CIE 337 Project Two Report (Part I)

Quantizater Program Using MATLAB® App Designer

Written by Rawan Eldalil

Communication and Information Engineering Department

University of Science and Technology at Zewail City

May 30, 2021

Overview

Quantization is the process of approximating a signal with an infinite set of amplitudes to another signal with discrete amplitudes. It can be either *uniform* or *non-uniform*. In uniform quantization, quantization levels are equally spaced by a step size $\delta = \frac{2m_p}{L}$, where m_p is the peak quantization lever and L is number of quantization levels. On the other hand, non-uniform quantization can be applied to a given signal by compressing the signal in accordance with μ-law or A-law practices, yet only μ-law is emphasized, then apply regular uniform quantization. In this part of the project, it is required to build a GUI-based program using MATLAB® App Designer tool, where the user is allowed to enter a time vector and the baseband signal, choose the quantization technique (i.e. uniform or non-uniform) and specify the needed parameters for each technique. In return, a plot of the quantized signal and the original signal is drawn and the mean square quantization error is displayed.

The Program Layout Design on App Designer

The initial program layout design shown in **Figure 1** has the following components:

- An entry field for the time vector, that is used to get the sampling time T_s, with a guiding label next to it.
- An entry field for the to-be-quantized signal with a guiding label next to it.
- A drop-down component that allows the user to choose between uniform and non-uniform quantized.
- A push button when clicked, it displays the the square mean quantization error and plot of the quantized signal and the original signal.
- An axes component for the plots to be drawn on.
- A label component to display the mean square quantization error

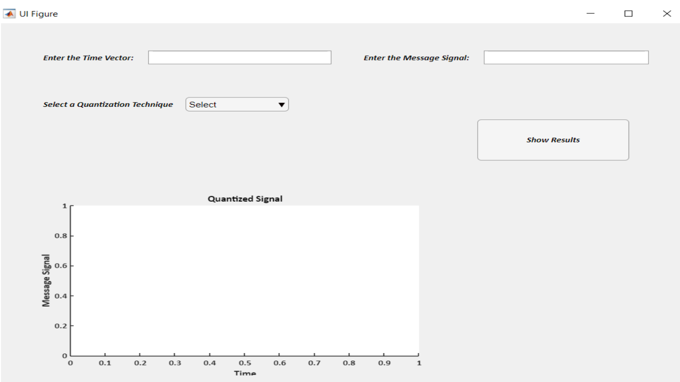

Figure 1: The initial Program Layout

How the Program works

When running the app, the window shown in **Figure 1** pops up, then the user is asked to enter both the time vector and the signal to be quantized, where it is a **must** to leave a space between each element in the time and message vectors. Then, the user should select the quantization technique, where two options are provided. For the uniform quantization selection, an entry field shows up to ask the user to enter L, m_p and whether the quantization is mid-rise or mid-tread, **all in order**; a space between each entered parameter **must** exist. On the other hand, when the user selects non-uniform quantization, an entry field shows up to ask the user to enter μ, L, m_p and whether the quantization is mid-rise or mid-tread, **all in order**; a space between each entered parameter **must** exist. Also, one **must** enter **R**, if the desired quantization is mid-rise and **T** for mid-tread quantization. To obtain the results, the user should click on the **Show Results** button.

The Results of the Test Input Signals

In this section, it is required to test the program with four test cases for a cosine signal with $f = 10\ Hz$ and amplitude of 5, which are generated by a MATLAB® code submitted with this report and have a different set of parameters.

The Input Signal Representation for Plotting

The original signal representation that is plotted in the following subsections is an approximation of the input signal, where it is ***interpolated*** with the interpolation method is 'spline'.

Case I

The signal in this case is sampled at a frequency of $40\ Hz$; the time it takes for one cycle to occur is $\frac{1}{10}\ s$. Accordingly, the time vector is $[0\ 0.025\ 0.05\ 0.075\ 0.1]$, and the signal to be quantized is $[5\ 0\ -5\ 0\ 5]$. With $L = 8$, $m_p = 5$ and $\mu = 0$, and the quantization type is specified to be mid-rise. The quantized signal is shown in **Figure 2**.

