Experiment 2: Quantization

Digital Signal Processing Lab

Avinab Saha,

Roll: 15EC10071, Group: 28

06th, March 2018

Quantization

Quantization, in mathematics and digital signal processing, is the process of mapping input values from a large set (often a continuous set) to output values in a (countable) smaller set. Rounding and truncation are typical examples of quantization processes. Quantization is involved to some degree in nearly all digital signal processing, as the process of representing a signal in digital form ordinarily involves rounding. Quantization also forms the core of essentially all lossy compression algorithms.

The difference between an input value and its quantized value (such as round-off error) is referred to as quantization error. A device or algorithmic function that performs quantization is called a quantizer. An analog-to-digital converter is an example of a quantizer.

In this assignment, we study Uniform Quantization, A-Law and Mu-Law Transformation based quantization as well as Quantization after adding noise.

Part 1 : Uniform Quantization With A Law Encoding and Decoding

An A-law algorithm is a standard companding algorithm, used in European 8-bit PCM digital communications systems to optimize, i.e., modify, the dynamic range of an analog signal for digitizing. It is one of two versions of the G.711 standard from ITU-T, the other version being the similar μ-law, used in North America and Japan.

For a given input x, the equation for A-law encoding is as follows,

$$F(x) = \operatorname{sgn}(x) \begin{cases} \frac{A|x|}{1+\log(A)}, & |x| < \frac{1}{A} \\ \frac{1+\log(A|x|)}{1+\log(A)}, & \frac{1}{A} \leq |x| \leq 1, \end{cases}$$

where A is the compression parameter. In Europe, A=87.6

A-law expansion is given by the inverse function,

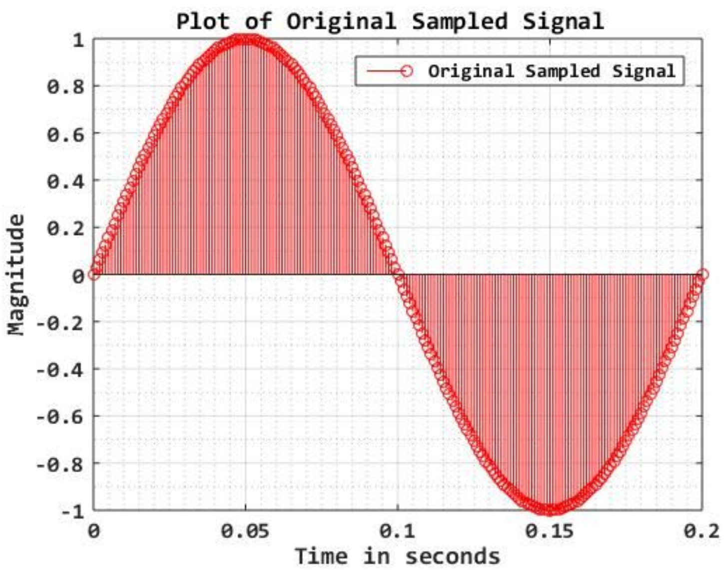
$$F^{-1}(y) = \operatorname{sgn}(y) \begin{cases} \frac{|y|(1+\ln(A))}{A}, & |y| < \frac{1}{1+\ln(A)} \\ \frac{\exp(|y|(1+\ln(A))-1)}{A}, & \frac{1}{1+\ln(A)} \leq |y| < 1. \end{cases}$$

The reason for this encoding is that the wide dynamic range of speech does not lend itself well to efficient linear digital encoding. A-law encoding effectively reduces the dynamic range of the signal, thereby increasing the coding efficiency and resulting in a signal-to-distortion ratio that is superior to that obtained by linear encoding for a given number of bits.

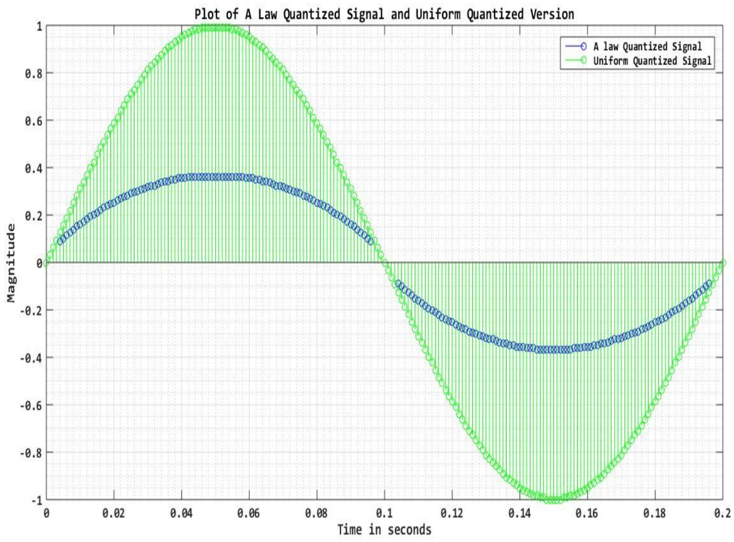
Error Comparison:

Specification	Error Amount
Uniform Quantization	5.5031e-06
Uniform Quantization with A Law Encoding and Decoding A=1	0.0100
Uniform Quantization with A Law Encoding and Decoding A=2	0.1695
Uniform Quantization with A Law Encoding and Decoding A=5	0.1822
Uniform Quantization with A Law Encoding and Decoding A=10	0.1860
Uniform Quantization with A Law Encoding and Decoding A=50	0.1912
Uniform Quantization with A Law Encoding and Decoding A=87.6	0.1923

