Least Squares Audio and Speech Compression

Linear Predictive Coding

Erik Sandström

ECE174, Department of Electrical and Computer Engineering
University of California, San Diego

November 7, 2017

Contents

1 Problem description and goals

The objective of this project is to compress the size of a sound signal sent from a transmitter and reconstructed by a receiver without affecting the perceived quality of the sound signal. The problem is relevant in order to optimize information flow over telecommunications networks specifically and in this project, the least squares optimization technique will be utilized in order to find the best approximative solution for the compression of audio data. `MatLab` is used as a tool to implement the concept.

2 Theory

To send any sound signal, the samples need to be quantized. Infinite precision of the samples is in reality impossible and a trade-off between audio quality and data size needs to be made. An uncompressed sound signal $y(n)$ can be quantized to $y_q(n)$ (direct quantization) and as a measurement of the error over the whole range of n, the Mean Squared Error (MSE),

$$MSE(y(n), y_q(n)) = \sum_{k=1}^{N} (y(k) - y_q(k))^2, \tag{1}$$

will be used, where N is the number of samples in the sound signal. Equation 1 will be used as a point of reference whether the compression is successful or not. Quantizing $y(n)$, which has a large range in magnitude, to high precision requires many bits r per sample. This is not an effective way of sending high quality audio so to compress the sound signal, $y(n)$ will be modeled using an IIR filter from the error signal $e(n)$ to the uncompressed sound signal $y(n)$ according to equation 2,

$$y(n) = \sum_{k=1}^{\ell} a(k)y(n-k) + e(n), \tag{2}$$

where $\ell = 10$ for the purpose of this specific model and the coefficients $a(k)$ are the filter coefficients. The initial conditions $y(0), y(-1), ..., y(-9)$ are set to zero. This is a resonable assumption since the first audio sample is $y(1)$ and the system is considered causal. The relationship in equation 2 can also be described using an FIR filter from the uncompressed sound signal $y(n)$ to the error signal $e(n)$ via equation 3 i.e. a linearly predictive model.

$$e(n) = y(n) - \sum_{k=1}^{\ell} a(k)y(n-k) \tag{3}$$

If $e(n)$ and $a(k)$ are known, then it is possible to recreate $y(n)$ using equation 2. This means that $e(n)$ and $a(k)$ contain as much information together as $y(n)$. The magnitude range of $e(n)$ will be small compared to $y(n)$ and therefore fewer bits should be needed when representing the quantized version $e_q(n)$ (residual

quantization) of $e(n)$ in order to get the same MSE for $e(n)$ and $e_q(n)$ as given when direct quantization is used.

The size of $e(n)$ will depend on how well the filter approximates $y(n)$ and since the filter is a weighted average filter of previous values, sudden quick jumps in amplitude of $y(n)$ will not be accounted for by the filter and large errors will develop. Over a limited range of values $y(n)$ can, however, be considered stationary and not vary too much in frequency and magnitude and $a(k)$ approximates $y(n)$ well. Each block will then have its separate set of filter coefficients. The blocks need to be small enough so that the stationary condition holds, but large enough to minimize unnecessary computations.

To calculate $a(k)$, $e(n)$ is pretended to be zero in equation 2 and the expression can be rewritten, for each block containing M elements, using vectors and matrices to the form

$$\underbrace{\begin{bmatrix} y_p(1) \\ y_p(2) \\ \vdots \\ y_p(M-1) \\ y_p(M) \end{bmatrix}}_{Y} = \underbrace{\begin{bmatrix} y_{p-1}(M) & y_{p-1}(M-1) & \dots & & y_{p-1}(M-9) \\ y_p(1) & y_{p-1}(M) & y_{p-1}(M-1) & \dots & y_{p-1}(M-8 \\ y_p(2) & y_p(1) & y_{p-1}(M) & \dots & y_{p-1}(M-7) \\ & & & \ddots & \\ & \dots & y_p(M-7) & y_p(M-8) & y_p(M-9) \end{bmatrix}}_{A} \underbrace{\begin{bmatrix} a_p(1) \\ a_p(2) \\ \vdots \\ a_p(9) \\ a_p(10) \end{bmatrix}}_{a}, \tag{4}$$

where p denotes the number on the M long block in the complete array $y(n)$. No exact solution exists for the inverse problem (since $e(n)$ is not zero), but the least squares solution minimizes $e(n)$. Since rank(A) $= \ell = 10$, A is one-to-one and there is a simple formula for the inverse problem (assuming that the weighting matrix W is the identity matrix)

