DATA CONCEALMENT IN AUDIO USING A NONLINEAR FREQUENCY DISTRIBUTION OF PRBS CODED DATA AND FREQUENCY−DOMAIN LSB INSERTION.

Tio M. M. Cedric   Robertus Wahendro Adi   Ian Mcloughlin

School of Computer Engineering
Nanyang Technological University
Nanyang Avenue, Singapore 639798

Abstract: In this paper, a novel method of transparent data concealment in audio streams is discussed. The proposed system makes use of sub−band coding, least significant bit coding (LSB) and a Pseudo−Random Bit Stream Generator (PRBS). A maximum of about 6% of the audio file can be used to hide data transparently with no perceptible distortion. A solution to solve the positive bias problem that is inherent in LSB encoding is also presented.

I INTRODUCTION

Data hiding in the LSBs of audio samples in the time domain is one of the simplest algorithms to implement, but suffers by introducing audible distortion. Adjusting the LSBs of audio samples introduces noise. The proposed system aims to reduce the perceptibility of this noise by hiding the data in the LSBs in the frequency domain, of selected sub−bands. It aims to shape the spectral distribution of data to reduce any perceptible degredation.

Typical applications of such a system might be:

a) Embedding of descriptive song/track information into raw PCM audio signal without having to introduce a header.
b) Watermarking, for which encryption of the data to be hidden as well as a robust watermark insertion scheme are necessary.
c) Synchronisation triggering in digital multi−tracking recording systems.
d) Inserting secret data into in an audio signal.

II SYSTEM DESCRIPTION

2.1 The Algorithm.

The proposed algorithm conceals data in audio streams using sub−band coding, LSB coding and a shift register based data scrambler. The broad−band input audio signal is divided into 32 equal−width sub−bands by a polyphase analysis filterbank. The pre−randomized data is then inserted into the least significant bits (LSBs) of selected sub−bands. During LSB encoding, regular offsets are introduced in the encoded data to reduce positive bias. Once the data is hidden, the sub−bands are synthesized into a broadband signal again. At the decoding end, extraction of the hidden data involves analyzing the coded broad−band signal into sub−bands; extracting the appropriate LSBs with hidden data and unscrambling this information. The encode and decoder user identical decision algorithms and hence no identifier is required to flag presence of hidden data. A block diagram of the system is given in the Appendix.

2.2 The Near Perfect Reconstruction Filterbank.

The polyphase filterbank [3] used for the sub−band coding is adapted from the MPEG 1 standard [2]. Data hiding in the LSBs of the sub−bands is possible as a result of the near perfect reconstruction [4] properties of this filterbank. A 100% recovery of the hidden data in the sub−band LSBs is thus achievable. The analysis, synthesis and re−analysis process preserves the values of the LSBs of the sub−bands, although a delay of 32 samples in the broad band signal was found necessary for the filter−bank to stabilize and thereafter give the first correct sub−band output.

2.3 Pseudo−Random Shift Registers.

The insertion of data into the sub−band LSBs can result in an addition of audible frequencies to the original broadband audi. To reduce this, it is advantageous to randomize the data to be concealed prior to encoding resulting in audio characteristics that are less noticeable and more pleasant perceptually. The randomizer must be invertible so that the randomized data can be unscrambled to get the original hidden data from the sub−band LSBs. The chosen randomizer is a 24−bit shift register with 3 optimal tap positions [1] as shown in fig.1 for both encoder and decoder topologies. The repetition frequency for the pseudo−random bitstream is much less than 1Hz.