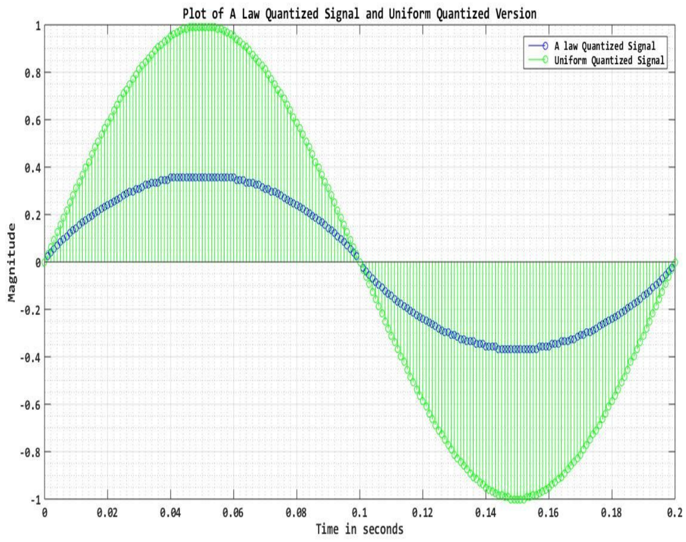
Output Waveforms:



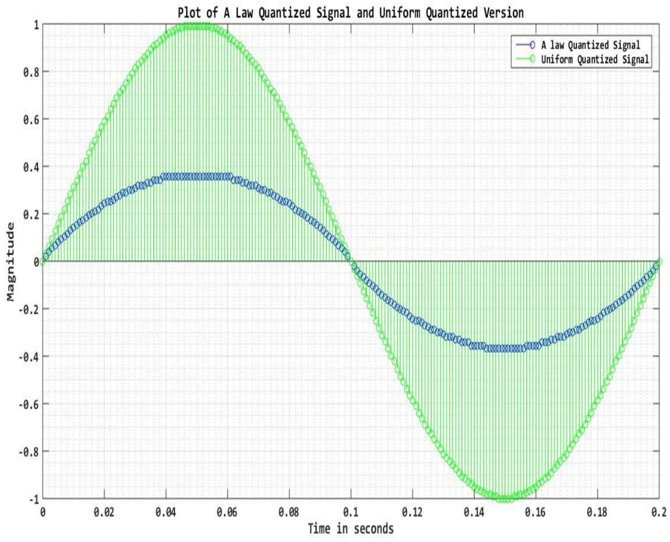
Input Sampled Signal



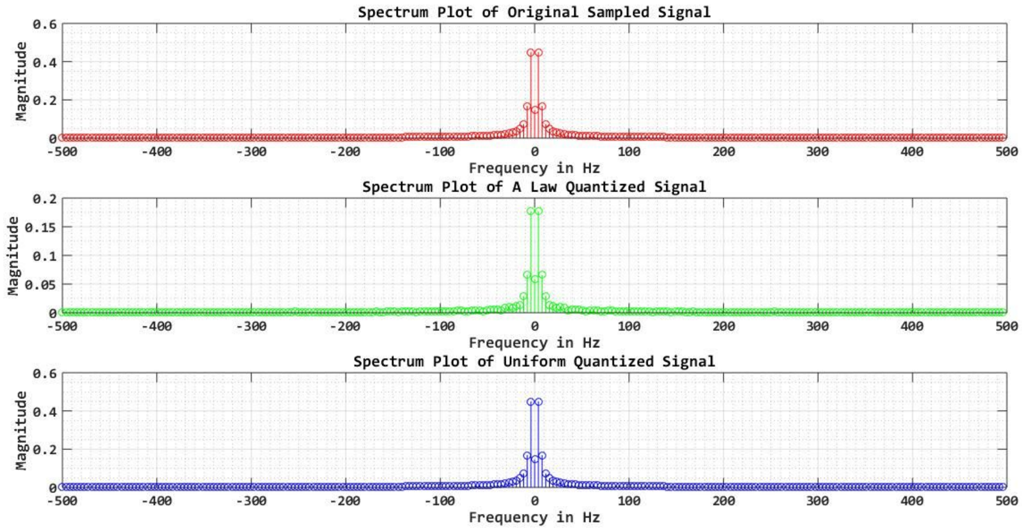
Comparison of Quantization With and Without A Law Companding with A=10



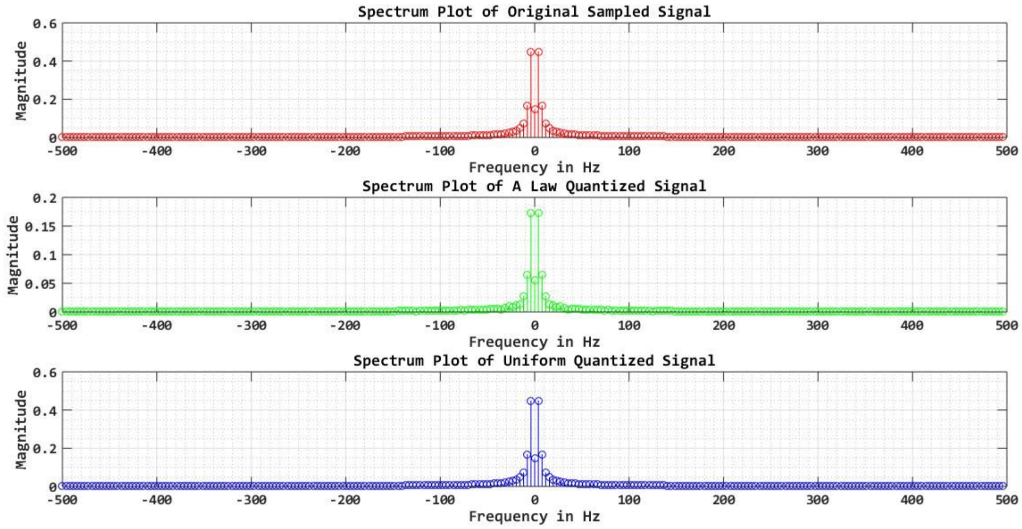
Comparison of Quantization With and Without A Law Companding with A=50



