstream one making up the first part of the total bitstream. The decoder may decode up to any rate desired. If bitstream one containing the transform coefficients is used up, then the decoder recognizes that the remaining bitstream is for the time-domain error signal, which it reconstructs and adds to the synthesized signal. The complete scalable-to-lossless system in Figure 1 will be analyzed further in Section 2.3.

2.1. A brief discussion of SPIHT

SPIHT [10] is a coding algorithm that allows the transmission of coefficients in a pseudo-sorted fashion where the most significant bits of the largest coefficients are sent first. The sorting is carried out according to the magnitudes of the coefficients. The generated bitstream is fully embedded, allowing best reduction of coding noise with every additional bit sent [10]. It can be truncated at any point to achieve the best reconstruction for the actual number of bits sent. The original design of SPIHT was aimed at image compression, and the intent was to use the algorithm in the frequency domain [10]. However, the algorithm may also be used in the time domain.

The encoder output consists of sorting information that is required to identify the significant coefficients with respect to an actual bitplane and of refinement information for enhancing the accuracy of significant coefficients. The algorithm employs a number of linked lists which are manipulated according to a significance test that is at first applied to sets and then eventually to individual coefficients. The sets are generated by defining offspring for each coefficient. The offspring of the linked coefficients are connected together to form sets. If a set becomes significant with regard to the tested bitplane, it gets partitioned into smaller sets which will be tested for significance again, until all significant coefficients are localized. In this paper, as in [11], the offspring of coefficient i are defined by

$$O(i) = iN + \{0, N-1\}. \tag{1}$$

where N is the number of offspring used. In our case, $N = 4$ is chosen. The decoder imitates the encoder action when it is given the test results and hence the decoder develops the same set of sorted coefficients as seen by the encoder.

Two factors that are important to the performance of SPIHT are the dynamic range of the input coefficients and the energy distribution across the coefficients. Small coefficients will be coded using less bits than large ones, and if the energy of the signal is concentrated in a few coefficients then SPIHT will quickly locate those coefficients and transmit their significant bits.

2.2. Achieving lossless compression with MLT-SPIHT

As a starting point in our discussion about lossless compression it is important to clarify what is meant exactly by achieving lossless compression. Assuming that the original audio signal $x(n)$ is PCM coded and consists of a sequence of integers, it is sufficient for perfect reconstruction that a synthesized audio signal $\hat{x}(n)$ can be

generated that satisfies

$$|x(n) - \hat{x}(n)| < 0.5 \tag{2}$$

for all n, because then a rounding operation allows us to recover $x(n)$ from $\hat{x}(n)$ without error. In other words, the linear synthesis part of an audio coder does not necessarily need to produce an error free reconstruction. It only needs to bring $\hat{x}(n)$ close enough to $x(n)$ that the nonlinear rounding operation finally yields perfect reconstruction.

SPIHT allows one to specify the accuracy to which the given coefficients or samples be coded. It is also possible to precisely define the total bit rate that can be used for coding. When considering these facts with the knowledge that the condition for lossless representation is given by (2), then it can be deduced that with a high enough coding resolution of the MLT transform coefficients one can already achieve lossless compression. To show that this is indeed possible, we conducted a number of experiments where the coding resolution for the lossy part (the right side of Figure 1) was varied between 10 and 25 bits at various limiting maximum bit rates.

The frame length used is 1024 samples, with 512 samples of overlap. That corresponds to 23.2 ms a frame at a sampling rate of 44.1 kHz. The maximum bit rates were set at 192 kbps, 353 kbps and 512 kbps, respectively. The nearness of the synthesized audio to the original was estimated through the calculation of the first-order entropy of the error signal $\varepsilon(n) = \tilde{x}(n) - x(n)$, where $\tilde{x}(n) = \text{round}(\hat{x}(n))$ denotes the rounded output signal of the MLT synthesis bank, using the well known equation:

$$H(\varepsilon) = -\sum_{\varepsilon} p(\varepsilon) \log_2 p(\varepsilon) \tag{3}$$