$$A^+ Y = \hat{a}, \tag{5}$$

where

$$A^+ = (A^T A)^{-1} A^T \tag{6}$$

The complete idea is to use $\hat{a}(n)$ and calculate $e(n)$ using equation 3, quantize $e(n)$, transmit the signal and recreate the signal in the receiver using equation 2 which yields

$$\hat{y}(n) = \sum_{k=1}^{\ell} \hat{a}(k) y(n-k) + e_q(n), \tag{7}$$

where $\hat{y}(n)$ denotes the recreated signal. The corresponding MSE

$$MSE(y(n), \hat{y}(n)) = \sum_{k=1}^{N} (y(k) - \hat{y}(k))^2 \tag{8}$$

will then be compared with the result given by equation 1.

3 Method and Results

The blocksize is chosen to $M = 160$ elements per block. For a sampling frequency of 44 100 Hz, which is used, this means that 160 samples translates to 3.6 ms. The human voice membrane is expected to be stationary for at least 20 ms which means that the frequency output is constant during this time. Therefore, the choice of block size $M = 160$ is too good, but for the purpose of the project, 160 elements per block will still be used.

3.1 Quantization

An uncompressed sound signal is first normalized to magnitude values within the range ± 1 and to increase robustness of the signal, sample points that exceed the mean value $\pm$ $\alpha\cdot$ standard deviation within each block are truncated to the mean value $\pm$ $\alpha\cdot$ standard deviation respectively. The real scalar α, which can be choosen freely, dictates what samples will be truncated. A visualization of the effect of the truncation is given by figure 1 for the case with $y(n)$. Higher values of α means that more of the original signal is kept.

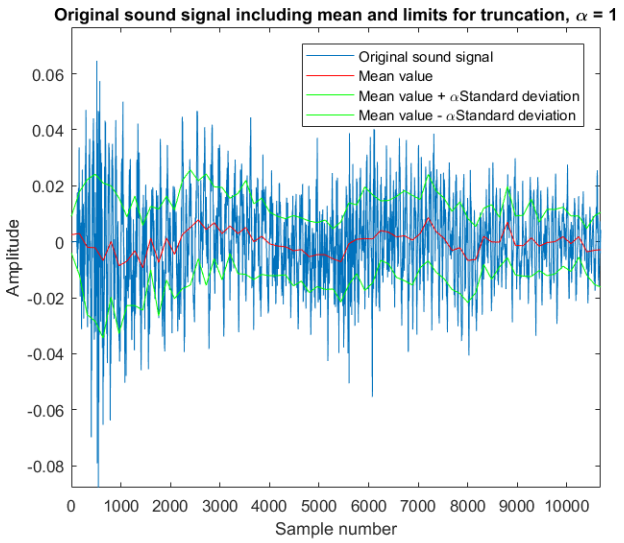

Figure 1: A small piece of the uncompressed sound signal visualized with the truncation process. Please note that the mean and truncation limits are the same for each 160 element long block and do not vary continuously along the sampling numbers, but connecting the dots makes for a more visualy intuitive illustration.

Next, the truncated version of the signal is quantized using r bits per sample equaling $L = 2^r$ quantization levels. It is also crucial to choose an effective

quantization interval value q, where q denotes the difference in magnitude between each quantization level. To limit the quantization values between the truncated values,

$$q = \frac{max(p(n)) - min(p(n))}{L - 1}, \tag{9}$$

where p is any truncated sequence. The quantized value is then retrieved by,

$$p_q(n) = \texttt{round}\Big(\frac{p(n)}{q}\Big)q, \tag{10}$$

where `round` is the corresponding `MatLab`-function.

3.2 MSE-error

For the direct quantization case the MSE according to equation 1 is calculated for different values of α and r and similarly for the residual quantization case the MSE according to equation 8 is calculated for the same values of α and r. The result is summarized in figure 2.

