Audio Engineering Society
Convention Paper
Presented at the 111th Convention
2001 September 21–24 New York, NY, USA

Efficient compression of oversampled 1-bit audio signals

Reiss, J. D. and Sandler, M. B.
Electronic Engineering, King's College,
University of London, Strand, London WC2R 2LS, U.K.
josh.reiss@kcl.ac.uk

ABSTRACT

Sigma delta modulation is a popular technique for high-resolution analog-to-digital conversion and digital-to-analog-conversion. It has been considered as a new format for recording and storage of audio signals. To reduce the storage capacity, a lossless compression scheme can be applied. However, this scheme offers less than 3:1 compression. This may not be sufficient for storage on media such as a Digital Versatile Disk (DVD). We propose a scheme based on a technique known as bit-grouping. Errors are introduced in the compression, but they are confined to frequencies outside the audible range. Our studies indicate that bit-grouping allows one to achieve greater than 4:1 compression.

INTRODUCTION

Sigma-delta (or delta-sigma) modulation is a popular method for high-resolution A/D and D/A converters. It is frequently used in audio processing and has a wide range of applications. Sigma-delta modulators operate using a trade-off between oversampling and low-resolution quantization. That is, a signal is sampled at much higher than the Nyquist frequency, typically with one bit quantization, so that the signal may be effectively quantized with a resolution on the order of 14-20 bits.[1] Recent work has concentrated on tone suppression[2, 3], multibit modulation[4] and chaotic modulation[5-7].

A new recording format for audio signals, Direct Stream Digital, employing 1-bit oversampling sigma-delta modulation, has recently been proposed and employed as an alternative to the currently widely used multi-bit recording format.[8] The oversampling rate is chosen to be $64 \cdot f_s$ with $f_s = 44.1\text{kHz}$. One of the drawbacks of sigma-delta modulation is the high oversampling rate. This results in a raw (uncompressed) audio data capacity which is typically 4 times as high as that needed for current CD signals. Thus compression becomes essential. However, the theoretical limits on lossless compression of 1-bit sigma delta signals are prohibitive. Fortunately, the signal is intended for audio. Thus only a narrow frequency range needs to be undistorted. This implies that some loss is allowable if it does not degrade the audio. Therefore it might be possible to modify the signal so that a far better compression ratio might be achieved, yet without causing a significant degradation of the output. This is the main focus of this paper.

BACKGROUND

The simplest, first order sigma-delta modulator consists of a 1-bit quantizer embedded in a negative feedback loop that also contains a discrete-time integrator. The analogue output is sampled at a frequency higher than the Nyquist frequency and is converted into a binary output. The system may be represented by the map[9]

$$U_n = U_{n-1} + X_{n-1} - Q(U_{n-1})$$

where X represents the analogue input signal and Q is the quantizer

$$Q(u) = \begin{cases} 1 & \text{if } u \geq 0 \\ -1 & \text{if } u < 0 \end{cases}$$

In this representation, $Q(U_{n+1})$ represents the quantization of input X_n. This system works by quantizing the difference between the input and the accumulated error. Thus when the error grows sufficiently large, the quantizer will flip in order to reduce the error. On average, the quantization output will be approximately equal to the input. Higher order modulators are typically used in commercial applications since they often yield improved signal-to-noise ratios. However, the essential structure: oversampling, quantization, and noise shaping, remains the same.

PULSE GROUP MODULATION

The average pulse repetition frequency (PRF) is defined as the reciprocal of the average time between consecutive rising edges of the pulse stream. The PRF of the output of a sigma-delta modulator depends on the oversampling ratio L and the composition of the limit cycles in the output. The maximum possible pulse repetition frequency of a SDM is $L \cdot f_s / 2$, which occurs for the repeating limit cycle 1,-1,1,-1,1... A straightforward method of reducing the Pulse Repetition Frequency of the SDM bitstream and forcing it to be constant is to group together samples with the same sign, so that after the sample-and-hold the transitions are reduced. This technique is sometimes referred to as pulse group modulation (PGM).[10] The output is divided into frames of length N and the samples in each frame are reordered so that all the 1s occur in a single group at the end. For instance, if N=4, then the sequence 0100101101101101 would become 0001011100110111.