The test material that has been used in this work was obtained from [12] and is part of the Sound Quality Assessment Material (SQAM) used by MPEG. Table 1 lists the test material and the associated file names. In the following we present results that were obtained using file x1 as they are sufficient to demonstrate the conditions under which SPIHT combined with the MLT will reach lossless compression. Figure 2 shows the results of the experiment. There are a number of points to note from the figure, first given a high enough rate and coding resolution the MLT-SPIHT system does produce an exact copy of the original as indicated by the entropy reaching zero. Secondly the maximum rates defined do not affect the entropy result until at least 15 bits are being used for the quantization. This illustrates how the two factors of limiting rate and quantization resolution interact to affect the quality of the synthesized signal. One can say that above a certain coding resolution the limiting rate is the important factor for the quality of the synthesized signal. The presented results also allow for a comment about the expected lossless rate when coding the error with an entropy code. For example, at a coding resolution of 20 bits and a lossy rate of 192 kbps the final lossless rate should be approximately 345 kbps (assuming an entropy code that codes at first-order entropy and reading from the figure that at 20 bits and 192 kbps maximum rate the entropy is approximately 3.5 bits per sample). Note that this rate is well below the 512 kbps rate which achieves approximately zero error entropy at 23 bits coding resolution using the straightforward MLT-SPIHT coder. This observation is very important regarding the scheme proposed here as it shows that a lossy scheme based on SPIHT combined with a lossless scheme will produce a better lossless compression ratio than the MLT-SPIHT scheme alone.

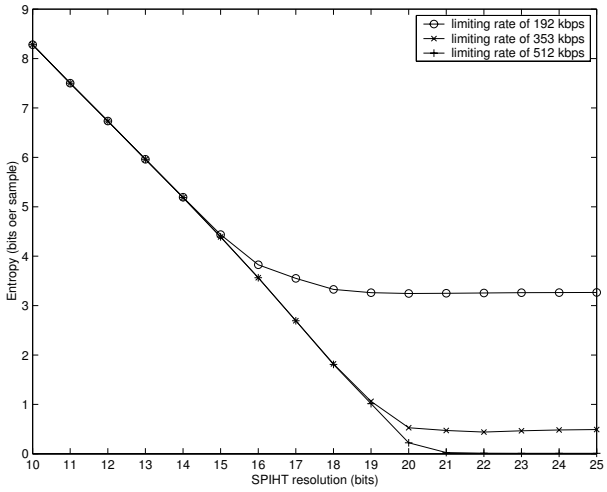

Figure 2: *The entropy of the error signal using different SPIHT resolutions at three maximum rates.*

2.3. The scalable-to-lossless scheme

It has been mentioned in the previous section that if an entropy code for the residual error was to be combined with the lossy MLT-SPIHT scheme, then a good overall lossless compression ratio may be expected. It has also been outlined that one of the factors that generally influence the performance of SPIHT is the dynamic range of the input data. Hence, it is possible to reason that if the dynamic range of the synthesis error was sufficiently small then SPIHT could still be used to code the error signal at an acceptable rate, whilst maintaining the scalability of the coder in terms of waveform matching, until the lossless condition is met.

An example of the difference in dynamic range between the original audio signal $x(n)$ and the error signal $\varepsilon(n)$ is shown in Figure 3 where $x(n)$ is coded at 64 kbps. It can be seen from the figure that the reduction in dynamic range is significant. It is also important to determine the statistical properties of the error signal, particularly the similarity between the error signal and white noise. This is important as it determines if there would be any gain in the use of a transform to further decorrelate the error signal. As expected, an analysis shows that the more bits that are spent on the compression of the original signal the more white-noise like is the error signal, and the less benefit one can expect from transforming the error signal. To illustrate this, Figure 4 shows the Power Spectral Densities (PSDs) of two versions of the error signal for a coded frame of audio at rates of 64 kbps and 128 kbps, respectively.

As mentioned earlier, the scalable-to-lossless scheme is based on the combination of a lossy scalable component with a scalable error coding component. To obtain good lossless performance one must adjust the bit rates used by the lossy component, as has been illustrated in the discussion about Figure 2. Here we present further analysis of what the lossy rate should be set at to obtain good lossless results. We will also discuss the effect of this lossy rate on the subjective quality of the reconstructed signal as described objectively by pleasantness parameters.