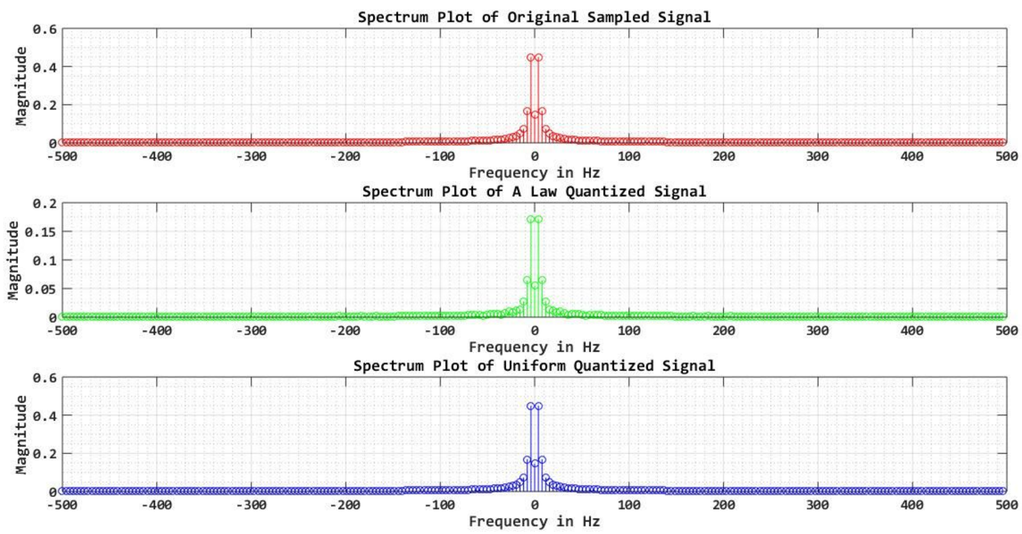
Comparison of Quantization With and Without A Law Companding with A=87.6



Spectral Comparison of Sampled Signal, Quantization With and Without A Law Companding with A=10



Spectral Comparison of Sampled Signal, Quantization With and Without A Law Companding with A=50



Spectral Comparison of Sampled Signal, Quantization With and Without A Law Companding with A=87.6

Discussions:

We observed that the error increased with an increase in the compression factor A. This can be attributed to the fact that a greater compression would result in some additional loss of information due to data compaction thereby leading to a higher error. However, a larger value of A would mean lesser requirement of memory for storing the data. Hence, there is a trade-off between the minimising storage volume via compression and data accuracy and one of these parameters have to be compromised to realise the benefit of the other one.

We also sketched the DFT of the original signal along with the signals obtained by the direct uniform quantisation and A-law companding. No appreciable distortion in the frequency spectrum of the all the signals.

Part 2 : Uniform Quantization With Mu- Law Encoding and Decoding

The µ-law algorithm is a companding algorithm, primarily used in 8-bit PCM digital telecommunication systems in North America and Japan. It is one of two versions of the G.711 standard from ITU-T, the other version being the similar A-law, used in regions where digital telecommunication signals are carried on E-1 circuits, e.g. Europe.

Companding algorithms reduce the dynamic range of an audio signal. In analog systems, this can increase the signal-to-noise ratio (SNR) achieved during transmission; in the digital domain, it can reduce the quantization error (hence increasing signal to quantization noise ratio). These SNR increases can be traded instead for reduced bandwidth for equivalent SNR.

For a given input x, the equation for µ-law encoding is

$$F(x) = \operatorname{sgn}(x)\frac{\ln(1+\mu|x|)}{\ln(1+\mu)} \quad -1 \leq x \leq 1$$

where µ = 255 (8 bits) in the North American and Japanese standards. It is important to note that the range of this function is −1 to 1.

µ-law expansion is then given by the inverse equation

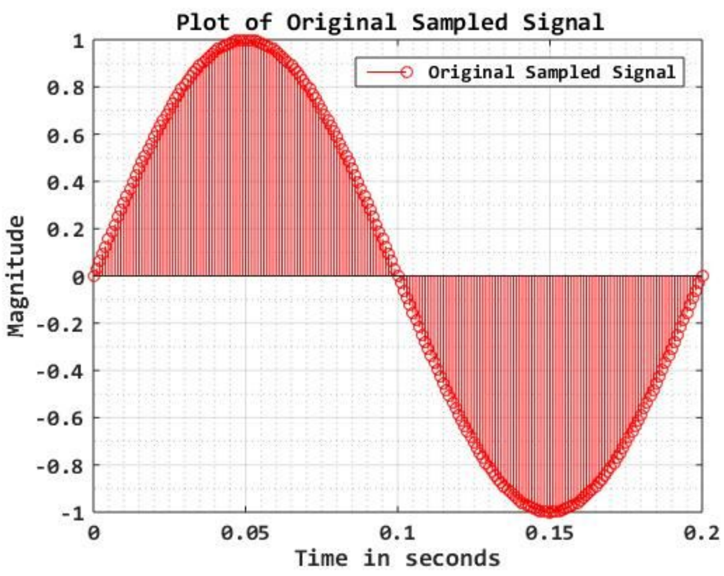
$$F^{-1}(y) = \operatorname{sgn}(y)(1/\mu)((1+\mu)^{|y|} - 1) \quad -1 \leq y \leq 1$$