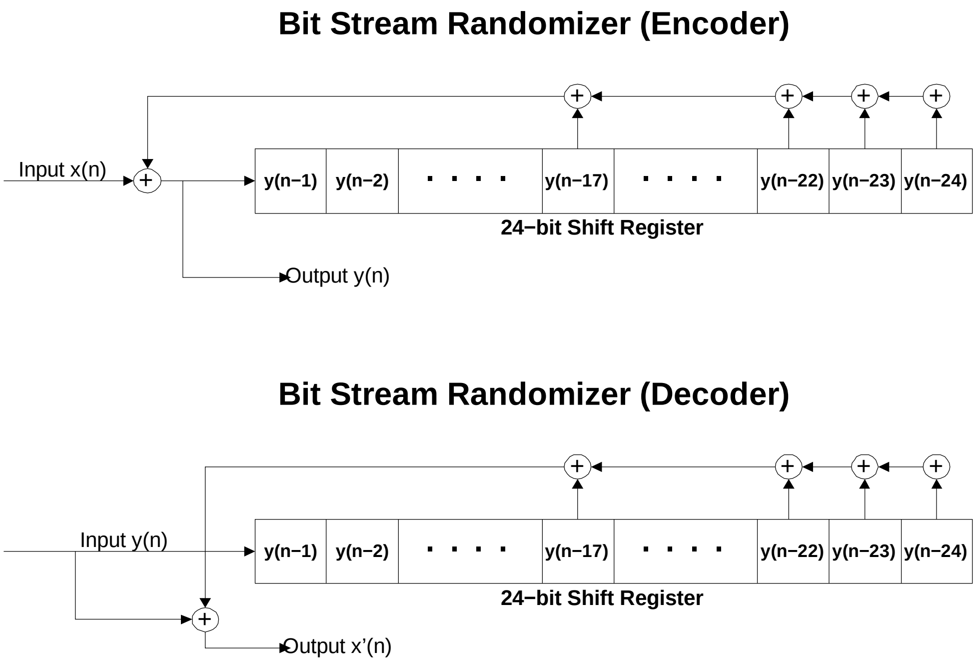

Fig. 1: Pseudo−random bitstream encoder and decoder.

2.4 Data Hiding.

The sub−bands whose LSBs will be used for data concealment were chosen based on their power and expected perceptual contribution.

For most audio streams, the highest sub−bands will have lowest signal power. Therefore, a simple method of choosing the sub−bands starting from the highest one was adopted, as shown in fig.2 which shows a stream of data distributed across the top 4 subbands.

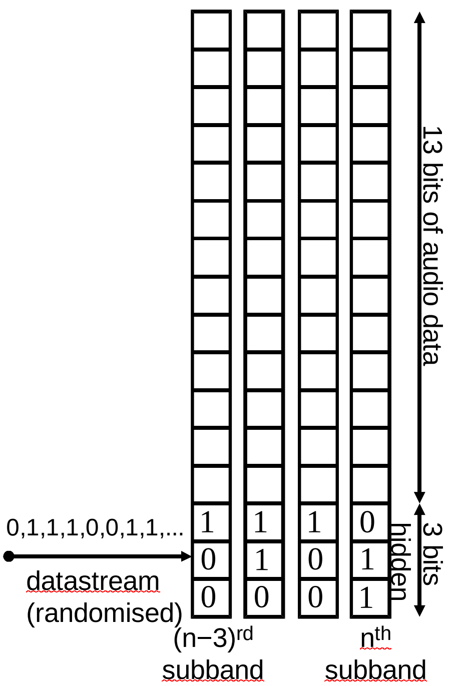

Fig. 2: Data Hiding in Sub−band LSBs.

III DISCUSSION

3.1 D.C. Bias in Silent Samples

The replacement of the LSBs by random bits is an unbiased operation when the audio signal is non−zero. However, when the audio signal is silent (i.e. has a stream of values of '0's), as is frequently the case in higher sub−bands, a positive DC bias will be introduced. Consider the following situation, the 32nd sub−band is silent (string of '0's). If hidden data is inserted into 3 LSBs of the 16−bit audio samples by replacing"000 with an unsigned 3−bit number, on the average, each audio sample of that sub−band will be increased by 3.5 steps out of the 65536 steps (16−bit numbers) available.

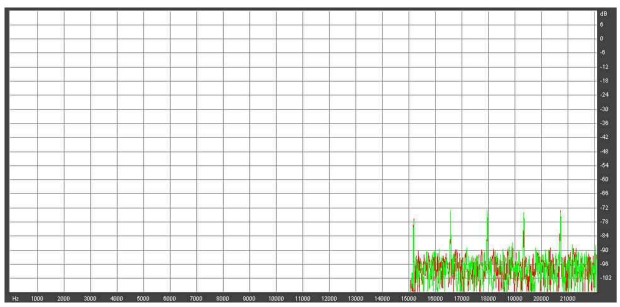

Fig. 3: Distortion (Positive bias) Introduced by Hidden Data in LSBs.