2.4. Determining the maximum lossy rate

A number of experiments have been conducted to determine what rate the lossy scalable component of the coder should be set at.

Table 1: The Signal Content.

Signal Name	Signal Content	Signal Name	Signal Content
x1	Bass	x9	English Female Speech
x2	Electronic Tune	x10	French Female Speech
x3	Glockenspiel	x11	German Female Speech
x4	Glockenspiel	x12	English Male Speech
x5	Harpsicord	x13	French Male Speech
x6	Horn	x14	German Male Speech
x7	Quartet	x15	Trumpet
x8	Soprano	x16	Violoncello

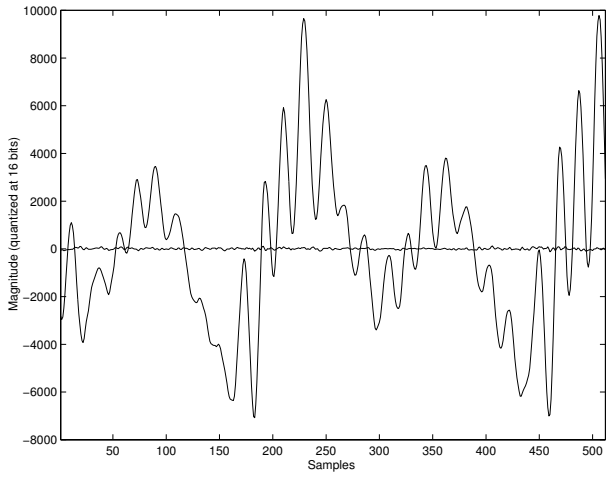

Figure 3: The difference in the dynamic range between the error signal and the original signal when the lossy coder is operating at 64 kbps (the smaller signal is the error).

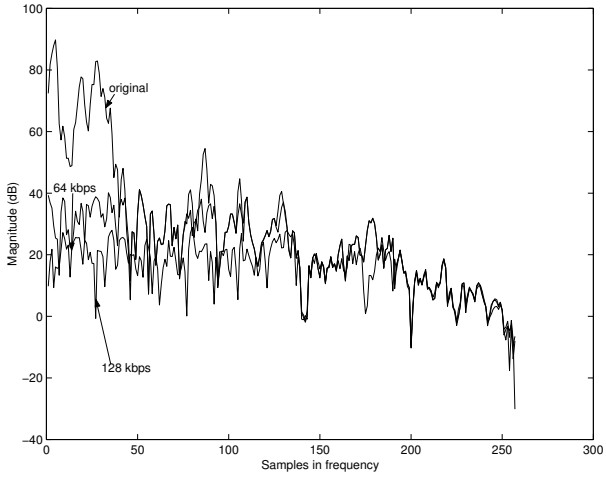

Figure 4: *PSD of the error signal at 64 kbps and 128 kbps as compared to the original.*

Figure 5 shows the results of one such experiment where the lossy maximum rate was set to values between 16 kbps and 256 kbps (inclusive) in 16 kbps intervals. Here 18 bits for SPIHT coding resolution was used. At each maximum lossy rate the entropy of the error has been calculated and used to determine the lower bound for the rate required to achieve lossless compression if an entropy code was to be used to code the error. Two of the three curves on the graph describe the expected rate in different situations, and one gives the collected rate with the proposed coder. The top curve (i.e. the one with the worst performance) describes the expected lossless rate if lossy rate reservation was used, that is if bitstream one was allocated the maximum lossy rate all the time. SPIHT does not require such reservation of bitstream space. The second curve from the top takes this into account and does not assume that bitstream one is allocated the maximum rate all the time, instead it utilizes the actual rate required by SPIHT for a complete reconstruction of the coefficients up to the coding resolution that is hard coded at both encoder and decoder. This curve continues to decrease with the decreasing entropy of the error signal and finally at 192 kbps crosses the lowest curve in the figure. The lowest curve in the figure is the actual rate collected for the proposed coder. It is noticeable that the SPIHT scheme outperforms the lossy-plus-entropy code scheme until the 192 kbps mark for the maximum lossy rate. The reason behind this is that SPIHT transmits only the significant bits of the coefficients, and importantly, for zero coefficients or samples the algorithm does not transmit a single bit. An entropy code must transmit at least one bit (and in most cases two) per coefficient, even if that coefficient was zero. SPIHT avoids these extra bits by recognizing large sets of zero coefficients or samples and treating them collectively in the sorting process.

