# A SCALABLE TO LOSSLESS AUDIO COMPRESSION SCHEME

*Mohammed Raad, Alfred Mertins and Ian Burnett*

School of Electrical, Computer and Telecommunications Engineering
University Of Wollongong
Northfields Ave Wollongong NSW 2522 Australia
email: mr10@uow.edu.au

## ABSTRACT

This paper outlines a scalable to lossless coder, that is the coder presented is a scalable coder that scales from lossy quality to lossless quality. Lossless compression is achieved by concatenating a lossy, scalable transform coder with a scalable scheme for the compression of the synthesis error signal. The lossless compression results obtained are comparable with the state of the art in lossless compression (that is a compression ratio ranging from 1.74 to 5.27). The added advantage of the compression scheme presented is the scalability, which is obtained by basing the lossy coder on the Set Partitioning In Hierarchical Trees (SPIHT) algorithm.

## 1. INTRODUCTION

With the introduction of third generation cellular phone systems and the possible expansion of those systems, digital cellular phone users may in the near future have access to data rates above 144 kbps [1]. This is a considerable increase on what the second generation of mobile telephony presented, known as GSM [1, 2], which in most implementations only provides users access to 9.6 kbps of data [1]. Such an explosion in the possible bit rate and the nature of the proposed bit streams means that multimedia compression schemes may be adjusted to allow for increased quality of the delivered product to the user. The other well known medium of multimedia delivery, the internet, is also experiencing an increase in possibilities with the introduction of broadband technology.

The increase in bit rates means that audio compression algorithms with higher bit rates than currently used, such as MPEG's mp3 [3], can be used 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 and lossless schemes have become rather interesting from an application point of view.

Currently, lossless audio coding has been approached from a signal model perspective [4],[5],[6]. The signal is typically modelled using a linear predictor, which may either be FIR or as in the case of [5] IIR. The compression ratio of such coders typically depends on the nature of the audio signal being coded and may range between 1.4 and 5.3 [4].

Similarly, scalable audio compression has been approached from a signal model point of view. Recent scalable coding schemes, such as the scheme 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 presented more as an 'off-line' tool in [7].

With the aim of standardizing a scalable compression scheme, MPEG proposed different audio coders for different rates [3]. A scalable parametric coder has been adopted by MPEG as described in [9], which is built around a sinusoidal model of the audio signal.

Having described the advances in the bandwidth availability for cellular telephones, and that for internet users, it is clear that a compression scheme that combines both scalability and lossless compression is of interest and potential use. MPEG have started a process of standardization for such a scheme [10].

In this paper we present an implementation of a scalable audio coder that allows very fine granular scalability as well as compression at the lossless stage. The compression scheme is built around transform coding of audio. Particularly, the Set Partitioning In Hierarchical Trees (SPIHT) algorithm [11] is used to allow scalability as well as perfect reconstruction. The results presented show that significant lossless compression is obtained.

## 2. LOSSLESS AUDIO COMPRESSION

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 major-

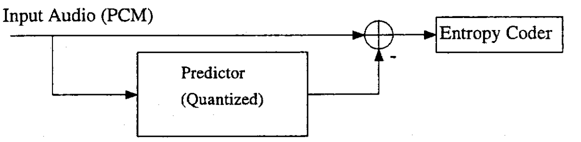


Figure 1: Lossless Compression using Linear Prediction

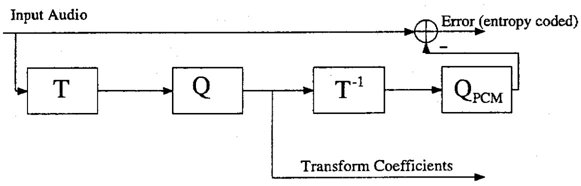


Figure 2: Lossless Compression Using Transform Coding

ity of digital audio material in use today is quantized using 16 bits per sample and obtained at a sampling frequency of 44.1 kHz. That is the CD standard of a digital audio signal, however other sampling rates may be used and a different quantization scheme utilized.

A lossless compression scheme achieves the stated aim by the removal of redundancy in the original signal. This redundancy is typically removed by the use of a linear predictor [4]. The output of the linear predictor is treated as the approximation of the audio signal. The error between the original signal and the approximate signal is typically coded through the use of an entropy code, such as Huffman or Rice, which are lossless. Figure 1 illustrates a typical lossless scheme.

