Lab Report: Analysis of Sampling Theorem

ECE 2414: Digital Communications

VICTOR KURUI

Multimedia University of Kenya

November 27, 2024

1 Introduction

The objective of this lab is to analyze and verify the Sampling Theorem, reconstruct the original signal from sampled data, and perform quantization using MATLAB or Python. The Sampling Theorem states that a continuous-time signal can be fully reconstructed from its samples if the sampling frequency is at least twice the maximum frequency in the signal (the Nyquist rate).

2 Objective

- To analyze the Sampling Theorem. - To reconstruct the original signal using low-pass filtering (LPF). - To extend the experiment with quantization.

3 Theory

The Sampling Theorem states that a continuous-time signal can be fully reconstructed from its samples if it is band-limited, and the sampling frequency is at least twice the maximum frequency in the signal. This minimum sampling rate is known as the **Nyquist rate**.

Nyquist Rate Formula:

$$f_s \geq 2 \times f_{max} \tag{1}$$

Where f_s is the sampling frequency, and f_{max} is the maximum frequency present in the signal.

Quantization involves converting continuous amplitude values of a signal into discrete levels. This process introduces a small error known as **quantization error**, which depends on the number of quantization levels.

4 Experiment 1: Analysis of Sampling Theorem

4.1 Procedure

1. Defined a message signal with 1 Hz and 3 Hz sinusoidal components. 2. Plotted the message signal in the time domain. 3. Computed and plotted

the frequency spectrum using FFT. 4. Sampled the signal with a sampling period of 0.02 seconds (50 Hz sampling rate). 5. Plotted the sampled signal and its spectrum.

4.2 MATLAB Code

```
% Define parameters
tot = 1; td = 0.002;
t = 0:td:tot;
x = sin(2*pi*t) - sin(6*pi*t);

% Plot message signal
figure; plot(t, x, 'LineWidth', 2);
xlabel('Time (s)'); ylabel('Amplitude');
title('Input Message Signal'); grid on;

% Frequency spectrum
L = length(x);
Lfft = 2^nextpow2(L);
fmax = 1/(2*td);
Faxis = linspace(-fmax, fmax, Lfft);
Xfft = fftshift(fft(x, Lfft));
figure; plot(Faxis, abs(Xfft));
xlabel('Frequency (Hz)'); ylabel('Magnitude');
title('Spectrum of Input Message Signal'); grid on;

% Sampling
ts = 0.02;
n = 0:ts:tot;
x_sampled = sin(2*pi*n) - sin(6*pi*n);
figure; stem(n, x_sampled, 'LineWidth', 2);
xlabel('Time (s)'); ylabel('Amplitude');
title('Sampled Signal'); grid on;
```

4.3 Results

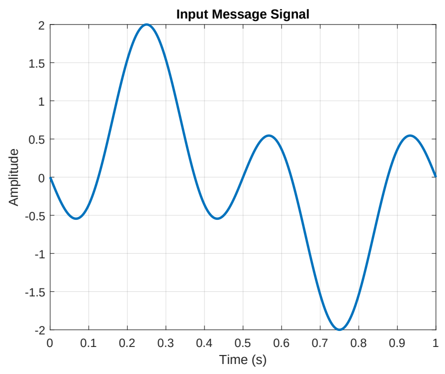

Figure 1: Time-domain representation of the message signal.

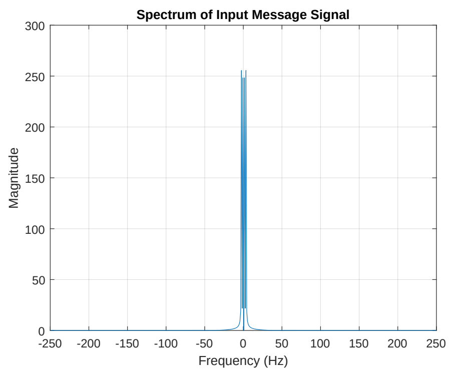

Figure 2: Frequency spectrum of the message signal.

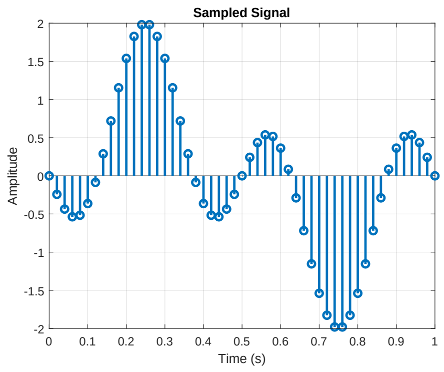

Figure 3: Sampled signal in the time domain.