2.5. Psychoacoustic analysis of the lossy component

Having analyzed the results in terms of lossless compression, the performance of the coder has to be analyzed for its subjective effects on the synthesized audio at different lossy rates. A psychoacoustic analysis of the lossy scalable component of the coder was performed to add a perceptual dimension to the choice of the maximum lossy scalable rate. The analysis determined the mean variation between the pleasantness parameters of the original signal and the synthesized signal at different maximum lossy rates.

Sensory pleasantness describes the acceptability of a given sound to the human ear [13]. The contributing factors to sensory pleasantness are sharpness, roughness, loudness and tonality. Sharpness may be viewed as a measure of the density of loudness across the spectrum in different critical bands. Sharpness is most heavily influenced by the center frequency of the sound as well as the spectral content [13]. Loudness is a relative measure indicating Sound Pressure Level (SPL) of a 1 kHz signal that would sound as loud as the given sound. Roughness describes the inability of the ear to distinguish tonal components. That is, a sound that is noise-like sounds rough. Finally, tonality describes in relative

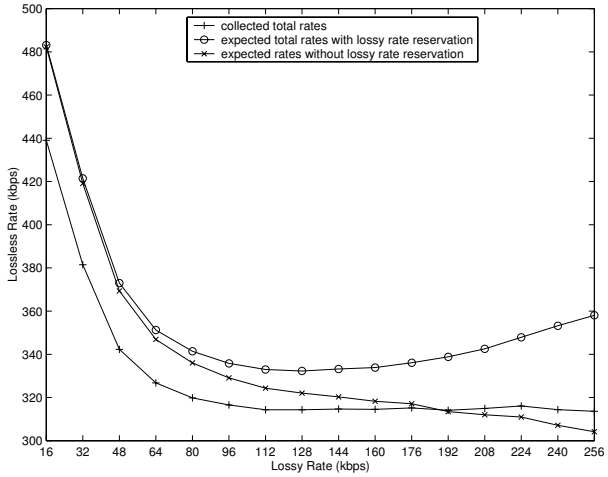

Figure 5: *Mean lossless rates collected, compared with the lossless rates expected.*

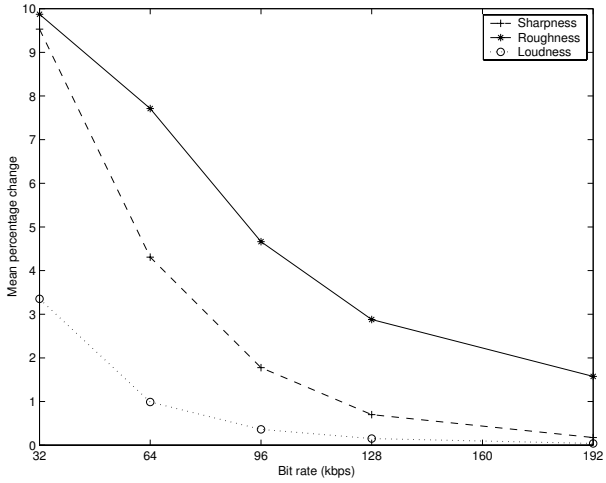

Figure 6: *Sharpness, roughness and loudness variations at different lossy rates for x1.*

terms how tone-like the signal is. Mathematical models describing the sharpness, roughness and loudness have been proposed in [13] and these models are used in the analysis here. Tonality can be approximated subjectively as suggested in [13], however, here we focus on the calculation of the contributing factors and hence tonality has been ignored. It should be pointed out that these factors may not necessarily be unique to a given sound. Whilst this is true, if a high SNR is also obtained, then one can confidently state that the sound has been reproduced with high fidelity.