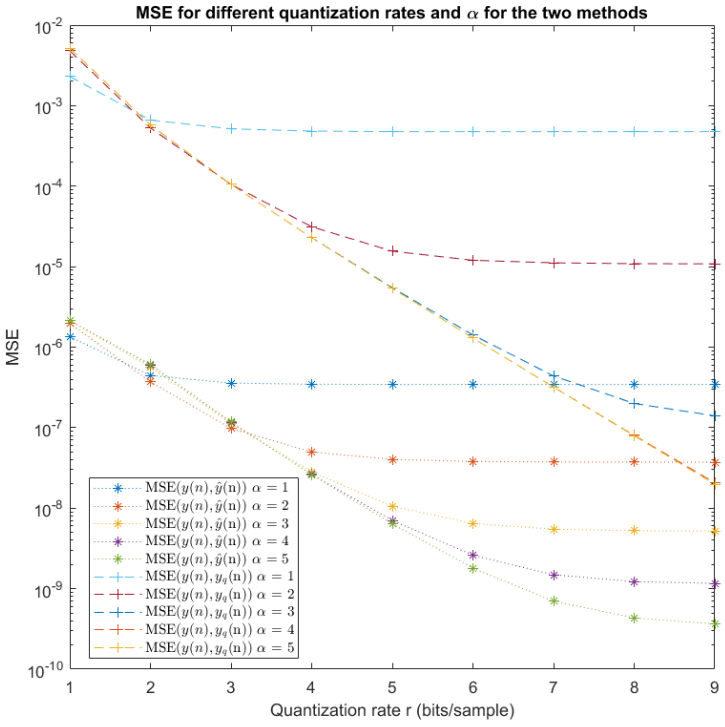

Figure 2: The MSE for the direct and residual quantization cases for different values of α and r.

For the direct quantization case, when the bit rate per sample $r \leq 3$, small values of α yield a smaller MSE. For $r > 3$, higher values of α yields smaller MSE, but only to a certain point. This phenomenon can be seen in figure 2, where the graph for $\alpha = 4$ for the direct quantization case precisely overlap, which is why it is not visible. For the residual quantization case, the trends are very similar with an even more emphasized effect for $r \leq 3$ where small values of α yields a smaller MSE compared to higher values of α. One can also note that for both cases, for every value of α, there seems to be a point where r does not affect the MSE anymore. This is especially clear for the case $\alpha = 1$ in figure 2. Generally the residual quantization method generates about 10^3 times smaller MSE than the direct quantization method, but for very large values of r, the direct quantization technique actually catches up and at $r = 9$, the direct quantized version of $\alpha = 5$ has the steepest slope of all graphs in figure 2.

After listening to the sound signal for different values of r and α, one can conclude that the MSE value is a very good estimate of the perceived sound quality of the signal. At an MSE of around 10^{-6}, the sound quality is so good, that it is practically impossible to hear any difference compared to the original signal. As a final note, the quality of the signal for a given MSE is perceived to be identical between the direct quantization case and the residual quantization case.

In reality, the filter coefficients $a(k)$ also need to be quantized. After quantization of $a(k)$ the MSEs for the newly reconstructed y called $\hat{y}_a(n)$ and the uncompressed signal $y(n)$ are plotted resulting in figure 3.

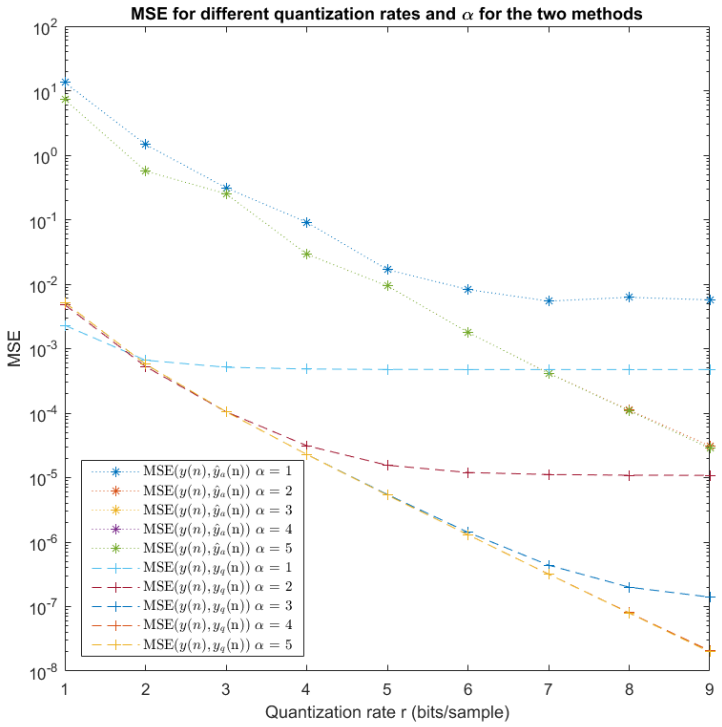

Figure 3: The MSE for the direct and residual quantization cases for different values of α and r. Now, the filter coefficients have also been quantized, denoted by the subscript "a".

