Lossy Audio Compression

Ashutosh Mittal Navneet Agrawal

***Abstract*—The project aims to reduce bit rate of audio files from 64kbps to 32kbps using lossy compression with minimal reduction in audio quality. Filter banks are implemented to perform variable quantization over the different frequency components of the audio.**

I. Introduction

THE audio files can take up a large space in the hard disk or large bandwidth if being streamed online. Unless listener is really an audiophile, in most cases audio can undergo lossy compression to an extent but still sound just as before.

Investigating this possibility in this project, audio files will be dissected into three frequency bands, variable quantization of the data values will be performed, and then knitted again back into an audible file. Using optimized number of quantization bits, project aims to produce a trimmed audio file which is more or less like the original one. Value of Signal to quantization noise ratio (SQNR) for the compressed audio as well as manual inspection of difference in sound quality are the inspection methodologies used for the results.

II. Theory

A. Filter Bank

Design of the filter is paramount to performance of the compression technique. In order to be able to work on different spectral components independently, we need to isolate them first. We are using a filter bank design shown in fig. (1). The primary filter bank, which will divide the input signal into two spectral halves, consists of two filters H_1 and H_0 in parallel followed by down sampling. To reconstruct the signal, the decimated signals are upsampled, then passed through filter G_1 and G_0, and added together in the end. We need to design this primary filter bank in such a way that it leads to perfect reconstruction of signal.

1) Design: In order to achieve perfect reconstruction and avoid aliasing, following conditions on the filters are sufficient:

$$G_0(z) = H_1(-z) \tag{1}$$

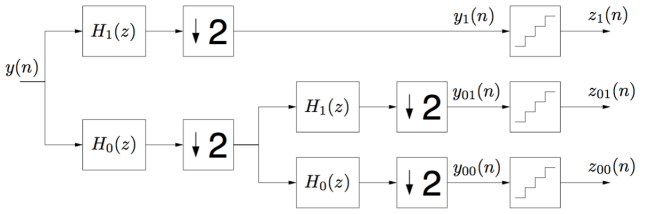
$$G_1(z) = -H_0(-z) \tag{2}$$

$$G_0(z)H_0(z) + G_0(-z)H_0(-z) = 2z^{-l} \tag{3}$$

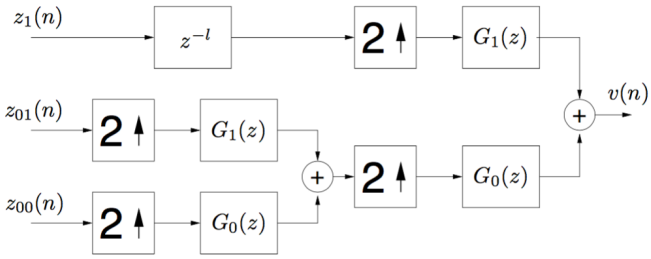
For real world implementation, filters need to be causal, so in order to ensure that equation (3) has a factor of $2z^{-l}$ which denotes the delay l between input and output. Following design parameters are taken for our filter bank.

$$G_0(z).H_0(z) = (1+z^{-1})^4.Q(z) \tag{4}$$

Fig. 1: Filter Bank Design



(a) Analysis Bank



(b) Synthesis Bank

$$Q(z) = p_0 + p_1z^{-1} + p_2z^{-2} \tag{5}$$

$$G_0(z) = \frac{1}{2}(1+z^{-1})^2 \tag{6}$$

Solving equations (1)-(6) following filters were obtained:

$$\begin{aligned} H_0(z) &= \frac{1}{4}(-\frac{1}{2} + z^{-1} + 3z^{-2} + z^{-3} - \frac{1}{2}z^{-4}) \\ G_0(z) &= \frac{1}{2}(1+z^{-1})^2 \\ H_1(z) &= \frac{1}{2}(1-z^{-1})^2 \\ G_1(z) &= \frac{1}{4}(\frac{1}{2} + z^{-1} - 3z^{-2} + z^{-3} + \frac{1}{2}z^{-4}) \end{aligned} \tag{7}$$

The frequency spectrum plot of the filters are shown in fig. (2).