To analyze the effect of the pulse grouping, consider a sequence $v(n)$ of 1-bit data from the SDM, assumed for clarity to have the values 0 and 1 rather than –1 and 1. At each sample instant, the sum of the present sample and the previous N-1 samples is taken.

$$y(n) = \sum_{k=0}^{N-1} v(n-k)$$

Or equivalently, in the z-domain:

$$Y(z) = \sum_{k=0}^{N-1} V(z) z^{-k} = V(z)\left(\frac{1-z^{-N}}{1-z^{-1}}\right)$$

$$M(z) = Y(z)/V(z) = \left(\frac{1-z^{-N}}{1-z^{-1}}\right)$$

thus the summation is equivalent to a moving average filter of length N. Every N^{th} sample of the summation corresponds in amplitude to the group size of each PGM pulse. The operation of taking every N^{th} sample and discarding the remaining samples is that of decimation, and the conversion to a pulse group is uniformly sampled pulse width modulation (PWM), which involves a sample rate increase by a factor N. The decimation produces aliassing and the PWM introduces harmonic distortion, carrier and sideband tones and intermodulation noise.

COMPRESSION USING PULSE GROUP MODULATION

For a sequence of M events, the best expected number of bits required to encode this sequence is given by the Shannon entropy,

$$H(P) = -\sum_{i=1}^{M} p_i \log_2 p_i$$

So, for truly random sequences, an N bit sequence could be in one of 2^N possible combinations, each with equal probability. That is, the best expected number of bits required to encode this sequence is $H(P) = -\sum_{i=1}^{2^N} \frac{1}{2^N} \log_2 \frac{1}{2^N} = N$. In other words, a pure random sequence with no redundancy can't be compressed. However, if pulse group modulation is applied and the N bits are ordered so that all 0s appear at the beginning and all 1s at the end, then there are only N+1 possible combinations. For N=4, they are 0000, 0001, 0011, to 0111, and 1111. Out of the 2^N combinations before ordering, $\binom{N}{k} = \frac{N!}{(N-k)!k!}$ of them have k 0s followed by N-k 1s. Therefore, frequencies can be assigned to the N+1 possible combinations that result from PGM. The combination with k 0s followed by N-k 1s occurs as output of an N-window from the PGM with an expected frequency of

$$f\left(\underbrace{00..0}_{k}\underbrace{11..1}_{N-k}\right) = \binom{N}{k} \Big/ 2^N \quad (1)$$

So, the best expected number of bits required to encode N symbols is

$$H(P) = -\sum_{i=0}^{N} \frac{\binom{N}{i}}{2^N} \log_2 \frac{\binom{N}{i}}{2^N}$$

and the minimum expected bits required per symbol for encoding is $-\frac{1}{N}\sum_{i=0}^{N} \frac{\binom{N}{i}}{2^N} \log_2 \frac{\binom{N}{i}}{2^N}$.

Thus, the application of an N-window PGM gives a best expected compression of

$$\text{Compression Ratio (expected)} = \frac{N}{-\sum_{i=0}^{N} \frac{\binom{N}{i}}{2^N} \log_2 \frac{\binom{N}{i}}{2^N}} \quad (2)$$

Since the high sampling rate ensures that the input to the modulator appears nearly constant over short time periods, the output of a sigma delta modulator is not expected to be random. For a nonrandom sequence, the worst case for the compression ratio occurs when each of the $\binom{N}{k}$ combinations of bit orderings occurs with equal probability. Then the expected number of bits required to encode N symbols is

PGM window size	Compressed file size (no PGM)	Compressed file size (PGM)	Compression ratio (PGM)	Predicted compression ratio (PGM)
1	100378	100378	1	1
2	100374	75301	1.333	1.333
4	100386	50980	1.969	1.970
6	100414	39050	2.570	2.571
8	100392	31940	3.142	3.144

Table 1. Compressed file sizes for a random sequence of 1s and 0s with and without pulse group modulation. The original file consisted of 800,000 points and had a file size of 100361 bytes.

$H(P) = -\sum_{i=0}^{N} \frac{1}{N+1} \log_2 \frac{1}{N+1} = \log_2(N+1)$, which provides a lower bound on an optimal compression scheme

$$\text{Compression Ratio (worst case)} = \frac{N}{\log_2(N+1)} \quad (3)$$

Equations (2) and (3) completely describe the effectiveness of PGM in a compression technique. In effect, (2) can be used to estimate the compression of pure random data and hence provides a rough lower bound on the compression of actual data. On the other hand, (3) provides an absolute lower bound, where this worst case performance would only be achieved in highly unusual circumstances.