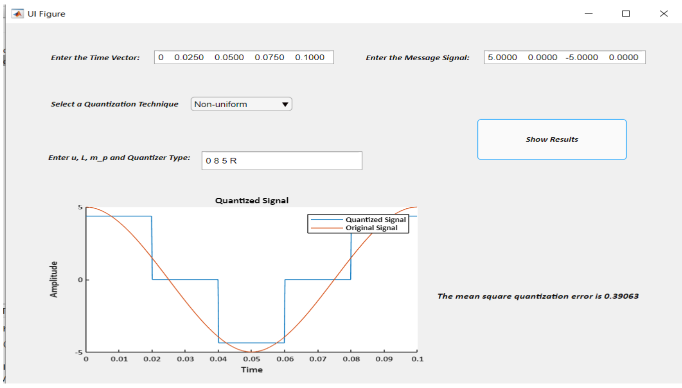

Figure 2: The First Test Case Results

Case II

The signal in this case is sampled at a frequency of $20\ Hz$; the time it takes for one cycle to occur is $\frac{1}{10}\ s$. Accordingly, the time vector is $[0\ 0.05\ 0.1]$, and the signal to be quantized is $[5\ -5\ 5]$. With $L = 32$, $m_p = 5$ and $\mu = 0$, and the quantization type is specified to be mid-rise. The quantized signal is shown in **Figure 3**.

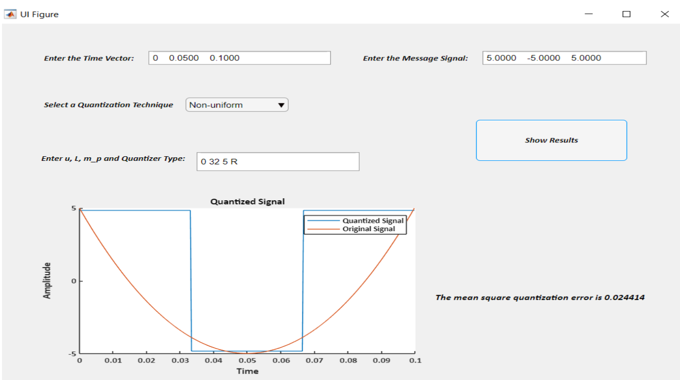

Figure 3: The Second Test Case Results

Case III

The signal in this case is sampled at a frequency of $15\ Hz$; the time it takes for one cycle to occur is $\frac{1}{10}\ s$. Accordingly, the time vector is $[0\ 0.667]$, and the signal to be quantized is $[5\ \ -2.5]$. With $L = 16$, $m_p = 5$ and $\mu = 0$, and the quantization type is specified to be mid-rise. The quantized signal is shown in **Figure 4**.

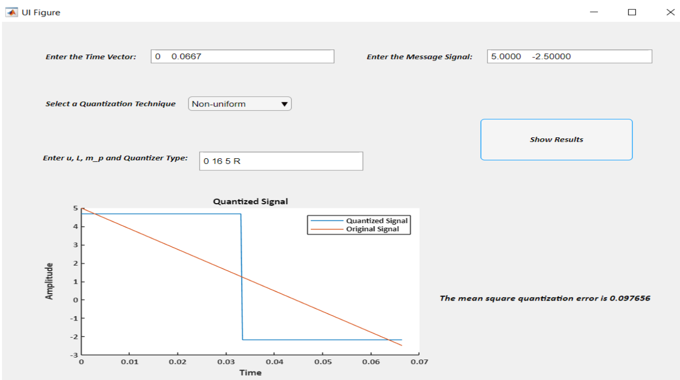

Figure 4: The Third Test Case Results

Case IV

The signal in this case is sampled at a frequency of $20\ Hz$; the time it takes for one cycle to occur is $\frac{1}{10}\ s$. Accordingly, the time vector is $[0\ 0.05\ 0.1]$, and the signal to be quantized is $[5\ \ -5\ 5]$. With $L = 32$, $m_p = 5$ and $\mu = 100$, and the quantization type is specified to be mid-rise. The quantized signal is shown in **Figure 5**.

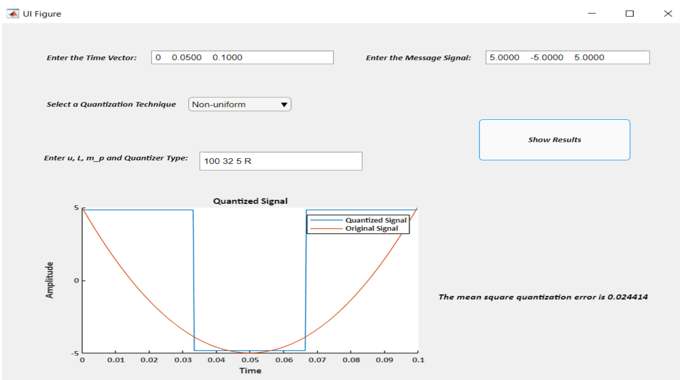

Figure 5: The Forth Test Case Results

Comments on the Results

Looking at the results, one observes the following:

- The representation of the input signal becomes less accurate when the sampling frequency is less than or equal to double the frequency of the signal f_m.
- When the comparison between Case I signal and the rest of the cases is held with respect to the ***increase*** in the number of levels, one finds that the mean square quantization error ***decrease*** with m_p constant, accordingly.
- For Case I and Case IV signals, the change in the value of μ doesn't affect the result, since the output value of the μ-law formula is 1 in both cases.