Error Comparison:

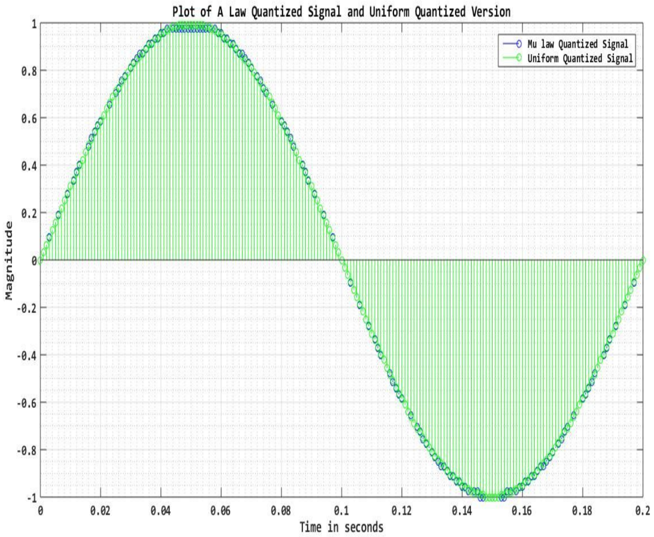
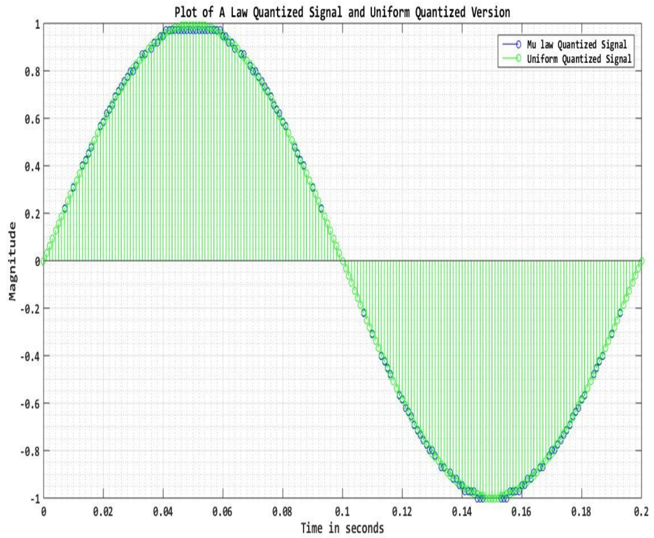
Specification	Error Amount
Uniform Quantization	5.5031e-06
Uniform Quantization with µ Law Encoding and Decoding µ=7	2.3050e-05
Uniform Quantization with µ Law Encoding and Decoding µ=15	3.7949e-05
Uniform Quantization with µ Law Encoding and Decoding µ=31	5.7574e-05

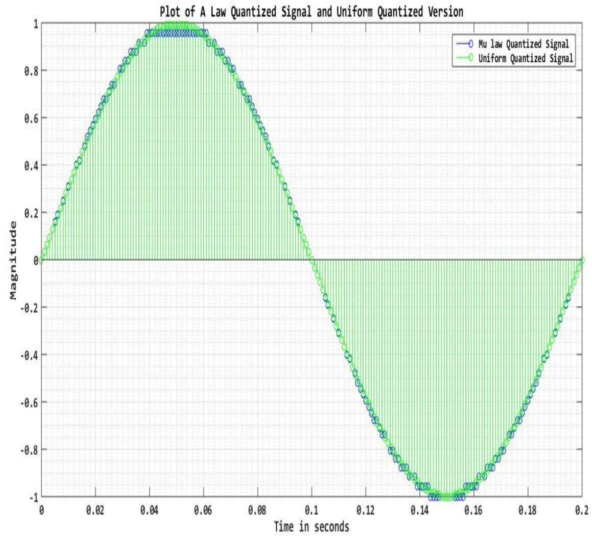
Uniform Quantization with μ Law Encoding and Decoding μ=127	1.1038e-04
Uniform Quantization with A Law Encoding and Decoding μ=255	1.5112e-04

Output Waveforms:

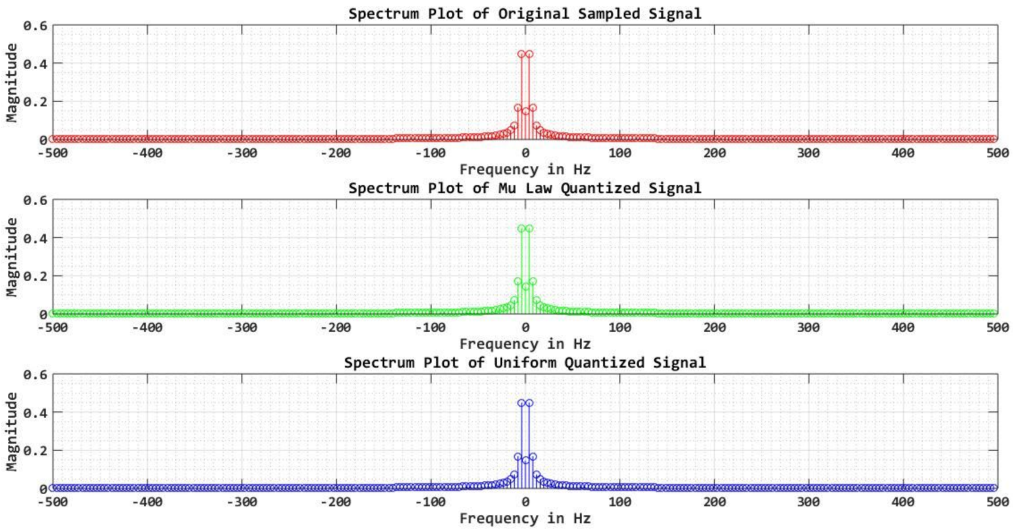


Input Sampled Signal

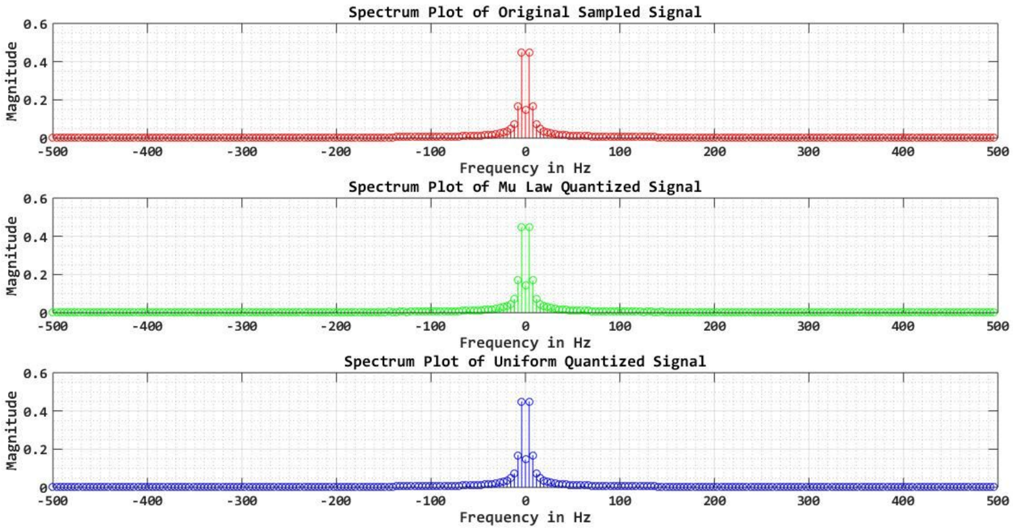

Comparison of Quantization With and Without µ Law Companding with µ=15

Comparison of Quantization With and Without µ Law Companding with µ=31



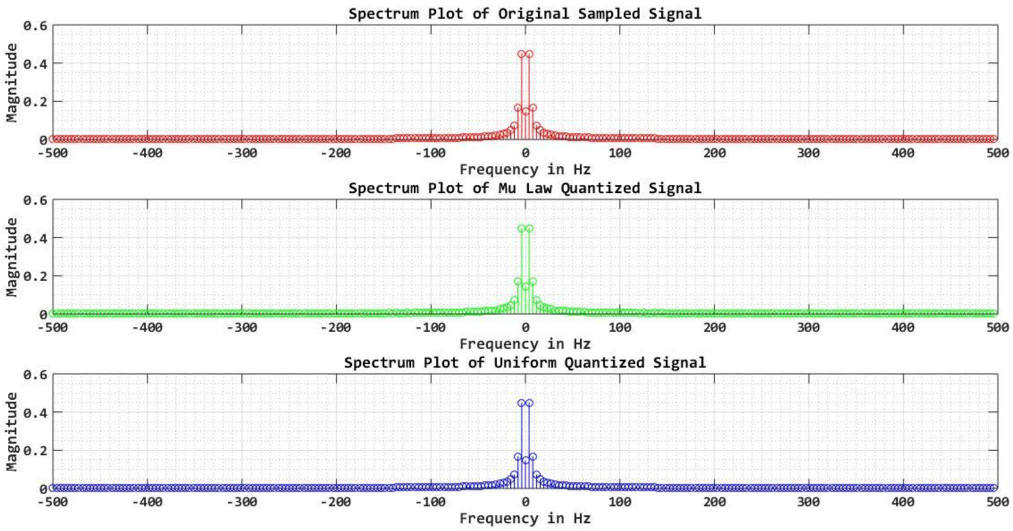
Comparison of Quantization With and Without μ Law Companding with μ=255



Spectral Comparison of Sampled Signal, Quantization With and Without μ Law Companding with μ=15



Spectral Comparison of Sampled Signal, Quantization With and Without µ Law Companding with µ=31



Spectral Comparison of Sampled Signal, Quantization With and Without µ Law Companding with µ=255

Discussions:

Similar as in the A-law companding case, we explain the fact that the MSE goes on increasing as the value of µ is increased which can be observed in the table above.

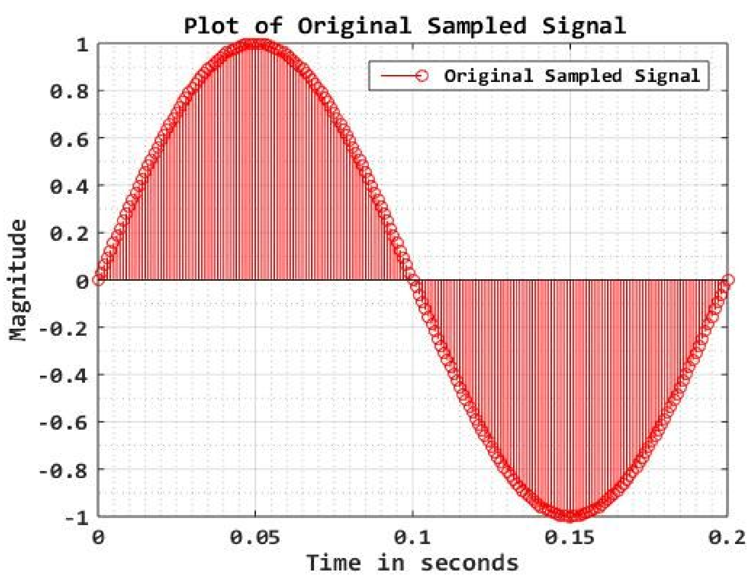
Part 3 : Dithering

A sinusoidal signal is taken and 12 bit quantization is performed. Then two processes are done: One signal is further truncated to 8 bits. Two, 5 bits of uniform noise is added, and then again truncated to 8 bits. This way, dithering is done and its effects are observed.