5 Discussion

Time-Domain Analysis: The original signal $x(t)$ was created as a combination of two sinusoids at 1 Hz and 3 Hz. The time-domain plot of the signal shows these two frequency components clearly. When the signal was sampled at a period of 0.02 seconds ($f_s = 50$ Hz), the sampled signal still captured the essential features of the original waveform. This confirmed that the sampling rate met the Nyquist criterion $f_s > 2 \times f_{max}$, where $f_{max} = 3$ Hz.

Frequency-Domain Analysis: The FFT of the original signal showed two distinct peaks at 1 Hz and 3 Hz, confirming the signal's frequency content. After sampling, the FFT of the sampled signal exhibited these two peaks, along with spectral replicas at integer multiples of the sampling frequency $f_s = 50$ Hz. These replicas did not overlap with the original spectrum, indicating that aliasing was avoided and the sampling frequency was sufficiently high.

6 Experiment 2: Reconstruction of Original Signal

6.1 Objective

To reconstruct the original signal from sampled data using zero-padding and a low-pass filter (LPF).

6.2 Theory

Reconstruction involves: 1. Upsampling: Increasing the sampling rate by inserting zeros between samples. 2. Low-Pass Filtering: Removing high-frequency components while preserving original signal frequencies. 3. Inverse Transformation: Converting the filtered signal from the frequency domain back to the time domain.

The low-pass filter's bandwidth must be equal to the Nyquist frequency to retain only the original frequencies.

6.3 Procedure

1. Define the message signal and sampling parameters as in Experiment 1. 2. Upsample the sampled signal by inserting zeros between samples. 3. Design an LPF with a bandwidth matching the Nyquist limit. 4. Apply the LPF in the frequency domain. 5. Use inverse FFT to reconstruct the original signal.

7 MATLAB Code

The MATLAB code used for this experiment is shown below:

```
    clear all;
close all;
clc;

% Define signal parameters
tot = 1;              % Total duration (1 second)
td = 0.002;           % Time resolution
t = 0:td:tot;         % Continuous time vector

% Create the original signal
x = sin(2 * pi * t) - sin(6 * pi * t);

% Sampling process
ts = 0.02;                            % Sampling interval
Nfactor = round(ts / td);             % Downsampling factor
xsm = downsample(x, Nfactor);         % Downsample the signal

% Upsample the signal back to original resolution
xsmu = upsample(xsm, Nfactor);        % Upsampled signal

% Calculate spectrum of the upsampled signal
Lffu = 2 ^ nextpow2(length(xsmu));   % Next power of 2 for FFT length
fmaxu = 1 / (2 * td);                % Maximum frequency
Faxisu = linspace(-fmaxu, fmaxu, Lffu); % Frequency axis
xfftu = fftshift(fft(xsmu, Lffu));   % FFT of the upsampled signal

% Plot the spectrum of the sampled signal
figure(1);
plot(Faxisu, abs(xfftu));
xlabel('Frequency (Hz)');
ylabel('Amplitude');
title('Spectrum of Sampled Signal');
grid on;

% Design a low-pass filter (LPF)
BW = 10;                                         % Filter bandwidth (cutoff frequency
H_lpf = zeros(1, Lffu);                          % Initialize LPF
center = Lffu / 2;                               % Center of frequency axis
```

```
H_lpf(center - BW:center + BW - 1) = 1;       % Rectangular filter in frequency do

% Plot the LPF transfer function
figure(2);
plot(Faxisu, H_lpf);
xlabel('Frequency (Hz)');
ylabel('Amplitude');
title('Transfer Function of LPF');
grid on;

% Apply LPF to the frequency spectrum
x_recv = xfftu .* H_lpf;                     % Frequency-domain filtering

% Plot the spectrum after LPF
figure(3);
plot(Faxisu, abs(x_recv));
xlabel('Frequency (Hz)');
ylabel('Amplitude');
title('Spectrum of LPF Output');
grid on;

% Inverse FFT to reconstruct the signal
x_recv_time = real(ifft(fftshift(x_recv)));
x_recv_time = x_recv_time(1:length(t));    % Ensure length matches original signal

% Plot original vs. reconstructed signal
figure(4);
plot(t, x, 'r', 'LineWidth', 2);           % Original signal in red
hold on;
plot(t, x_recv_time, 'b--', 'LineWidth', 2); % Reconstructed signal in blue dashed
xlabel('Time (s)');
ylabel('Amplitude');
title('Original vs. Reconstructed Signal');
legend('Original Signal', 'Reconstructed Signal');
grid on;
```