Lossless audio compression may be viewed as an adaptation of more general lossless coding schemes such as 'Lempel Ziv' [12] which attempt to reduce the storage capacity required for a given set of samples. However, it has been found that such algorithms only produce a small amount of compression when applied to PCM audio signals [6], and hence the use of algorithms that take advantage of the nature of the audio signal. The linear prediction algorithms tend to use frame lengths of approximately 25 ms to take advantage of the pseudo-stationary nature of the audio signal of that length.

It has been argued in [13] that transform coding would be more suitable for lossless compression as it models the audio signal more accurately. Lossy audio compression is fundamentally based on transform coding [14] as that allows the harmonic nature of the audio signal to be captured. Based on this model, the work in [13] proposes approximating the audio signal by the use of a transform coder and coding the error signal by the use of an entropy code. Figure 2 shows the structure of the lossless coder proposed in [13]. This approach is actually very similar in nature to the linear prediction approach as the use of 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 [4]. The compression ratios reported in [13] again varied with the nature of the input audio signal and ranged between 2.2 and 3.2 for the test set that was used, which had some similarity with that presented in [4] but was not exactly the same.

The vast majority of lossless compression algorithms can be grouped under the two groups of prediction based coders and transform based coders as described previously [4], [13], [10]. However, such coders have been optimized to obtain the greatest possible compression at lossless operation. Also, all the coders reviewed here are variable rate coders. Having an optimized lossless scheme limits the ability to use such a scheme in a scalable system. Similarly, a scalable system should be theoretically scalable to lossless with a preferably linear increase in perceptual quality. In the next section we present a scalable to lossless scheme that allows bit rates to be controlled bitwise. That is, each bit received adds to the quality of the synthesized signal, in terms of signal to noise ratio.

## 3. A SCALABLE TO LOSSLESS SCHEME USING SPIHT

To achieve lossless compression, that is to obtain zero error between the original signal and the synthesized signal, a number of approaches may be taken. Section 2 discussed the current approaches taken by researchers. Building on the approach of [13], a lossy transform compression scheme may be combined with a lossless compression scheme for the error signal. In our case, the lossy transform scheme is actually a scalable scheme. In a previous work involving SPIHT [15] we presented results that indicated the developed coder produced very good synthesized audio (that is, near perceptually insignificant error) at rates around and above 56 kbps. That coder used the MLT, as well as a perceptual model that removed perceptually insignificant signal components.

The solution proposed here is also built on that system of [15] but without the perceptual model. Figure 3 shows the proposed coder The input signal is transformed using the MLT, here floating point calculations are used, the signal is coded using SPIHT and the bit-stream transmitted to the decoder. We will refer to the first bit stream as bit stream one. Bit stream 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 typically has a dynamic range that is equal to or less than that of the original integer signal (otherwise the lossy compression scheme would not have done a good job of mimicking the original signal). An Example of the difference in dynamic range between the original audio signal and the synthesized signal is shown in Figure 4 where the original signal is coded at 64 kbps. The smaller dynamic range is important when coding a signal with losslessly (see the discussion in [16]). In the present case, the given coefficients are in the time domain. Although SPIHT was originally aimed at frequency domain signals [11], the error signal has an important property in common with a frequency domain signal (that is one transformed from the time domain) in that its individual samples are much less correlated with each other than the original samples. In fact the more bits that are spent on the compression of the original signal the more white-noise like is the error signal. To illustrate this, Figure 5 shows the PSD of two versions of the error signal for a coded frame of audio at rates of 64 kbps and 128 kbps.

Hence, the shrinking of the dynamic range improves the coding performance of SPIHT for the error signal, and the uncorrelated nature of the time domain error signal means that very little will be gained if the signal is transformed, and so in this case it is not.

Having developed the idea to this stage, it is now important to decide the rate to which the lossy side of the coder operates. The accuracy of the synthesized signal to the original is dictated by the quantization resolution that is chosen in SPIHT. Quantization resolution in this context refers to the resolution chosen to quantize the frequency domain transform coefficients, and should not be confused with the resolution used for the quantization of the time domain signal, which is 16 bits PCM. Figure 6 shows the relationship between the SegSNR and the quantization resolution used in SPIHT (here complete reconstruction of the quantized transform coefficients is used). The 13 SQAM files (Cd quality, monotone) used in obtaining the results discussed in this paper are listed in Table 1. The curves in the figure are quite linear indicating a linear trade-off between synthesized signal quality and quantization resolution (remember that in this case no perceptual model is being used). As a matter of experience, a SegSNR of 50 dB produces very good quality and so the quantization resolution that is used in the first section of the coder has been set to 18 bits, as all of the coded files show a SegSNR well above 50 dB at that quantization resolution.

