

Experiment 7

Sampling & Quantization

MATLAB Simulation

Objectives

The main objectives of this experiment are:

- 1) To gain a clearer understanding of sampling theory and uniform quantization.
- 2) To learn how to simulate these operations using MATLAB.

Pre-Lab Work

- Use the MATLAB function **pulstran** to generate and plot three periodic pulse trains with different shapes each with three different pulse durations. Read the help of that function and try to take advantage of the example given is at the end of the help for **pulstran**. However, do not copy it!
- Given a vector $x=[1.1,2.6,-3.5,-0.001,0.768]$, apply the MATLAB functions **fix**, **round**, **floor**, & **ceil** on x . State the differences between the resulting vectors.

Overview

An analog signal is characterized by the fact that its attributes (like: amplitude, frequency and phase) can take any value over a continuous range. On the other hand, digital signals can take only discrete and finite values. One can convert an analog signal to a digital signal by sampling and quantizing (collectively called analog-to-digital conversion, or ADC).

It is typically more efficient to process the resulting discrete signals by digital signal processors,. The processed signals are then converted back into analog signals using a reconstruction or interpolation operation (called digital-to-analog conversion, or DAC).

Sampling:

To sample a continuous-time signal $x(t)$ is to represent $x(t)$ at a discrete number of points, $t = nT_s$, where T_s is the sampling period. The sampling theorem states that a band-limited signal $x(t)$ with a bandwidth W can be reconstructed from its sample values $x(n) = x(nT_s)$ if

the sampling frequency $f_s = 1/T_s$ is greater than twice the bandwidth W of $x(t)$. Otherwise, aliasing would result in $x(t)$. The minimum sampling rate of $2f_s$ for an analog band-limited signal is called the Nyquist rate.

Quantization :

In order to process the sampled signal digitally, the sample values have to be quantized to a finite number of levels, and each value can then be represented by a string of bits. For example, if the signal is quantized to N different levels, then $\log_2(N)$ bits per sample are required.

Notice that to quantize a sample value is to round it to the nearest point among a finite set of permissible values. Therefore, a distortion will inevitably occur. This is called quantization noise (or error).

Quantization can be classified as uniform and non-uniform. In the case of uniform quantization, the quantization regions are chosen to have equal length. However, in non-uniform quantization, regions of various lengths are allowed. Non-uniform quantization can be implemented through compression-expansion (or companding) of the signal, and this is commonly used (as in telephony) to maintain a uniform signal-to-quantization noise ratio over the full dynamic range of the signal (refer to your textbook for more details).

Lab Work

- 1) Generate a time vector \mathbf{t} from -0.5 to 0.5 with a step size of 0.001. Implement the function $x(t) = e^{-|t|/\tau}$ where $\tau = 0.1$. In order to simulate an A/D converter, perform the following tasks:
 - a. Plot \mathbf{x} and its magnitude spectrum in a two-panel figure window. What is the bandwidth of \mathbf{x} ?
 - b. Set the sampling frequency to twice the bandwidth of the signal \mathbf{x} (which is approximately 25Hz). Generate a rectangular pulse train starting at -2 to 2 where the step size is $1/(\text{sampling frequency})$ and with a duration $1\text{E-}10$. In a two-panel figure window, plot the pulse train and its magnitude spectrum. What is the relation between the time and frequency domain representation of the pulse train?
 - c. Sample \mathbf{x} using the pulse train and plot the resulting sampled version of \mathbf{x} , (say \mathbf{xs}). Also, plot its magnitude spectrum. What can you observe from both plots?

Now, to be able to simulate a D/A converter, perform the following tasks:

- d. Let \mathbf{h} be a vector of forty ones (40). Plot its magnitude spectrum. In order to correctly plot $H(\mathbf{f})$, use the `fft` function with a number of FFT points equal to the length of \mathbf{x} .

e. Use the function **filter** to form the reconstructed signal (say **xd**) from **xs** using the filter **h**, i.e. **xd=filter(h,1,xs)**. In a 4-panel figure window, show the plots of **x**, **xd**, and their magnitude spectrums. Comment on your findings.

Useful MATLAB functions: pulstran

- 2) Repeat 1 using a sampling frequency equal to 1.5 times the bandwidth of $x(t)$.
- 3) Repeat 1 using a sampling frequency equal to 10 times the bandwidth of $x(t)$.
- 4) Generate a sine signal (call it **x**) with frequency equal to $1/2\pi$ over a time interval of three periods. The step size should be equal to 1E-4. The quantized signal will be equal to $x_q = \Delta x \text{round}(x/\Delta)$, where Δ is the quantization step size. You can compute it using the formula $\Delta = 2 \max(|x|)/L$ where $L = 2^b$ and b is the number of quantization bits. Use 2 bits to quantize **x** and show the following:
 - a. In one figure plot **x** vs. time and the quantized signal vs. time.
 - b. Compute the Signal-to-Quantization Noise Ratio (SQNR) defined by the formula: $SQNR = 10 \log_{10}(\sum x^2 / \sum (x - x_q)^2)$ in dB.
- 5) Repeat Part 4 using 3, 4 and 5 bits per sample.
- 6) Plot the SQNR vs. the number of bits per sample for 2, 3, 4, and 5 bits. What is your conclusion?

Additional Questions:

- Q1.** Can you process real-world analog signal by digital computers? List possible applications where sampling and quantization are necessary.
- Q2.** From the above results, what can you conclude about the improvement in SQNR with every additional quantization bit?