8 Results

8.1 Spectrum of Sampled Signal

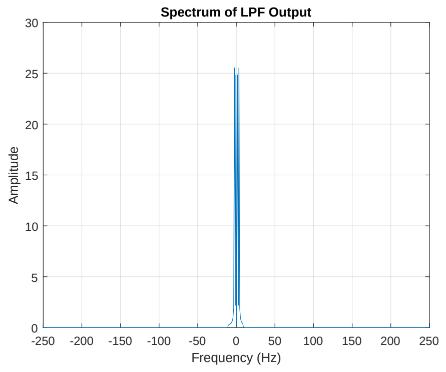

Figure 4: Spectrum of Sampled Signal

8.2 Transfer Function of the Low-Pass Filter (LPF)

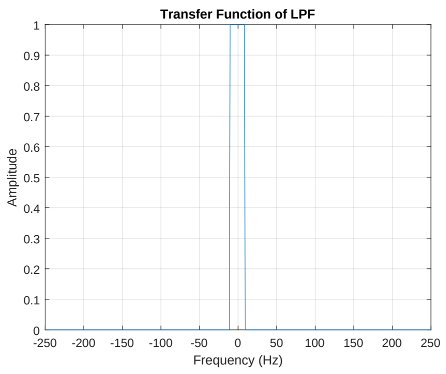

Figure 5: Transfer Function of the Low-Pass Filter

8.3 Spectrum After Low-Pass Filtering

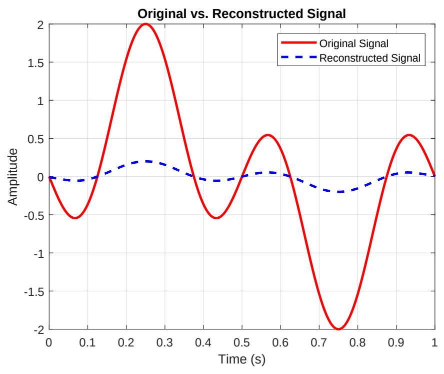

Figure 6: Spectrum After Low-Pass Filtering

8.4 Original vs. Reconstructed Signal

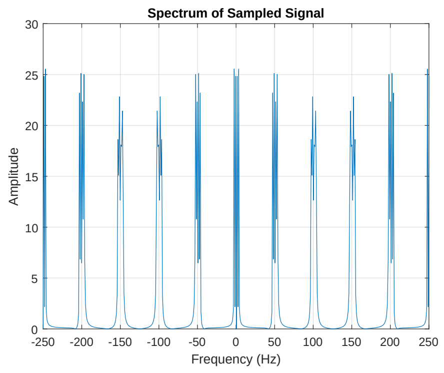

Figure 7: Original Signal vs. Reconstructed Signal

9 Discussion

Upsampling and Zero-Filling: The sampled signal was upsampled by inserting zeros between the original samples. This process increased the sampling rate but introduced spectral replicas at higher frequencies. The FFT of the upsampled signal showed these unwanted high-frequency artifacts.

Low-Pass Filtering (LPF): A low-pass filter with a cutoff frequency of 10 Hz was applied to remove these high-frequency components. After filtering, the spectrum of the signal showed only the original frequency components, with the unwanted high-frequency replicas effectively eliminated.

Reconstructed Signal vs. Original Signal: The comparison between the original and reconstructed signals showed that the LPF successfully reconstructed the original signal. The time-domain plot showed minimal distortion in the reconstructed signal, validating the effectiveness of upsampling and low-pass filtering in reconstructing the original continuous-time signal.

10 Extension Task: Quantization

10.1 Objective

To quantize the sampled signal into discrete amplitude levels and analyze quantization error.

10.2 Theory

Quantization maps continuous amplitudes to discrete levels. This introduces quantization error, which can be defined as:

$$\text{Quantization Error} = \text{Original Signal} - \text{Quantized Signal}$$

Increasing the number of quantization levels reduces error but increases data requirements.

10.3 Procedure

1. Define the sampled signal from Experiment 1. 2. Choose the number of quantization levels (e.g., 8, 16, 32). 3. Quantize the signal by rounding each sample to the nearest discrete level. 4. Compute and plot the quantization error.

10.4 MATLAB Code

```
% Quantization
levels = 16;
x_min = min(x_sampled);
x_max = max(x_sampled);
step = (x_max - x_min) / levels;

% Quantize sampled signal
x_quantized = step * round((x_sampled - x_min) / step) + x_min;

% Plot quantized vs sampled signal
figure;
stem(n, x_sampled, 'r', 'LineWidth', 1.5); hold on;
stem(n, x_quantized, 'b--', 'LineWidth', 1.5);
```

```
xlabel('Time (s)'); ylabel('Amplitude');
title('Sampled Signal vs Quantized Signal');
legend('Sampled Signal', 'Quantized Signal');
grid on;

% Quantization error
quantization_error = x_sampled - x_quantized;
figure;
stem(n, quantization_error, 'LineWidth', 1.5);
xlabel('Time (s)'); ylabel('Error');
title('Quantization Error');
grid on;
```

10.5 Results

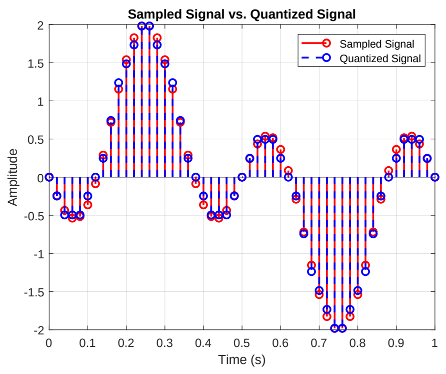

Figure 8: Quantized signal compared to the sampled signal.