It is important to note here that the 'expected number of bits' refers to the use of the best encoder possible. It is well known that arithmetic coding does very close to this.[11] Thus although many different encoding schemes are possible (Huffmann coding and predictive encoding are among the schemes that have been applied[12]), this work will concentrate on arithmetic coding and assume that the results of arithmetic coding are close to the theoretical limits.

Table 1 gives the results of the application of an adaptive arithmetic coding algorithm to a file of 800,000 binary outputs with and without the application of PGM. The procedure described in [13] was used to encode the data, and similar results were also obtained using a fixed frequency arithmetic encoding scheme based on the frequencies given by (1). A small amount of error is present in the resulting file sizes because of overhead in the arithmetic coding and slight variations in the determination of file size. There is little or no compression without the application of PGM. This is as expected because the input was a random sequence. For the case of PGM, the resulting file sizes are slightly larger than the predicted values from (2). This is because the arithmetic compression algorithm has a small amount of overhead. This can be reduced through a more efficient implementation of arithmetic coding. It is also worthwhile to note that this compression method can be easily implemented using integer arithmetic, variable frequency models and windowing. Thus it is ideally suited for use in realtime applications with simple circuits.

FEEDBACK IN PGM

An additional noise shaping (beyond that already in sigma delta modulation) can be applied that takes into account the pulse group modulation scheme. This is depicted in Figure 2. However, now the error incorporates aliasing noise, intermodulation noise and harmonic distortion, as well as the quantization noise. The PGM nonlinearity may then be modeled in the *z*-domain as an additive sequence $W(z)$

$$Y(z) = z^{-(N-1)}V(z) + W(z)$$

A more thorough discussion of the noise introduced by PGM and the appropriate feedback model is available in [10]. It is enough to note here that both theoretical and simulated results have been found which provide a thorough description of the noise introduced due to PGM, with or without feedback. For example, both theory and simulation confirm that, for a 64 times oversampled signal, feedback in an 8-window pulse group modulation scheme would result in an ~9 dB improvement in the SNR over the same system without feedback. Given knowledge of the SNR of a given sigma delta modulator and the specifications it must meet, it is relatively straightforward to estimate the maximum allowable PGM window size.

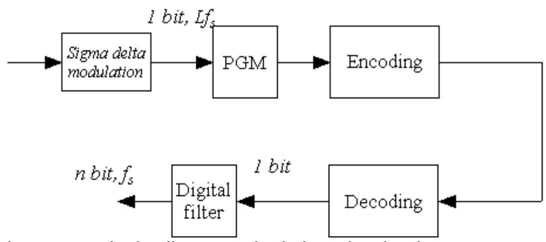

Figure 1. Block diagram depicting the basic structure of compression used in a 1 bit A/D converter.

PGM AND FILTER DESIGN

When applied to compression, pulse group modulation is sometimes more complicated than necessary. If the sigma delta modulator is used in A/D conversion, then the output of the modulator is typically passed through a digital filter in order to downsample and convert from 1 bit to multibit. Just prior to this stage is the most appropriate place for compression to occur (see Figure 1). This implies that the structure of the digital filter may dictate the preferred compression scheme. For instance, if the first stage of the filter is decimation through the use of a (nonoverlapping) moving average filter, i.e., N-tap Finite Impulse Response digital filter with coefficients of 1/N, then the ordering of the bits within each averaged window is irrelevant. Therefore pulse group modulation can be applied using the same window size as is used for the filter with *no effect* whatsoever on the downsampled data. In fact, if any FIR filter is used where 2 coefficients are equal, then those bits may be interchanged without affecting the output of the filter. Thus a bit reordering scheme specific to the symmetric properties of the filter is recommended in