The combination of bit stream one and bit stream two gives the complete bit stream which determines the rate of the coder. The base rate which produces the least overall rate may be determined experimentally, Table 2 shows the results of an experiment with a voiced section of signal x1. The table includes the first order entropy of the error signal, obtained with the equation:

$$H(x) = -\sum_{x \in X} p(x) \log_2 p(x) \tag{1}$$

The first order entropy indicates the expected limit to losslessly code the error bit stream. Using this expected limit, the expected lowest rate is calculated. Table 2 shows that the lowest expected rate is obtained when a base rate of 192 kbps is used. This does not match the actual rate results exactly (the final column in the table), yet it is close. The difference between the expected overall rate and the actual is due to the sorting carried out by SPIHT. The entropy describes the minimum

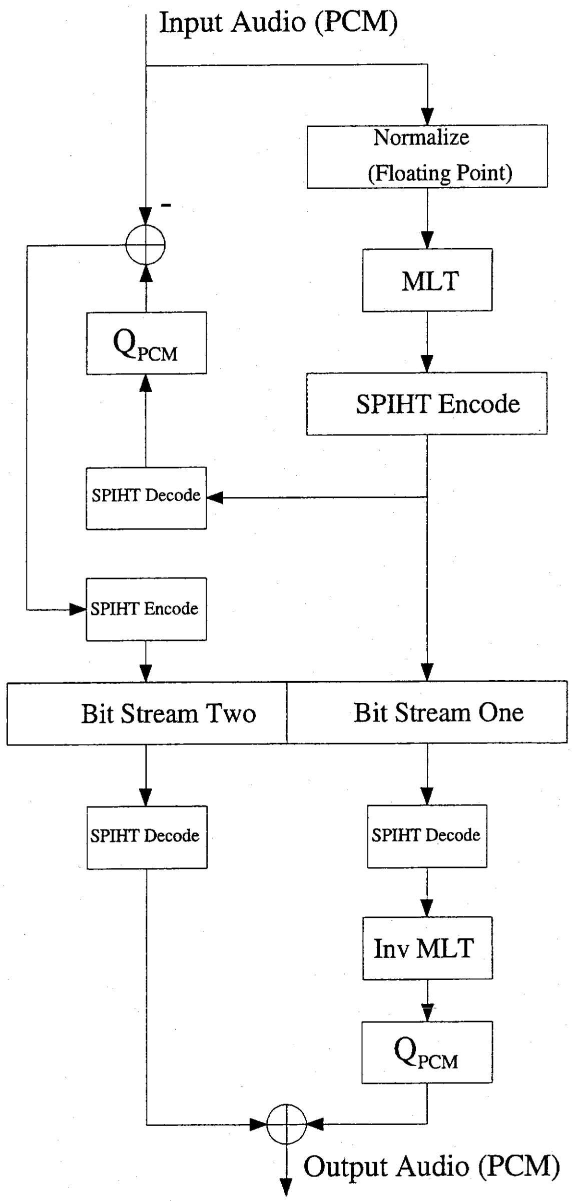


Figure 3: The Scalable to Lossless scheme based on SPIHT

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 |

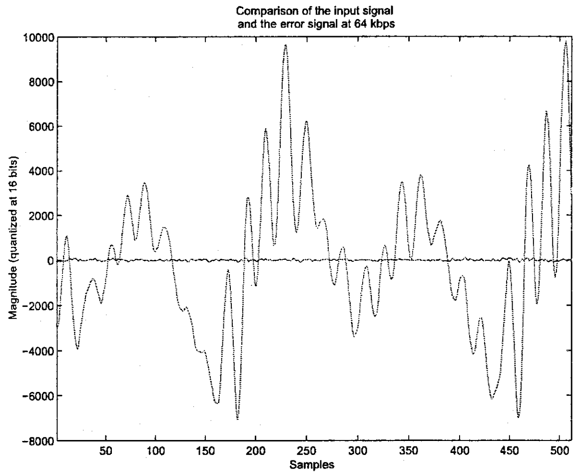


Figure 4: The difference in the dynamic range between the error signal and the original signal when the lossy coder is operating at 64 kbps (the smaller signal is the error)