2) Analysis Bank: A high pass and a low pass filter subdivides the input signal into two spectral halves. Branch with the high pass filter H_1 extracts frequency band corresponding to $[Fs/4, Fs/2]$, where Fs is the sampling frequency of the input. The other branch containing low pass filter H_0 extracts frequency band corresponding to $[0, Fs/4]$. Applying a low-pass or high-pass filter before down-sampling the signal should reduce the aliasing effect. Since our filters are not perfect, there will be some aliasing after down-sampling. However we have designed our filter bank in a way that after reconstruction the effect of aliasing will be completely removed. We further

divided $[0, Fs/4]$ into $[0, Fs/8]$ and $[Fs/8, Fs/4]$ bands by passing the low-pass filtered component into another primary filter bank. This will give us a total of three decimated signal $y00, y01, y1$ to work with.

For an input signal $y[n]$, the expression for the frequency domain representation of the decimated outputs y_1, y_{01}, y_{00} are given in the Appendix. In our experiments, we have used an audio sample with sampling frequency of 8 KHz. Hence, the frequency band of y_1 will be between 2 KHz to 4 KHz. This corresponds to high pitch sound in upper middle range of audible frequency. This range is often beyond the vocal range and defines the "presence" of music. y_{00} correspond to 0 to 1 KHz frequency range. Most vocals and bass instruments lie in this range. y_{01} correspond to 1 KHz to 2 KHz. This range contains higher pitch sounds such as vocals of a female vocalist or sound from instruments like flute.

3) Synthesis bank: The decimated signals are fed to the synthesis bank. Here the signals are upsampled by the factor of two and passed through corresponding filters (G_1 for high-pass and G_0 for low-pass) to reproduce the signal in its original spectral band. In order to keep the primary filter causal, we introduce a delay of l (equation 3). There is a delay of l between the input and the output signals due the design of primary filter bank. The low-pass component of the signal is again passed through a primary filter bank as shown in fig. (1). Hence we have to add a delay of l to the high-pass component (z_1) before upsampling to compensate for additional primary filter bank in the lower part. This delay of l gets upsampled by a factor of two during reconstruction. Thus the overall system has a delay of $L = l+2*l = 3*l$. The final output thus created gives a perfect (but delayed) reconstruction of the input signal.

B. Quantization Bit Allocation

We aim to halve the bit rate of audio input from 64 KBit/s to 32 KBit/s. The input has a sampling frequency of 8000 samples/s with each sample represented by 8 bits. In order to halve the bit rate of the audio file, bit allocation for the three bands should follow the relation:

$$b_{00} + b_{01} + 2\,b_1 = 2\,b \tag{8}$$

where b_{00} is bits per sample (bps) for $[0, Fs/8]$ band, b_{01} is for $[Fs/8, Fs/4]$, b_1 is for $[Fs/4, Fs/2]$ and b is for original audio file. Derivation behind equation (8) lies in the fact that b_1 corresponds to half of the total number of samples, where as both b_01 and b_00 correspond to only one fourth of the total number of samples respectively.

A naive way of audio compression to this effect could be representing each sample by 4 bits irrespective of its frequency band. However, utilizing the spectral dissection, we can allocate bits to each band separately while still holding up equation (8). This is an optimization problem. In general, Signal to quantization ratio (SQNR) gives a fairly good estimate of noise added due to quantization.

$$SQNR = \frac{E[input^2]}{E[(output - input)^2]} \tag{9}$$

It is the ratio of signal energy to that of quantization noise. It's an important index to analyze performance of the quantization algorithm. In order to maximize overall SQNR, higher amplitude components of input signal should have lesser quantization noise, hence represented by more bits. However, SQNR does not provide the best measure of perceptible quality of audio as human ear is not uniformly sensitive over the range of frequencies as well as intensities. Hence we present two different algorithms that analyze the data in frequency and time domains to allocate bits to given frequency bands.

1) Frequency domain: A measure of high amplitude content in a particular spectrum could be the power spectral density. In present algorithm, we allocate bits such that the ratio of number of quantization levels (2^{bits}) assigned to a spectrum is directly proportional to the maximum value of power spectral density (PSD) in that spectrum.

$$\frac{R_{00}}{2^{b_{00}}} = \frac{R_{01}}{2^{b_{01}}} = \frac{R_1}{2^{b_1}} \tag{10}$$

where, R_{00}, R_{01} and R_1 are maximum of PSD values in the three spectral domains. The maximum value of the PSD gives a fairly good estimate of the energy content in that spectrum. A peak in the PSD plot will be a sinusoidal with higher amplitude in the time domain which in turn gives the estimate of amount of high amplitude content in that frequency spectrum.

Using equations (8) and (10),

$$\begin{aligned} b_{00} &= \frac{1}{4}(16 + 3log_2(\frac{R_{00}}{R_{01}}) - log_2(\frac{R_1}{R_{01}}) \\ b_{01} &= b_{00} + log_2(\frac{R_{01}}{R_{00}}) \\ b_1 &= b_{01} + log_2(\frac{R_1}{R_{01}}) \end{aligned} \tag{11}$$

Respective bits thus obtained were rounded off to nearest integer values such that they still hold up equation (8).