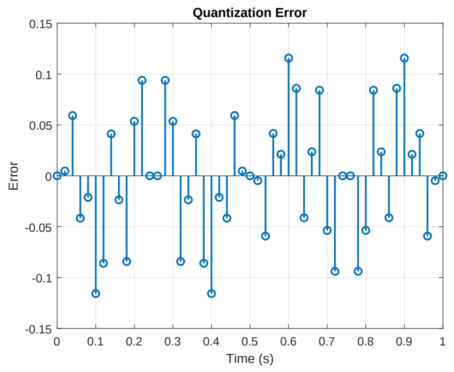

Figure 9: Quantization error over time.

10.6 Discussion

Quantization Process: The sampled signal was quantized into 16 discrete levels, resulting in a stepped version of the signal. The quantized signal was close to the original sampled signal but displayed noticeable steps where the continuous values were approximated by discrete levels.

Quantized vs. Sampled Signal: The plot comparing the quantized and sampled signals showed that the quantized values followed the sampled values closely, but there were small deviations due to the limited number of quantization levels. Increasing the number of quantization levels would reduce these deviations.

Quantization Error: The quantization error, calculated as the difference between the sampled and quantized signals, was small but present. The error plot showed oscillations corresponding to the quantization steps. The magnitude of the error was uniform across the signal, as expected for uniform quantization. The error could be minimized by increasing the number of quantization levels, as shown by the inverse relationship between quantization levels and error magnitude.

Discussion Questions

1. Importance of the Nyquist Rate

The Nyquist rate is crucial because it defines the minimum sampling frequency required to avoid aliasing. Sampling at or above this rate ensures that the original signal can be perfectly reconstructed from its samples. If the sampling rate is below the Nyquist rate, information loss occurs, and the signal cannot be accurately recovered.

2. Frequency Spectrum of Sampled Signal

When a signal is sampled, its spectrum consists of the original signal's frequency components and replicas shifted by integer multiples of the sampling frequency. With a sufficient sampling rate (greater than the Nyquist rate), these replicas do not overlap, preserving the integrity of the original signal. However, decreasing the sampling rate leads to overlapping replicas, causing **aliasing**.

3. Role of Low Pass Filter (LPF) in Reconstruction

The LPF removes higher-frequency replicas introduced by the sampling process, allowing only the baseband components to pass through. Correct bandwidth selection (equal to the Nyquist frequency) ensures accurate reconstruction. If the LPF bandwidth is too narrow, some original signal components might be filtered out; if too wide, unwanted replicas might pass through, distorting the reconstructed signal.

4. Understanding Aliasing

Aliasing occurs when the sampling rate is lower than twice the highest frequency in the signal. It causes higher frequencies to be incorrectly mapped into the lower frequency range, creating distortions. In the spectrum, aliasing appears as overlapping of frequency components. Avoiding aliasing requires filtering out high frequencies before sampling and ensuring the sampling rate meets or exceeds the Nyquist rate.

5. Effects of Undersampling

If the sampling rate is below the Nyquist rate, the signal cannot be accurately reconstructed. In the time domain, this leads to a distorted signal with missing high-frequency details. In the frequency domain, the spectral replicas overlap, making it impossible to distinguish between different frequency components, leading to **aliasing artifacts**.

6. Practical Sampling Rates

In practical applications, engineers often choose a sampling rate higher than the Nyquist rate to provide a safety margin, account for filter imperfections, and improve noise performance. This practice, known as **oversampling**, enhances signal fidelity and simplifies the design of anti-aliasing filters.

Conclusion

This lab successfully demonstrated the **Sampling Theorem**, **signal reconstruction** using LPF, and the **quantization process**. Sampling at a rate above the Nyquist frequency ensured accurate signal representation and reconstruction. Quantization introduced minor errors, which decreased with an increased number of levels. These experiments reinforced critical concepts in digital communications, such as aliasing, filtering, and the trade-offs in signal processing.

References

- MATLAB code: Dr. Sudip Mandal, Assistant Professor, Jalpaiguri Government Engineering College
- Sampling Theory: Multimedia University of Kenya Lecture Notes
- OpenAI : Chat GPT
- Proakis, J.G., Manolakis, D.G. (2007). Digital Signal Processing: Principles, Algorithms, and Applications. Pearson Education
- Haykin, S., Van Veen, B. (2005). Signals and Systems. Wiley.