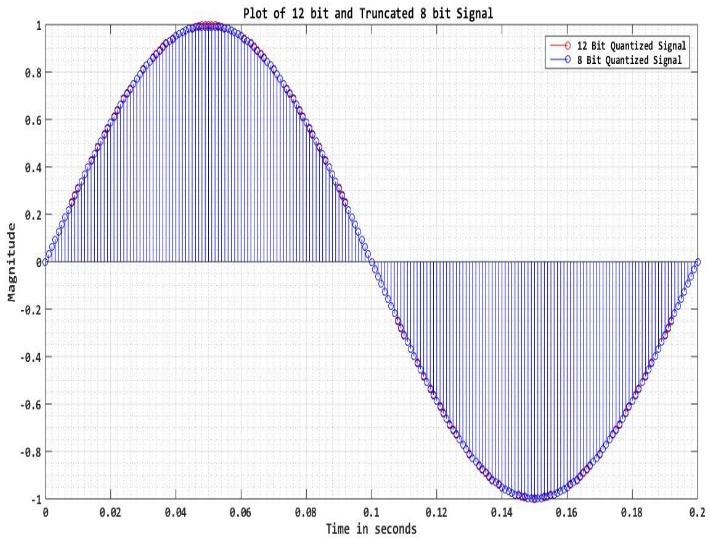
Error Comparison:

Specification	Error Amount
Sampled Signal and 12 Bit Quantized Signal	1.7199e-08
Sampled Signal and 8 Bit Truncated Quantized Signal	5.5590e-06
Sampled Signal and 12 Bit Quantized Signal with Noise added	2.0439e-05
Sampled Signal and 8 Bit Truncated Quantized Signal with Noise added	2.6286e-05

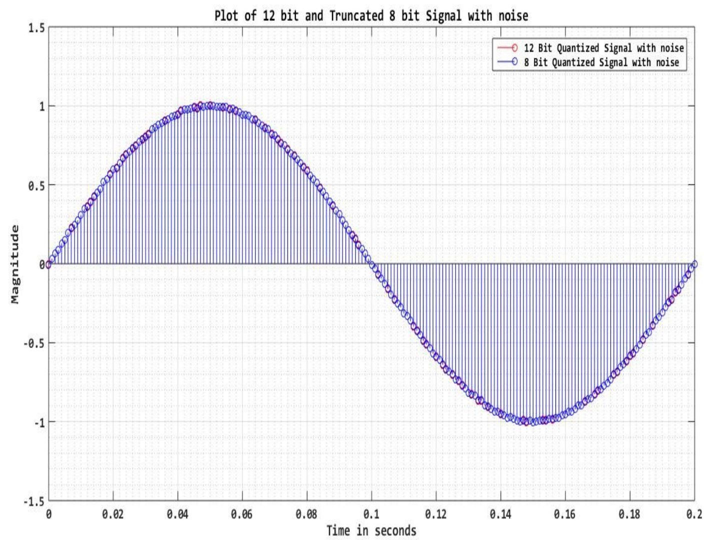
Output Waveforms:



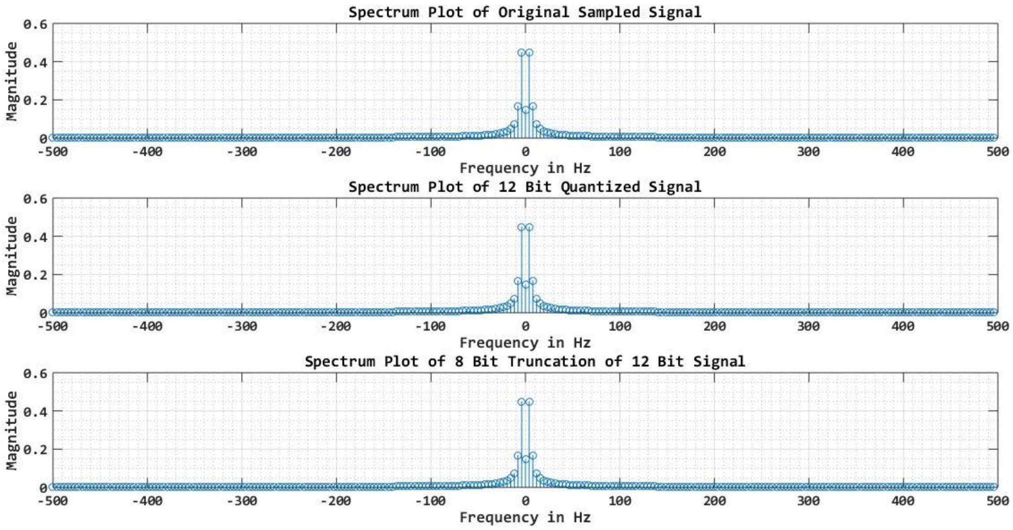
Input Sampled Signal



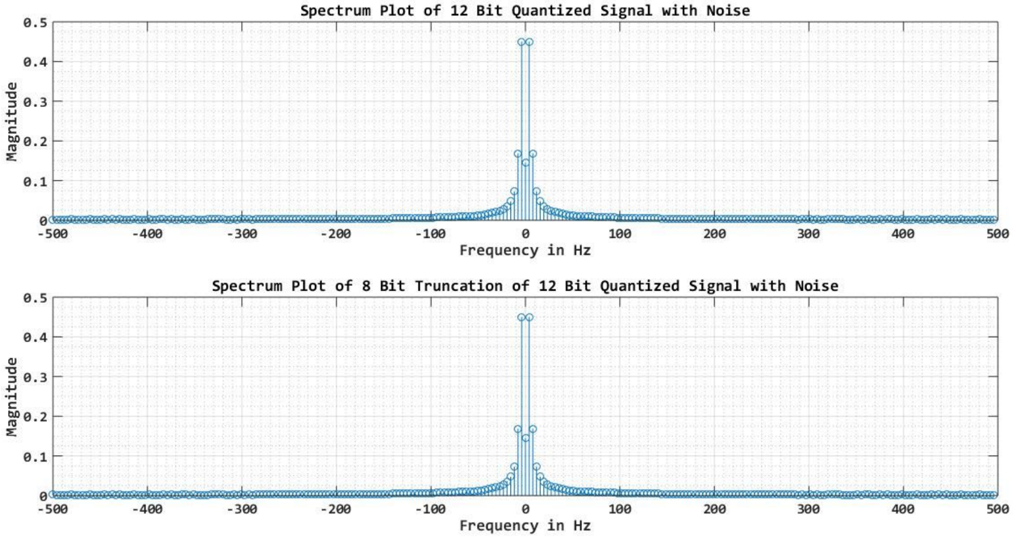
12 Bit Quantized Signal and 8 Bit Truncated Signal



12 Bit Noise added Quantized Signal and 8 Bit Truncated Signal



Spectral Comparison of Sampled Signal, 12 Bit Quantized Signal and 8 Bit Truncated Signal



Spectral Comparison of 12 Bit Noise Added Quantized Signal and 8 Bit Truncated Signal

Discussions:

As expected the error increased as we truncated the 12 Bit Quantization to 8 Bit Quantization values due to loss of information as error per term would increase. We also observed that the addition of white gaussian noise increased the root mean square quantization error. The spectrum was also plotted for each of the two cases. Very little distortion of the signal spectra was observed.

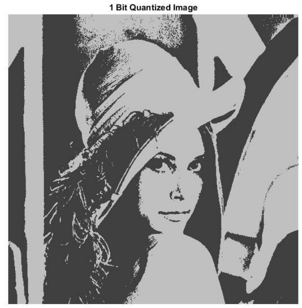
Part 4 : 1 Bit and 2 Bit Uniform Quantization of Grayscale image

In this part, we performed the quantization of an 8-bit grayscale image by designing a 1-bit and 2-bit uniform quantizer. For uniform quantization of pixel intensities, thresholding was performed and pixel values lying in a particular range was mapped to a certain pixel quantization value. In the 1-bit uniform quantisation case, the threshold was set at 127 and all the pixels below 127 were mapped to value 64 while the rest were mapped to 192. Thus, the entire image after quantisation was represented only by 2 pixel values. For 2 Bit quantization, the quantization was done to 4 levels.

Error Comparison:

Specification	Normalized Root Mean Squared Error
1 Bit Uniform Quantization	5.4297
2 Bit Uniform Quantization	1.4435

Output:



Original Image	1 Bit Quantized image	2 Bit Quantized image

Discussions:

As evident from the error table below, error increased as we increased the number of quantization levels from 2 to 4.

Part 5 : 1 Bit and 2 Bit NonUniform Quantization of Grayscale image - Max Lloyd Quantization

Max Lloyd's algorithm, also known as Voronoi iteration or relaxation, is an algorithm named after Stuart P. Lloyd for finding evenly spaced sets of points in subsets of Euclidean spaces and partitions of these subsets into well-shaped and uniformly sized convex cells. Like the closely related k-means clustering algorithm, it repeatedly finds the centroid of each set in the partition and then re-partitions the input according to which of these centroids is closest. However, Lloyd's algorithm differs from k-means clustering in that its input is a continuous geometric region rather than a discrete set of points. Thus, when re-partitioning the input, Lloyd's algorithm uses Voronoi diagrams rather than simply determining the nearest center to each of a finite set of points as the k-means algorithm does.

Lloyd's algorithm starts by an initial placement of some number *k* of point sites in the input domain. In mesh-smoothing applications, these would be the vertices of the mesh to be smoothed; in other applications they may be placed at random or by intersecting a uniform triangular mesh of the appropriate size with the input domain.

It then repeatedly executes the following relaxation step:

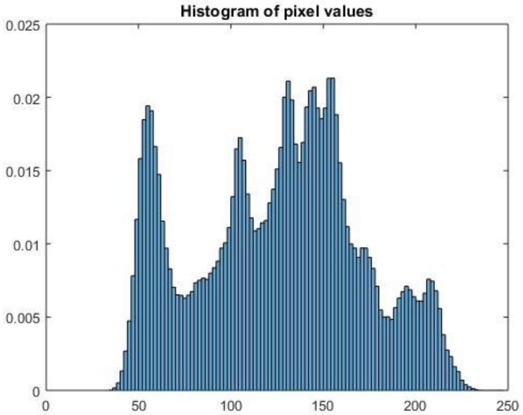
- The Voronoi diagram of the *k* sites is computed.
- Each cell of the Voronoi diagram is integrated, and the centroid is computed.
- Each site is then moved to the centroid of its Voronoi cell.