2) Time domain: Another model for optimal bit allocation can be the one which accounts for the fact that human ear is more sensitive to high intensity sound. Number of samples in time domain corresponding to a spectral band, with intensity above a threshold will give a good measure to decide the bit allocation scheme. We define this threshold as the mean intensity for the entire audio track. The expression for bit allocation here is similar as in equation (11) except for the PSD values R_{00}, R_{01} and R_1, we will use the number of samples in time domain with intensity above the mean intensity in a given spectrum.

III. Numerical Results

Filter bank as described above provided perfect reconstruction of the input audio file if the quantization block is not added. However, it gave an overall delay of *nine* corresponding to designed delay in primary filter bank ($l = 3$), which is in accordance to what is theorized above. The two audio files examined here are *thank.wav* and *orinoccio.wav*. The spectral content of both of them are largely within the bands 0 to 2kHz (Fig. 3 to Fig. 5).

The provided audio had a bit rate of 64 Kbits per second with 8 bits quantization. We first investigate the quality of

audio obtained by directly reducing the quantization bits from 8 bits to 4 bits for the original signal. This, of course, gives a very bad quality compared to the compression method proposed using filter banks (Table 1 and 2). However, when we quantized as per the two algorithms specified in the theory, we got considerably better performance. The reconstructed audio was almost as good as the original one, except with halve the bit rate. Table 1 and 2 shows SQNR values for the respective audio files and the quantization algorithms.

TABLE I: Quantization Bit allocation for *thank.wav*

Allocation Method	b_{00}	b_{01}	b_1	SQNR (dB)
Compressing each sample	4 bps			19.4
Frequency based method	5	3	4	30.97
Time samples based method	6	6	2	31.20

TABLE II: Quantization Bit allocation for *orinoccio.wav*

Allocation Method	b_{00}	b_{01}	b_1	SQNR (dB)
Compressing each sample	4 bps			9.2
Frequency based method	6	4	3	99.47
Time samples based method	7	5	2	108.85

where b_{00}, b_{01} and b_1 are allocated bits corresponding to y_{00}, y_{01} and y_1 respectively.

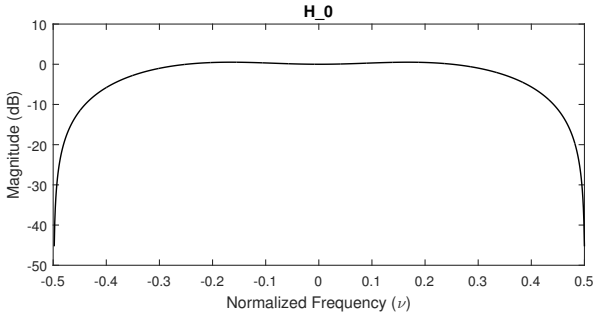
As it can be seen in the table I and II, using variable quantization we were able to improve the SQNR values and listening experience as compared to quantizing each sample with 4 bits. Although the given algorithms may not always give the highest SQNR possible, the audible quality of reconstructed samples is significantly high.The audio quality of samples with highest SQNR possible rather decreased comparatively as we could hear more noise in the background.

IV. Conclusion

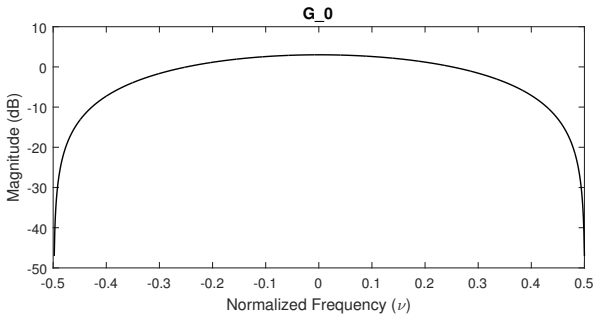
Usage of filter banks enabled us to variably quantize the audio and optimally assign bits to each of the spectral sub-band. It enabled us to test out two algorithms for quantizing audio inputs such that their bits rates are halved but the quality stays reasonably acceptable. These algorithm provide a better allocation scheme than 4 bits per sample for the entire audio. Even after losing half of the bits, audio was clearly understandable. No error was introduced by the filter bank even though non-ideal low/high pass filters were used. However, if the filters are closer to ideal ones, better SQNR could be achieved due to increased performance of the quantizer. The filter bank used in the project divides signal into 3 sub bands only. If more divisions are made, bit allocation could have been much more optimal and even higher SQNR could be achieved.