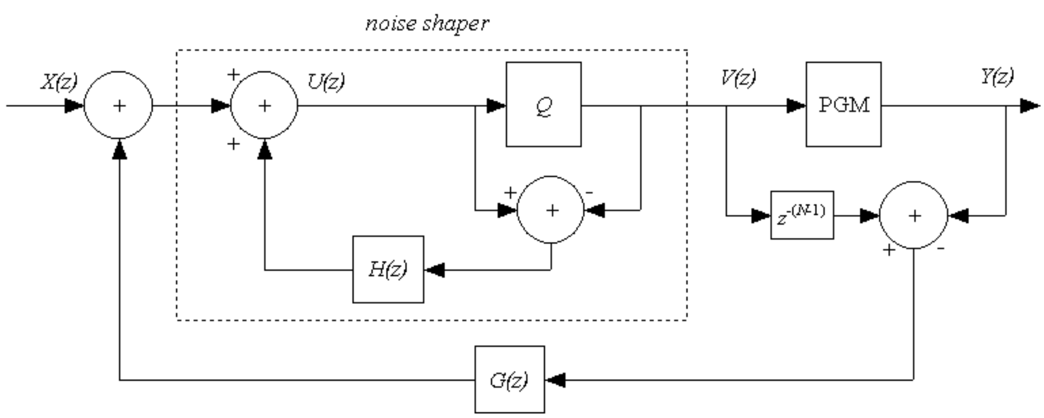

Figure 2. Model of pulse group modulation with feedback.

order to optimize compression. This applies without the use of feedback in the pulse group modulation scheme.

CONCLUSION

In this paper we have discussed a technique by which sigma delta bitstreams can be highly compressed. Under certain conditions, such as when averaging decimation filters are applied, this technique is effectively lossless. That is, the final output after decoding and filtering is exactly the same as if no compression had been applied. In other situations, a noise shaped feedback mechanism can be applied around the pulse grouper in order to reduce the noise and distortion. Since sigma delta signals are often grossly oversampled, this may be sufficient to make any losses through PGM based compression inaudible when this is used in a sigma delta converter for audio applications.

References

[1] W. Chou and R. M. Gray, “Dithering and its effects on sigma delta and multi-stage sigma delta modulation,” *IEEE Transactions on Information Theory*, vol. 37, pp. 500-513, 1991.

[2] V. Friedman, “The Structure of limit cycles in sigma delta modulation,” *IEEE Trans. Communication*, vol. 36, pp. 972-979, 1988.

[3] A. J. Magrath and M. B. Sandler, “Efficient Dithering of Sigma-Delta Modulators with Adaptive Bit Flipping,” *Electronics Letters*, vol. 31, 1995.

[4] S. J. Park, R. M. Gray, and W. Chou, “Analysis of a sigma delta modulator with a multi-level quantizer and a single-bit feedback,” in *Proceedings of the ICASSP 91*. Toronto, Canada, 1991, pp. 1957-1960.

[5] J. Reiss and M. B. Sandler, “The Benefits of Multibit Chaotic Sigma Delta Modulation,” *CHAOS*, vol. 11, 2001.

[6] O. Feely, “Nonlinear dynamics of discrete-time electronic systems,” *IEEE Circuits and Systems Society Newsletter*, vol. 11, pp. 1-12, 2000.

[7] O. Feely, “A tutorial introduction to non-linear dynamics and chaos and their application to sigma-delta modulators,” *Int. J. Circuit Theory and Applications*, vol. 25, pp. 347-367, 1997.

[8] D. Reefman and P. Nuijten, “Why Direct Stream Digital is the best choice as a digital audio format,” presented at 110th Audio Engineering Society Convention, Amsterdam, the Netherlands, 2001.

[9] R. M. Gray, “Oversampled Sigma-Delta Modulation,” *IEEE Transactions on Communications*, vol. COM-35, pp. 481-487, 1987.

[10] A. J. Magrath and M. B. Sandler, “Hybrid Pulse width Modulation / Sigma-Delta Modulation Power Digital-to-Analogue Converter.,” *IEE Transactions on Circuits, Devices and Systems*, vol. 143, pp. 149-156, 1996.

[11] T. M. Cover and J. A. Thomas, “Elements of Information Theory,” . New York: John Wiley & Sons, Inc., 1991, pp. 45.

[12] F. Bruekers, W. Oomen, R. van der Vleuten, and L. van de Kerkhof, “Improved Lossless Coding of 1-Bit Audio Signals,” presented at 103rd Convention of the Audio Engineering Society, New York, 1997.

[13] A. Moffat, R. Neal, and I. H. Witten, “Arithmetic Coding Revisited,” *ACM Transactions on Information Systems*, vol. 16, pp. 256-294, 1998.