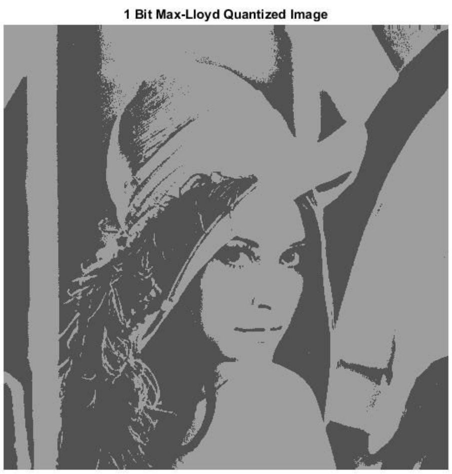
Because Voronoi diagram construction algorithms can be highly non-trivial, especially for inputs of dimension higher than two, the steps of calculating this diagram and finding the centroids of its cells may be approximated by a suitable discretization in which, for each cell of a fine grid, the closest site is determined, after which the centroid for a site's cell is approximated by averaging the centers of the grid cells assigned to it.

Error Comparison:

Specification	Normalized Root Mean Squared Error
1 Bit Non Uniform Quantization	2.4628
2 Bit Non Uniform Quantization	0.5675

Output:

	
Original Image	Histogram showing density of Pixel values

	
1 Bit Non Uniform Quantized image	2 Bit Non Uniform Quantized image

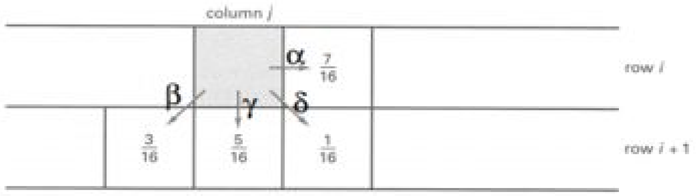
Discussions:

The error was found to be increase as the number of quantization levels were increased in case of Non Uniform Quantization, which is opposite to the trend observed in uniform quantization case.

Part 6 : Image Dithering

In this part, we performed error diffusion dithering of the 1 Bit uniformly quantised image. This was done to reduce effects of quantization by offering a better contrast and perceived resolution in the image. The distribution of errors among surrounding pixels exploits a concept known as spatial integration in our eye which displays a greater range of perceptible intensities which gives an illusion of improved image clarity.

We used Floyd Steinberg Algorithm was used to diffuse the errors among the pixels surrounding a particular pixel.



$$\alpha + \beta + \gamma + \delta = 1.0$$

The graphic above shows the diffusion of errors to the surrounding pixels. The four parameters can be chosen randomly. The only constraint imposed is that they should add up to 1.

Pseudo Code For Floyd Steinberg Algorithm :

```
for each y from top to bottom
   for each x from left to right
      oldpixel  := pixel[x][y]
      newpixel  := find_closest_palette_color(oldpixel)
      pixel[x][y]  := newpixel
      quant_error  := oldpixel - newpixel
      pixel[x + 1][y    ] := pixel[x + 1][y    ] + quant_error * 7 / 16
      pixel[x - 1][y + 1] := pixel[x - 1][y + 1] + quant_error * 3 / 16
      pixel[x    ][y + 1] := pixel[x    ][y + 1] + quant_error * 5 / 16
      pixel[x + 1][y + 1] := pixel[x + 1][y + 1] + quant_error * 1 / 16
```

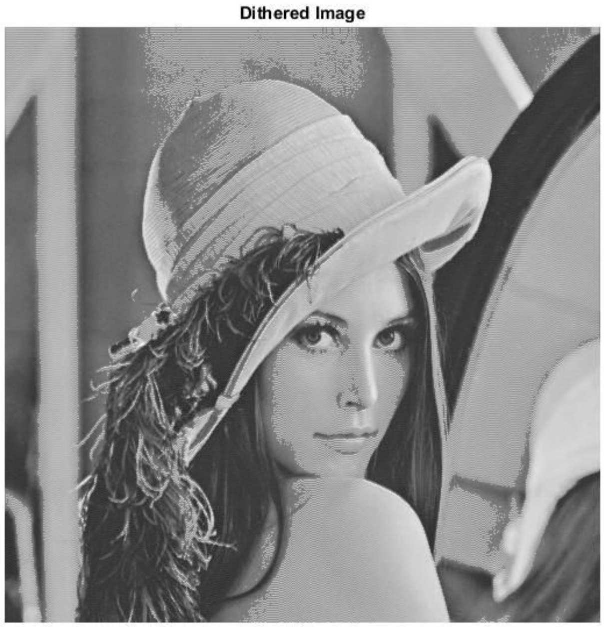
Error Comparison:

Specification	Normalized Root Mean Squared Error
1 Bit Uniform Quantization	5.3907
1 Bit Uniform Quantization with error diffusion to right pixel only	1.3945
1 Bit Uniform Quantization with error diffusion to the pixel below	1.3112
1 Bit Uniform Quantization with error diffusion to all four surrounding pixels	0.8133

Output:

	
Original Image	1 Bit Quantized Image

Error Diffusion to pixel on right



Error Diffusion to pixel below

Error Diffusion to nearby 4 pixels

Discussions:

We performed the error-diffusion dithering in the 3 ways. One error was diffused only to the right adjacent pixel, second error was propagated only to the bottom adjacent pixel, third error was propagated to all four adjacent pixels with certain weights summing up to unity.

In first case we saw that the image contrast got considerably improved in comparison to the 1 Bit quantized image. Some vertical lines were observed. In the second part, we also observed the improvement of the image contrast, but with addition of some horizontal lines. In the third part we diffused the error to all the neighbouring pixels, and obtained the best image clarity. Overall, diffusing error helped in reducing the error in the quantized image as compared to original image.