During the analysis, we also observed that SQNR is not the ultimate measure of audio quality. Psycho-acoustic model of the human ear enables perceptual coding to optimize in such a way that less bits can be assigned to aspects of audio which do not affect the listening experience. This effect can not be measured by SQNR. It only takes care of the quantization noise, not the listening experience.

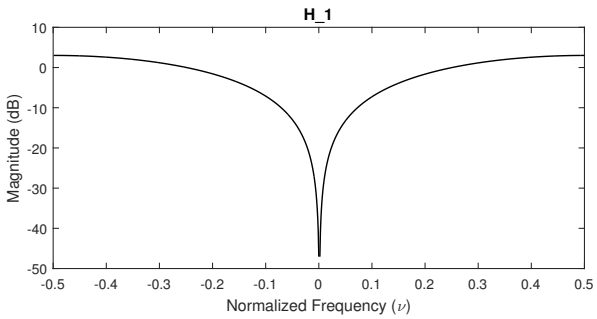
V. Appendix



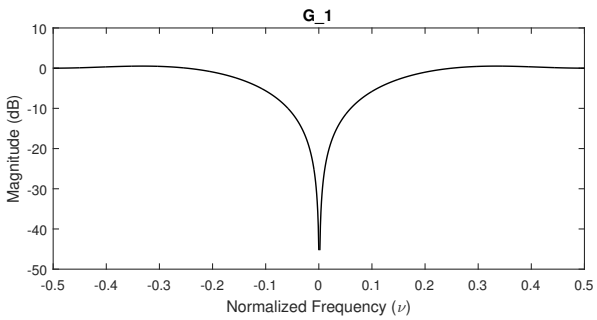
(a) H_0



(b) G_0

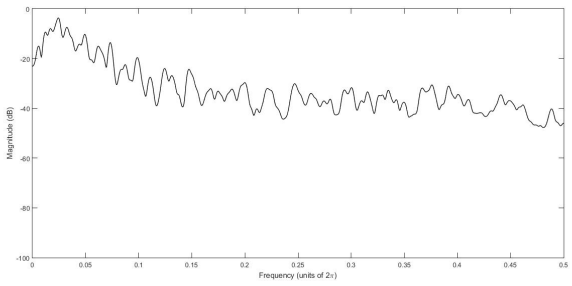


(c) H_1

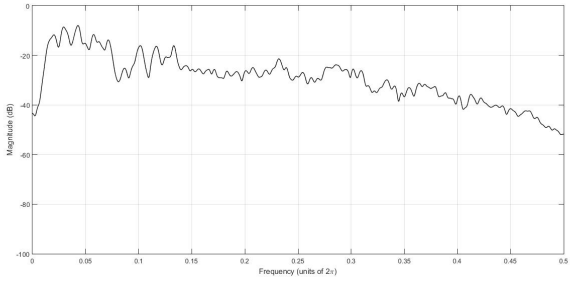


(d) G_1

Fig. 2: Filter Bank Design

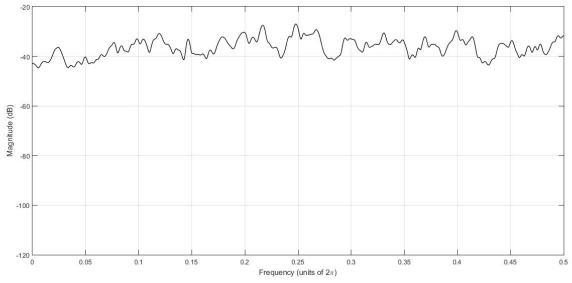


(a) $orinoccio.wav$

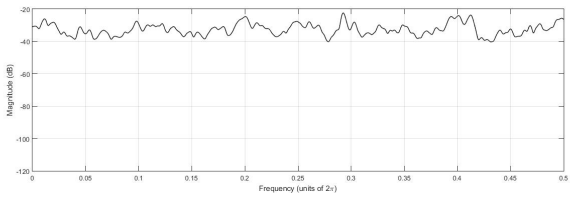


(b) $thank.wav$

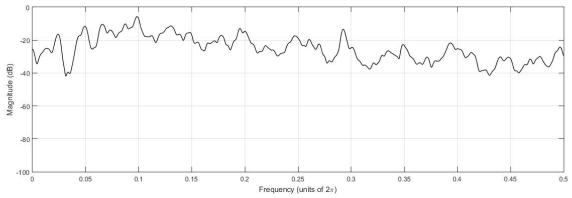
Fig. 3: Power spectrum of input (Fs=8KHz)