Figure 6 shows the results obtained at different maximum lossy rates for signal x1. The curves show the mean percentage variation in the psychoacoustic parameters, denoted as E_v and calculated by the use of the equation:

$$E_v = E\Big(\frac{|p - p_0|}{p_0}\Big) \times 100\% \qquad (4)$$

where p_0 is the value of a pleasantness factor calculated for the original signal $x(n)$, p is the pleasantness factor calculated for the reconstructed signal, and $E(\cdot)$ denotes the expectation operation. It can be seen that the mean variation decreases with the increasing rate, however it can also be seen that the variation is not massive at any rate, starting at near the 10% mark for sharpness and roughness and near the 3% mark for loudness. The low variation of loudness is expected as at 32 kbps, the lowest rate used, SPIHT would have transmitted good approximations of the most significant spectral components, leading to a loudness level that is similar to the original one. Sharpness is influenced by the center frequency of the signal and the distribution of spectral components, which should also be well approximated at 32 kbps. A similar line of reasoning follows for the roughness result. Thus the variation is expected to be small, the important property is how the variation is reduced. That is, at what rate does the reduction in variation saturate. The presented figure shows that the percentage variation reaches a knee point at around the 96 to 128 kbps marks. Similar results were obtained for other signals tested. The knee point position has been found to depend on the spectral content of the signal being used, which is expected, with highly tonal signals reaching the knee point at lower rates than more noise-like signals. Using the psychoacoustic results and the lossless rate versus lossy rate results presented in Section 2.4, it is safe to conclude that any lossy rate between 128 kbps and 192 kbps will produce good scalable-to-lossless performance.

3. RESULTS

Two sets of results are presented here: the lossless compression results and the objective scalable lossy results of the MLT-SPIHT coder. First we consider the lossless compression results.

3.1. The lossless compression results

Using the experiments described in Sections 2.4 and 2.5, it was determined that a lossy maximum rate of 192 kbps should be used in combination with a coding resolution of 18 bits per spectral coefficient and 16 bits (PCM) per time domain error coefficient. Table 2 shows the results for the lossless compression of the SQAM files of Table 1. Most of the files show a compression ratio that is above 2, which is competitive with the current state of the art in lossless compression [3]. The lowest compression ratio was 1.74 for female French speech, whilst the greatest ratio obtained was 5.27 for an electronic tune. The average compression ratio obtained was 2.46. As with other current schemes, the compression ratio depends strongly on the content of the signal [3]. In most current schemes, the compression ratio is higher for highly predictable signals that can be very well modeled by the use of a linear predictor. In this case, and because of the scalability capability, the more concentrated the energy of the signal is in the frequency domain the higher the compression ratio. The reason being that a signal with concentrated energy in the frequency domain is coded very well in the first part of the coder and so a very small, highly uncorrelated, error signal is produced leading to a high lossless compression ratio overall.

3.2. Objective Results for the scalable-to-lossless and lossy coders

Figure 7 shows the Segmental Signal-to-Noise Ratio (SegSNR) results for a lossy coded version of signal x1 (up to 240 kbps) as well as the performance of the scalable-to-lossless scheme described

Table 2: Results for the Lossless SPIHT Coder.

Signal	Mean Rate (kbps)	Compression Ratio	Bits/Sample
x1	318	2.22	7.20
x2	134	5.27	3.03
x3	206	3.43	4.65
x4	266	2.65	6.01
x5	346	2.04	7.84
x6	232	3.04	5.23
x7	354	1.99	8.01
x8	317	2.23	7.18
x9	366	1.93	8.28
x10	405	1.74	9.17
x11	362	1.95	8.19
x12	368	1.92	8.33
x13	360	1.96	8.15
x14	360	1.96	8.15
x15	255	2.77	5.75
x16	306	2.31	6.93

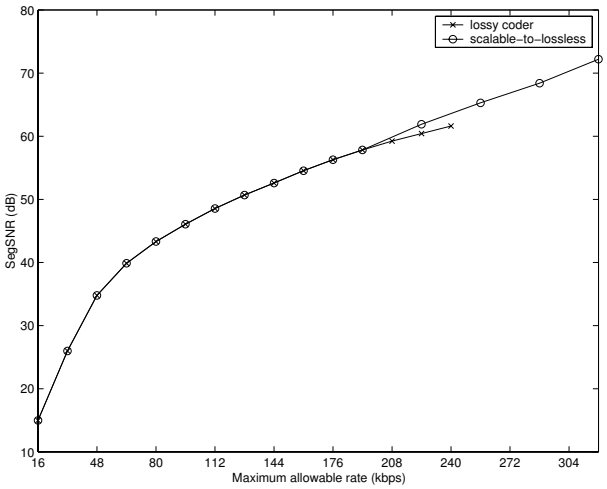

Figure 7: *Objective results for the lossy MLT-SPIHT coder and the scalable-to-lossless coder.*

