Notes on audio coding

Lucio Bianchi

January 22, 2014

1 Non-uniform quantization

A uniform quantizer provides the same spacing between successive levels throughout the entire dynamic range of the signal. An approach frequently used in speech and audio coding is to use a non-uniform quantizer, which provides more closely spaced levels at the low signal amplitudes and more widely spaced levels at large signal amplitudes. A non-uniform quantization is often obtained by passing the signal through a non-linear device that compresses the signal amplitude, followed by an uniform quantizer.

The logarithmic compressor standard used in European telecommunication systems is called *A-law* and is defined as

$$c(x) = \begin{cases} \frac{1+\log(A|x|)}{1+\log(A)}\text{sign}(x), & \frac{1}{A} \leq |x| \leq 1 \\ \frac{A|x|}{1+\log(A)}\text{sign}(s), & 0 \leq |x| \leq \frac{1}{A} \end{cases} , \tag{1}$$

where $A = 87.7$.

In the reconstruction of the signal from the quantized values, the decoder employs an inverse logarithmic relation to expand the signal amplitude. The *A-law* inverse relation is given by

$$x(c) = \begin{cases} \frac{c(1+\log(A))}{A}, & 0 \leq |c| \leq \frac{1}{1+\log(A)} \\ e^{|c|(1+\log(A))-1} \cdot \frac{1}{A}\text{sign}(c), & \frac{1}{1+\log(A)} < |c| \leq 1 \end{cases} . \tag{2}$$

2 Adaptive quantization

PCM and DPCM encoders are designed on the assumption that the signal power is constant, and hence they employ a fixed quantizer. The efficiency and performance of these encoders can be improved by having them adapt to the time-variant power level of the signal. This lead to the use of an adaptive quantizer.

2.1 Feedforward adaptive quantizer

A feedforward adaptive quantizer adjusts its step size for each signal sample, based on a measurement of the input signal variance. For example, an algorithm that exploits a feedforward adaptive quantizer will have the following steps.

1. Collect a block of N samples: $x_B(n)$.

2. Estimate the signal variance inside the block: $\hat{\sigma}_x^2 = 1/N \sum_{i=1}^{N} (x_B(i))^2$.

3. Normalize samples in the block: $\hat{x}_B(n) = x_B(n)/\sqrt{\hat{\sigma}_x^2}$.

4. Quantize the normalized samples using a fixed quantizer designed for unit variance.

2.2 Feedback adaptive quantizer

A feedback adaptive quantizer employs the output of the quantizer in the adjustment of the step size. In particular, we set the step size as

$$\Delta(n) = \alpha(n-1)\Delta(n-1), \tag{3}$$

where the scale factor $\alpha(n)$ depends on the previous quantizer output. In particular, if the previous quantizer output is small, we will select $\alpha(n) < 1$ in order to provide for finer quantization. On the other hand, is the quantizer output is large, then the step size should be increased to reduce the possibility of signal clipping. An algorithm that has been succesfully employed in speech and audio coding adjusts the step size recursively according to the relation

$$\Delta(n) = \Delta(n-1)M_k(n-1), \tag{4}$$

where $M_k(n)$ is a multiplier whose value depends on the k-th quantizer level for the sample $x(n)$, and $\Delta(n)$ is the step size of the quantizer for processing $x(n)$.

2.2.1 Estimation of the multipliers

Consider an observation of N samples. An optimal set of multipliers M_k will satisfy

$$\prod_{k=1}^{2^R} M_k^{n_k} = 1, \tag{5}$$

where 2^R is the number of reconstruction levels and n_k is the number of times the k-th multiplier is used. By taking the N root of both sides we obtain

$$\prod_{k=1}^{2^R} M_k^{n_k/N} = 1, \tag{6}$$

from which we can define the frequency of occurrence $P_k = n_k/N$. Thus we can rewrite (6) as

$$\prod_{k=1}^{2^R} M_k^{P_k} = 1. \tag{7}$$

Equation (7) is a requirement on M_k values, but it has infinite solutions. In order to obtain an unique solution, we require a specific form on M_k:

$$M_k = \gamma^{l_k}, \quad l_k \in \mathbb{N}, \quad \gamma \in \mathbb{R}, \quad \gamma > 1. \tag{8}$$

This way we obtain

$$\prod_{k=1}^{2^R} \gamma^{l_k P_k} = 1 \quad \rightarrow \gamma^{\sum_{k=1}^{2^R} l_k P_k} = 1 \quad \rightarrow \sum_{k=1}^{2^R} l_k P_k = 0. \tag{9}$$

CODER	PCM			DPCM		
B	2	3	4	2	3	4
M_1	0.60	0.85	0.80	0.80	0.90	0.90
M_2	2.20	1.00	0.80	1.60	0.90	0.90
M_3		1.00	0.85		1.25	0.90
M_4		1.50	0.80		1.75	0.90
M_5			1.20			1.20
M_6			1.60			1.60
M_7			2.00			2.00
M_8			2.40			2.40

Figure 1: Multipliers for adaptive step size adjustment.

Thus, the knowledge of the PDF of the signal allows for an optimal computation of the multipliers M_k.

However, the estimation of the statistics of the input signal should on a block basis and results to a costly operation. Values of the multipliers optimized for speech encoding have been given by Jayant [1] and are reported in Fig.1 for both PCM and DPCM encoders.

3 Modified Discrete Cosine Transform

The analysis and synthesis MDCT pair is defined as

$$X(k) = \sum_{n=0}^{N-1} x(n) w_a(n) \cos\left(\frac{2\pi}{N}(n+n_0)(k+1/2)\right), \quad k = 0, \ldots, \frac{N}{2} - 1 \quad (10)$$

$$x(n) = w_s(n)\frac{4}{N}\sum_{k=0}^{N/2-1} X(k) \cos\left(\frac{2\pi}{N}(n+n_0)(k+1/2)\right), \quad n = 0, \ldots, N-1, \quad (11)$$

where $n_0 = (N/2+1)/2$ and $w_a(n)$, $w_s(n)$ are the analysis and synthesis windows, respectively. An usual choice is

$$w_a(n) = w_s(n) = \sin\left((n-1/2)\frac{pi}{N}\right). \quad (12)$$

The MDCT can be computed in an efficient way using the FFT.

In order to evaluate the analysis MDCT the steps are:

[1] N. S. Jayant, "Adaptive quantization with one-word memory," *Bell Syst. Tech. J.*, pp. 1119-1144, Sept. 1973.

1. Pre-processing:
$$\widetilde{x}(n) = x(n) \cdot e^{-j\pi n/N}; \tag{13}$$

2. N-point FFT:
$$\widetilde{X}(k) = \mathrm{FFT}_N\{\widetilde{x}(n)\}; \tag{14}$$

3. Post-processing:
$$X(k) = \mathrm{Re}\{\widetilde{X}(k) \cdot e^{-j2\pi k n_0/N}\}. \tag{15}$$

On the other hand, the synthesis MDCT is evaluated with the following steps:

1. Pre-processing:
$$\hat{X}(k) = X(k) \cdot e^{j2\pi k n_0/N}, \quad k = 0, \ldots, N-1; \tag{16}$$

2. N-point inverse FFT:
$$\hat{x}(n) = \mathrm{IFFT}_N\{\hat{X}(k)\}; \tag{17}$$

3. Post-processing:
$$x(n) = 2 * w_s(n) * \mathrm{Re}\{\hat{x}(n) \cdot e^{j\pi(n+n_0)/N}\}. \tag{18}$$