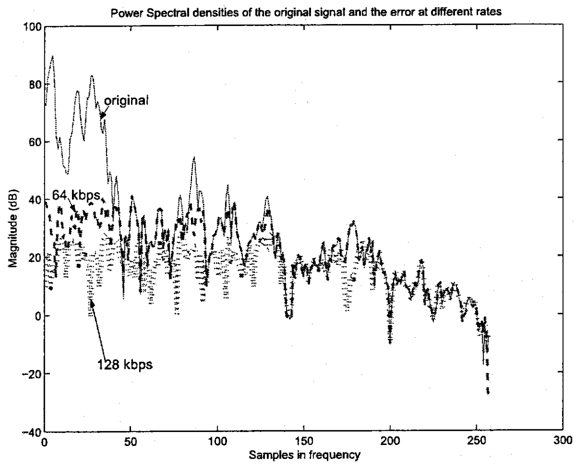


Figure 5: PSD of the error signal at 64 kbps and 128 kbps as compared to the original

number of bits required to faithfully recreate the error signal [12], however this can only be used as an indication as to the possible performance of SPIHT. The reason being that SPIHT sorts samples (or coefficients), thus requiring sorting information, but it also does not transmit insginificant bits or any zero samples. The non-transmission of zero samples is actually a saving on the typical implementation of an entropy code which normally requires some bits (the number of which depends on the statistics of the signal) to code each zero sample [12].

As the error signal is being sorted and coded in its time domain form, each loop of SPIHT reduces the time domain error between the original signal and the synthesized signal.

## 4. RESULTS

Table 3 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 [4]. 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 [4]. In most current schemes, the compression ratio is higher for highly predictable signals that can be very well modelled 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.

## 5. CONCLUSION

We have presented a lossless compression scheme for CD quality audio that is at the same time scalable. To achieve this, a

Table 2: Experimental Results for The Overall Rate Given Various Base Rates

| Base Rate (kbps) | First Order Entropy (bits per sample) | Total Rate Expected (kbps) | Actual Error Rate (kbps) | Actual Total Rate (kbps) |
|---|---|---|---|---|
| 64 | 7.02 | 374 | 344 | 408 |
| 80 | 6.48 | 366 | 329 | 409 |
| 96 | 6.01 | 362 | 305 | 401 |
| 128 | 5.24 | 359 | 273 | 401 |
| 192 | 3.72 | 356 | 213 | 405 |
| 256 | 2.56 | 369 | 170 | 426 |

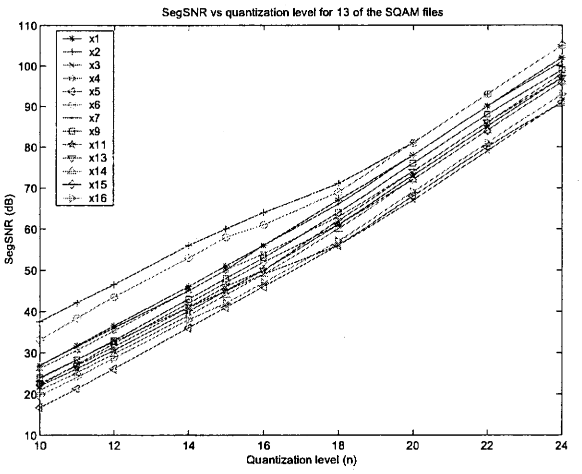


Figure 6: The SegSNR vs the quantization resolution for the lossy side of the coder

Table 3: Results for The Lossless SPIHT Coder

| Signal | 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 |

scalable lossy scheme has been combined with a scalable time domain scheme. Both schemes are built around SPIHT, and the combination of these two schemes has produced lossless compression results that are comparable with the current state of the art. The results presented are mean bit rates, however the coder does have the added advantage of having its bit rate controlled to a resolution of one bit.

## 6. ACKNOWLEDGEMENT

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

## 7. REFERENCES

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

[2] Jorg Eberspacher and Hans-Jorg Vogel, *GSM Switching, Services and Protocols*, John Wiley and Sons, Chichester, 1999.

[3] 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.

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

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

[6] 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.

[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] H. Purnhagen and N. Miene, "HILN - the MPEG-4 parametric audio coding tools," in *Proceedings of ISCAS 2000*, 2000, vol. 3, pp. 201–204.

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

[11] 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.

[12] S.W. Golomb, R.E. Peile, and R.A. Scholtz, *Basic Concepts in Information Theory and Coding*, Plenum press, NY, 1994.

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

[14] Peter Noll, "MPEG digital audio coding," *IEEE Signal Processing Magazine*, vol. 14, no. 5, pp. 59–81, Sept. 1997.

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

[16] C.D. Giurcaneanu, I. Tabus, and J. Astola, "Adaptive context-based sequential prediction for lossless audio compression," *Signal Processing*, vol. 80, no. 11, pp. 2283–2294, November 2000.