earlier up to and including 320 kbps. The SegSNR values are calculated using frames that are 17.5 ms long and not overlapping (note that this does not match the frame selection in the coding scheme). It can be seen from Figure 7 that there is a knee point for the coder at around the 64 kbps mark. It can also be seen that the SegSNR is above the 40 dB mark at 64 kbps indicating that a high quality signal has been synthesized. As expected, the lossy coder saturates at the high bit rates. In contrast the scalable-to-lossless scheme continues to improve the SegSNR of the synthesized signal. It is important to note that the values presented in the figure are calculated across frames that have not been perfectly reconstructed. At 320 kbps there were 530 frames (from a total of 1426) that were coded losslessly. The remaining error in the other frames is clearly very small. Note also that the rates listed in Table 2 are average rates, while the rates shown in Figure 7 are the maximum rates that the coder is permitted to use.

4. CONCLUSION

This paper has presented a scalable-to-lossless scheme that allows scalability from lossy compression to lossless compression with the use of a single bitstream. The bitstream can be truncated at any point to meet a desired bit budget and obtain the best signal approximation for the chosen rate. A complete analysis has been presented which included rate considerations as well as objective perceptual considerations. Currently the coding scheme does not include a perceptual model to allow the transmission of perceptually significant coefficients first. This will be implemented in the continuing development of this coder.

5. ACKNOWLEDGEMENTS

Mohammed Raad is a recipient of an Australian Postgraduate Award of Industry (APAI) grant. This work is supported by the Motorola Australian Research Centre. The authors wish to thank Mr. Christian Ritz for his helpful editorial comments.

6. REFERENCES

[1] K Brandenburg, O Kunz, and A Sugiyama, "MPEG-4 natural audio coding," *Signal Processing: Image Communication*, vol. 15, no. 4, pp. 423–444, Jan. 2000.

[2] K.W. Richardson, "UMTS overview," *Electronics and communication engineering journal*, vol. 12, no. 3, pp. 93–100, June 2000.

[3] M. Hans and R.W. Schafer, "Lossless compression of digital audio," *IEEE Signal Processing magazine*, vol. 18, no. 4, pp. 21–32, July 2001.

[4] P.G. Craven and M.J. Law, "Lossless compression using IIR prediction filters," AES 102nd convention, AES preprint 4415, March 1997.

[5] A.A.M.L. Bruekers, W.J. Oomen, and R.J. van der Vleuten, "Lossless coding for DVD audio," AES 101st convention, AES preprint 4358, November 1996.

[6] T. Liebchen, M. Purat, and P. Noll, "Lossless transform coding of audio signals," *Proceedings of the 102nd AES convention, AES preprint 4414*, March 1997.

[7] T.S. Verma, *A perceptually based audio signal model with application to scalable audio compression*, Ph.D. thesis, Department of Electrical Engineering, Stanford university, October 1999.

[8] S.N. Levine, *Audio representations for data compression and compressed domain processing*, Ph.D. thesis, Department of electrical engineering, Stanford university, December 1998.

[9] T. Moriya, "Report of AHG on issues in lossless audio coding," ISO/IEC JTC1/SC29/WG11 M7955, March 2002.

[10] Amir Said and William A. Pearlman, "A new, fast, and efficient image codec based on set partitioning in hierarchical trees," *IEEE Transactions on Circuits and Systems For Video Technology*, vol. 6, no. 3, pp. 243–250, June 1996.

[11] M. Raad, A. Mertins, and I. Burnett, "Audio compression using the MLT and SPIHT," *Proceedings of DSPCS' 02*, pp. 128–132, 2002.

[12] "Mpeg web site at http://www.tnt.uni-hannover.de/project/mpeg/audio," .

[13] E. Zwicker and H. Fastel, *Psychoacoustics*, Springer-Verlag, Berlin, second edition, 1999.