As is evident from figure 3, when the filter coefficients are quantized, the residual quantization method can not compete at all against the direct quantization method.

4 Discussion

The reason why it is favorable to use smaller values of α when $r \leq 3$ for both the direct and residual quantization cases depends largely on the fact that it becomes more important to choose the quantization levels "visely" when r is small, since fewer quantization levels exist. If outliers are not removed properly before quantization, the magnitude difference between the quantization levels will increase and as a result this can have a substantial effect on the MSE. It is true that the outlier values will contribute to a greater error when the signal is truncated before quantization, but since the outlier values are so few compared to the rest of the signal, the total contribution will be very small. It is thus

more important to facilitate small errors between the original values and the quantized non-outlier values, which is done by keeping α small (i.e. $\alpha = 1 - 2$).

For values of $\alpha \geq 4$, the original signal is not truncated at all. This can be seen in figure 4, where a small section of the original signal has been plotted together with the limits of truncation and mean value. Since the original signal is contained within the green curves, no values will be truncated and no outliers will subsequently be removed. Higher values of α will not make any difference and thus, the graphs for high values of α overlap which can be seen in figure 2, especially for the direct quantization case. One should note that, in reality, the mean value and truncation limits do not vary continuously across the blocks as seen in figure 4. Instead, the values are constant within a block and jump to the next block. The above discussion still hold and more convenient coding was achieved by plotting this way.

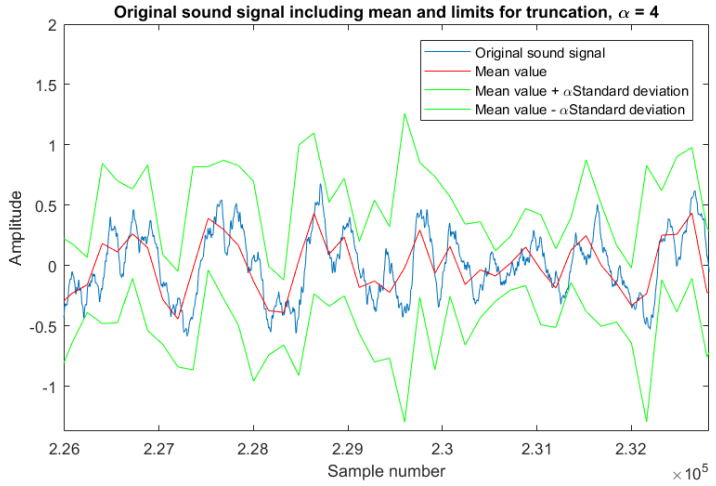

Figure 4: A small piece of the uncompressed sound signal together with mean values and truncation limits. For $\alpha = 4$ it is apparent that no values are truncated.

It is of importance to discuss why the MSE seems to converge to a specific value in some cases and there is a simple solution. For the direct quantization case, when no truncation of the original signal occurs prior to quantization, the quantized version of $y(n)$ will converge to the correct solution as r grows large. This is why the direct quantization technique can match the residual quantization technique for large values of r. If $y(n)$ on the other hand is truncated prior to quantization, no matter how high r is, the outlier values vill always contribute to a certain error in the MSE, which is why the MSE converges to specific values as can be seen in figure 2. For the case of residual quantization in figure 2, all graphs seem to converge in some way. The simple reason for this is that even

for $\alpha = 5$, the original error signal is still truncated to some extent.

To summarize, the residual quantization technique performs much better for small bit rates, which is what is needed for compression, than the direct quantization technique. To construct a compressed signal with equal quality as the uncompressed signal, the direct quantization technique requires approximately six times more bits per sample, which is substantial.

The bad performance of the residual quantization technique when the filter coefficients $a(k)$ are quantized suggests that it is essential construct a more elaborate quantization approach for the filter coefficients.

5 Conclusion

Despite the fact that the residual quantization technique performs badly when the filter coefficients are quantized, there is substantial potential in the technique when it comes to cost savings since the bit rate could at best be reduced to a sixth of the direct quantization case. There remains work to be done regarding refining the quantization algorithm and it is also important that the computational efficiency to perform the residual quantization is at least comparable to the direct quantization case.