(a) Power spectrum of y_1 (Fs= $4KHz$)

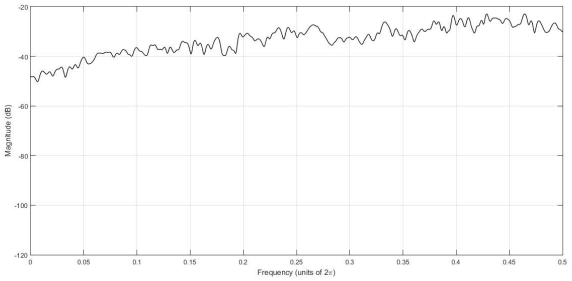


(b) Power spectrum of y_{01} (Fs= $2KHz$)

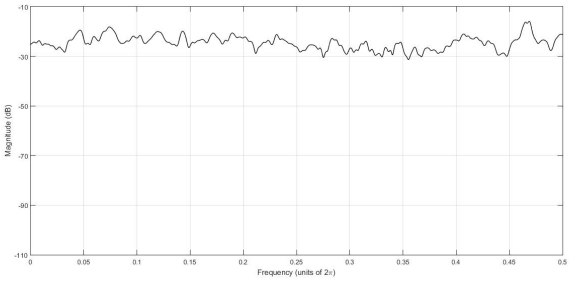


(c) Power spectrum of y_{00} (Fs= $2KHz$)

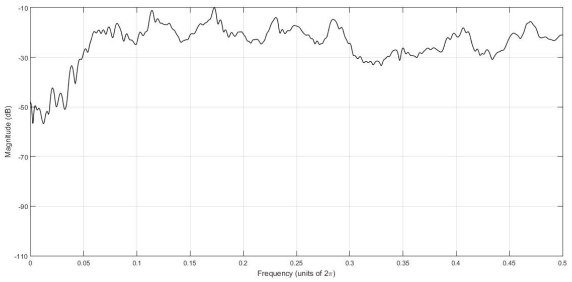
Fig. 4: Spectral sub-band signals of $orinoccio.wav$



(a) Power spectrum of y_1 (Fs= $4KHz$)



(b) Power spectrum of y_{01} (Fs= $2KHz$)



(c) Power spectrum of y_{00} (Fs= $2KHz$)

Fig. 5: Spectral sub-band signals of $thank.wav$

$$
\begin{aligned}
Y_1(\nu) =& \frac{1}{2}\Big[\, Y\big[\frac{\nu}{2}\big]H_1\big[\frac{\nu}{2}\big] + Y\big[\frac{\nu-1}{2}\big]H_1\big[\frac{\nu-1}{2}\big]\,\Big] \\
Y_{01}(\nu) =& \frac{1}{4}\Big[\, Y\big[\frac{\nu}{4}\big]H_0\big[\frac{\nu}{4}\big]H_1\big[\frac{\nu}{2}\big] + \\
& Y\big[\frac{\nu-1}{4}\big]H_0\big[\frac{\nu-1}{4}\big]H_1\big[\frac{\nu-1}{2}\big] + \\
& Y\big[\frac{\nu-2}{4}\big]H_0\big[\frac{\nu-2}{4}\big]H_1\big[\frac{\nu}{2}\big] + \\
& Y\big[\frac{\nu-3}{4}\big]H_0\big[\frac{\nu-3}{4}\big]H_1\big[\frac{\nu-1}{2}\big]\,\Big] \\
Y_{00}(\nu) =& \frac{1}{4}\Big[\, Y\big[\frac{\nu}{4}\big]H_0\big[\frac{\nu}{4}\big]H_0\big[\frac{\nu}{2}\big] + \\
& Y\big[\frac{\nu-1}{4}\big]H_0\big[\frac{\nu-1}{4}\big]H_0\big[\frac{\nu-1}{2}\big] + \\
& Y\big[\frac{\nu-2}{4}\big]H_0\big[\frac{\nu-2}{4}\big]H_0\big[\frac{\nu}{2}\big] + \\
& Y\big[\frac{\nu-3}{4}\big]H_0\big[\frac{\nu-3}{4}\big]H_0\big[\frac{\nu-1}{2}\big]\,\Big]
\end{aligned}
\tag{12}
$$

REFERENCES

[1] John G. Proakis,Dimitris K Manolakis *Digital Signal Processing* , 4th edition