Figure 3 shows the spectrum of a silent broadband audio stream with some data concealed in 3 LSBs of the last 10 sub−bands. The hidden data introduces high frequencies into the spectrum of the silent broadband signal. After synthesis, these 0 Hz DC components of each affected sub−band will be modulated to a higher frequency in the broadband stream that corresponds to the lower−pass band frequency of the particular sub−band, resulting in possible perceptible tones.

3.2 Removing DC bias by Data Re−mapping.

To eliminate the DC bias in silent audio, we can re−map the data inserted into the LSBs such that the probability of inserting a positive number is the same as the probability of inserting a negative number. This will give an overall bias of“0 in the audio signal with hidden data since the hidden data is pre−randomised (equal number of 1’s and 0’s). This also implies that the insertion of the hidden data into the LSBs now requires an addition/subtraction operation rather than a replacement operation.

For a 3 LSBs implementation, the following mapping is used:

Original Hidden Data	Remapped Hidden Data (2’s Complement)
0000000000000000 (0)	0000000000000000 (0)
0000000000000001 (1)	0000000000000001 (1)
0000000000000010 (2)	0000000000000010 (2)
0000000000000011 (3)	0000000000000011 (3)
0000000000000100 (4)	1111111111111100 (−4)
0000000000000101 (5)	1111111111111101 (−3)
0000000000000110 (6)	1111111111111110 (−2)
0000000000000111 (7)	1111111111111111 (−1)

The re−mapped data to be hidden is added to the audio sample. This mapping system will result in an addition of positive numbers with a probability of 3/8 (numbers 1, 2 and 3) and an addition of negative numbers (numbers –1, −2, −3, −4) with a probability of 4/8. The remaining 1/8, which does not affect the bias, is the case when ‘0s are added to the sub−band samples. Such a system should give a negative bias since addition of negative numbers occurs more than addition of positive numbers (3/8 versus 4/8). However, the magnitude of this negative bias is much smaller than the case when the LSBs are simply replaced by the hidden data.

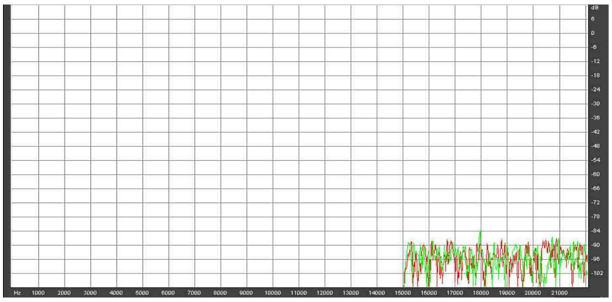

Fig. 4: Frequency Response of Broadband Signal after Re−map of Data.

Fig. 4 shows the frequency spectrum of an audio signal (identical to the sample used to generate the processed signal of fig. 3) with 3 LSBs used for data hiding but employing data re−mapping. Compared to fig. 3, the DC spikes are less pronounced.

To enable information recovery, the LSBs (those to be used for data hiding) of the original audio samples must be set to ‘0’ before the re−mapped data is added. Otherwise, the process will not be invertible and decoding would be impossible. Re−mapping of data is effective only when the audio signal is silent (all zeros). When the audio signal not silent, this re−mapping will introduce a DC component instead of eliminating it, due to the setting of the LSBs of the original audio signal to 0 prior to adding the remapped data. On average, however, the addition of the data will not introduce a DC component because an even number of positive and negative numbers are added to the audio signal.

The operation that adds a significant DC component is the setting of the LSBs to ‘0’ prior to adding the data. Fig. 5 shows the DC spikes introduced when the audio signal is non−silent.

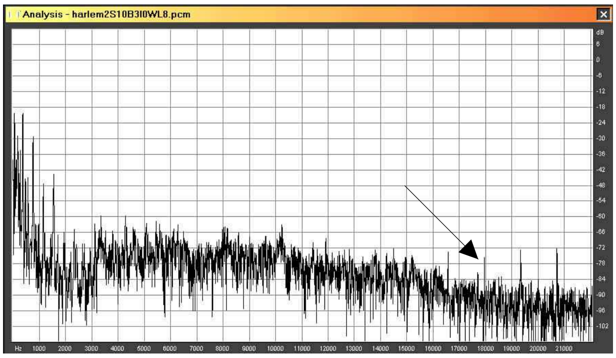

Fig. 5: DC Effect in Non−Zero Audio Signal when data is hidden in the LSBs.

