Audio Compression Using FFT and MDCT

-Zishan Kazi

Basics of compression method

There are two types of audio compression:

1. Lossless compression: FLAC(Free Lossless Audio Codec), ALAC(Apple Lossless Audio Codec)
2. Lossy compression: ATRAC (Adaptive Transform Acoustic Coding), AAC (Advanced Audio Coding) and WMA (Windows Media Audio).

MP3 is the most well-known format which uses lossy data compression to encode data with discarding some of the data. MP3 compression reduces accuracy of certain components of sound that are beyond the hearing capabilities of most humans. Briefly, it means limiting high-frequency information and reducing the detail of or eliminating low-level signals. This is also known as **perceptual coding** or as **psychoacoustic modeling**.

MP3 compression can achieve upto 75% to 95% reduction in size.

MP3 Encoding algorithm:

The useful audio is stored with help of following algorithms:

1. FFT (Fast Fourier Transform):
   FFT is and algorithm that computes the Discrete Fourier Transform and its inverse. FFT gives same result as evaluating DFT directly, but FFT has much faster rate.

   DFT is given by following equation:

$$X(k) = \sum_{n=0}^{N-1} x(n) \times e^{-i \frac{2\pi nk}{N}}$$

   X[0], ...X[n] are complex numbers.
   k = 0,1...N-1

2. MDCT (Modified Discrete Cosine Transform):
   It is a linear function & is used as analysis filter bank with time domain alias cancellation property. It is a lapped transform, it was designed to be performed on larger, consecutive blocks of datasets where some parts of these blocks are overlapped.
   It transforms 2N real numbers into N real numbers with the following equations:

$$F : R^{2N} \to R^{N}$$

$$X_k = \sum_{n=-\infty}^{`2N-1} x_n \cos \frac{\pi}{N} \left(n + \frac{1}{2} + \frac{N}{2}\right)\left(k + \frac{1}{2}\right)$$

Implementation of the algorithm

This algorithm is implemented in the following steps (just a brief overview):

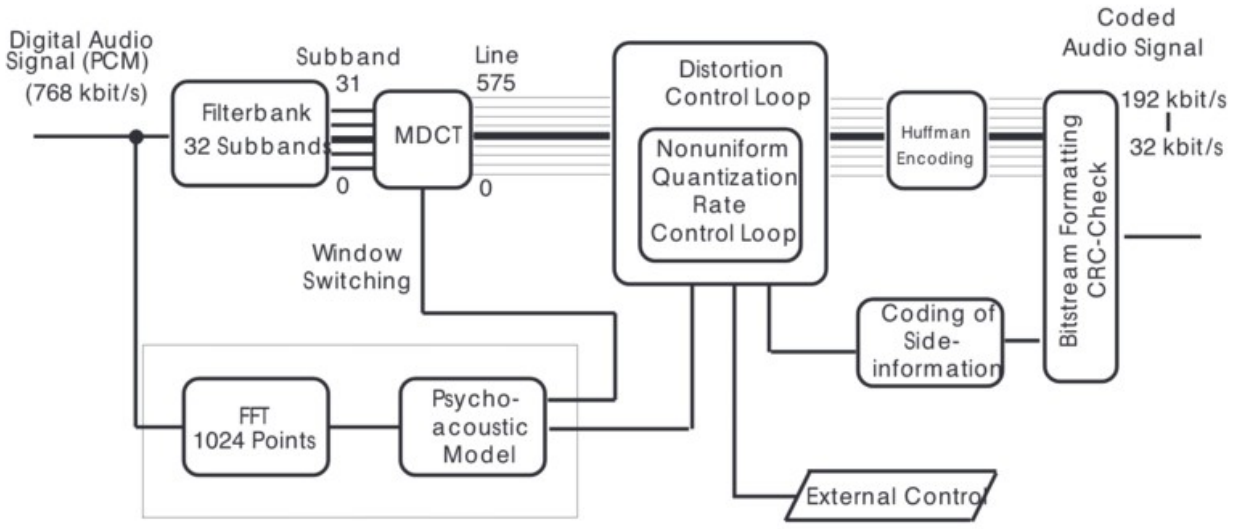
1. Divide the audio signal into smaller pieces called frames. Then an MDCT filter is performed on the output.

2. Passing the sample into 1024-point FFT and then psychoacoustic model is applied. Again a MDCT filter is performed on the output.

3. Noise allocation: quantification and encoding of each sample. It adjusts itself in order to meet the bit rate and sound masking requirements.

4. Formatting bitstream, also called audio frame.

These are just the major steps and are briefly explained. The actual procedure is vast and complicated.



How FFT and MDCT are used?

- **FFT**
  Used to filter out unwanted or unneeded data from sample.

1. Incoming audio samples, s(n), are normalized by equation:

$$x(n) = \frac{s(n)}{N(2^{b-1})}$$

where,
n = FFT length of sample

b = number of bits in sample

2. The masking threshold of sample is found by using an estimate of power density spectrum, P(k).

$$x(n) = P_k + 10\log\left[\sum_{n=0}^{N-1} h(n)x(n)exp\left(-j\frac{2\pi kn}{N}\right)\right]^2, \quad 0 \le k \le N\text{-}1$$

h(n) = Hann window
P(k) = found using 1024-point FFT.

- **MDCT**

The MDCT limits the sources of output distortion at the quantization stage (given in the figure). It is also used as analysis filter:

$$h_k(n) = w(n)\sqrt{\frac{2}{M}}\cos\left[\frac{(2n+M+1)(2k+1)(\pi)}{4M}\right]$$

The MDCT performs a series of inner products between the input data x(n), and the analysis filter hk(n). Finally, the equation can be written as:

$$x(n) = \sum_{k=0}^{M-1}\left[X(k)h_k(n) + X^P(k)h_k(n+M)\right]$$