3.3 Reducing the DC bias Effect by Dithering.

In order to reduce the DC bias effect in the encoded audio signal, the case when the audio signal is silent must be considered as a special case. The proposed solution to the DC bias effect makes use of dithering.

For every alternate audio sample which is silent (value=0), the ‘16−N’ Most Significant Bits (MSBs) are set to ‘1’, where N is the number of LSBs used for data concealment. The LSBs of the audio sample are then replaced by the data to be hidden. This effectively changes the encoded audio sample to a negative number and offsets the positive bias introduced by the replacement of the LSBs of a silent sample with the data. Consider the following example:

Original Audio Data	Dithered Audio Data
0000000000000000	0000000000000111 (7) (*replaced LSBs with 111*)
0000000000000000	1111111111111001 (−7) (*replaced LSBs with 001*)

The net bias of the above consecutive silent audio samples is −7 + 7 = 0. This will effectively reduce the DC bias caused by changing the LSBs of a silent signal.

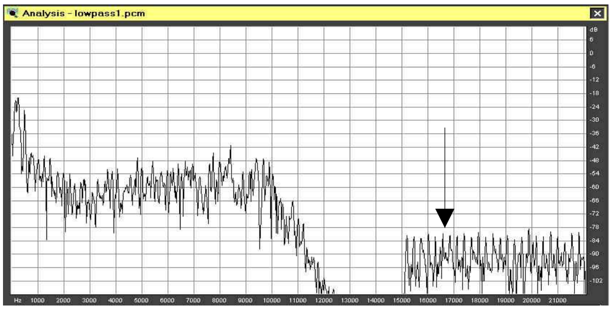

Fig. 6: Dithering reduces level of noise spikes.

Fig. 6 shows the effects of data hiding in the LSBs of sub−bands with dithering. It shows that dithering reduces the level of the individual DC spikes, although more spikes are created. This is a result of a better distribution of the distortion power. In any case, the distortion introduced by the data with dithering is a closer approximation of white noise than that without dithering as it is less tonal. The decoding of the system is unchanged by the dithering because the LSBs are not affected. Decoding simply involves the extraction of the LSBs.

IV PERCENTAGE OF HIDDEN DATA

A number of informal listening tests were conducted to establish the quanity degredation involved for differing numbers of affected subbands and bits per subband. It was found that a maximum of 5 LSBs can be used for data concealment in each of the last 10 sub−bands. For highest fidelity, 3 LSBs of 10 sub−bands is recommended for data hiding on the basis that the alteration is ruled as completely imperceptible from the tests. The percentage of hidden data relative to the total audio stream size can be computed as follows:

$$\% \text{ of hidden\_data} = \frac{\text{no. LSBs}}{16} \times \frac{\text{no. sub} - \text{bands}}{32} \times 100$$

$$= \frac{3}{16} \times \frac{10}{32} \times 100$$

$$= 5.85\%$$

V CONCLUSION

Data hiding in the LSBs of sub−bands reduces the audibility of distortion caused by the insertion, by frequency−shaping the error. With shaping and dithering, the error is transformed into a high frequency hiss that is fairly imperceptible. These enhancements refine the crude LSB encoding method into a frequency−based noise shaping method. Informal listening tests have shown that audio with data hidden in this manner is of high fidelity.

VI REFERENCES

[1] **Horowitz & Hill**, *The Art of Electronics*,publisher Cambridge University Press, (1989 edt).

[2] **ISO/IEC JTC 1/SC 29**, "Information Technology **-** Coding of Moving Pictures and Associated Audio for Digital Storage Media at up to about 1.5 Mbits/s **-** Part 3: Audio" ISO/IEC International Standard IS 11172-3:1993(E), (1st edt. 1993−08−01).

[3] **Vaidyanathan P P**, “Theory and Design of M−channel Maximally Decimated Quadrature Mirror Filters with Arbitrary M, having the Perfect Reconstruction Properties” *IEEE Trans. on ASSP*, vol. ASSP−35, (Apr 1987), pp 476−492.

[4] **Strang G & Nguyen T**, *Wavelets And Filter Banks* publisher Cambridge Press, (1996 edt).