Digital Communication Lab

Signal Sampling and Fourier Transform Analysis

Name: Sachin Kumar
Roll No: 220104026
Section: D
Lab No. : 2

October 19, 2024

1 Objective

The objective of this experiment is to analyze the spectrum of a sampled sinusoidal signal using an impulse train for a given continuous-time sinusoidal signal.

2 Theory

In the frequency domain, we analyze the spectral content of signals, which provides insights into the distribution of signal energy across various frequency components. The frequency representation is obtained through the application of the Fast Fourier Transform (FFT), which converts a time-domain signal into its corresponding frequency-domain representation.

For our continuous-time sinusoidal signal $m(t) = \cos(2\pi f_m t)$, the frequency spectrum can be derived as follows:

1. Message Signal Spectrum

The frequency spectrum of the message signal M_f is obtained by computing the FFT of the signal:

$$M_f = \frac{|FFT(m(t))|}{N}$$

where N is the number of points in the time-domain signal. The frequency vector is calculated as:

$$F = linspace\left(-\frac{1}{2T_s}, \frac{1}{2T_s}, N\right),$$

allowing us to relate the FFT output to actual frequency values.

2. Impulse Train Spectrum

The impulse train $g(t)$ exhibits a distinct frequency spectrum G_f, characterized by an infinite set of equally spaced impulses in the frequency domain. The FFT of the impulse train is given by:

$$G_f = |FFT(g(t))|.$$

The frequency response of the impulse train is fundamental in understanding its role in sampling and reconstructing signals.

3. Sampled Signal Spectrum

The sampled signal $M_d(t)$ also has a frequency spectrum M_{df} computed using the FFT:

$$M_{df} = |FFT(M_d(t))|.$$

The sampling process results in the replication of the original signal's spectrum, scaled by the sampling frequency, and introduces potential aliasing if the sampling frequency is not sufficient according to the Nyquist theorem.

The frequency domain representation is crucial for understanding signal characteristics, such as bandwidth and the presence of harmonics. It facilitates the analysis of systems and filters, providing insights into how signals can be manipulated for various applications in communications, audio processing, and signal processing.

3 Python Code

```
import numpy as np
import matplotlib.pyplot as plt

# Parameters
f_impulse = 100
f_message = 5
t_end = 1
interval = 0.01

# Time vector
t = np.arange(0, t_end, interval)

# Impulse train
impulse_train = f_impulse * (np.mod(t, 1/f_impulse) < interval)

# Message signal
message_signal = np.cos(2 * np.pi * f_message * t)

# Sampled signal
sampled_signal = message_signal * impulse_train

# FFT parameters
N = len(t)
f_axis = np.linspace(-1/(2*interval), 1/(2*interval), N)

# FFT computations
fft_message = np.abs(np.fft.fftshift(np.fft.fft(message_signal) / N))
fft_impulse = np.abs(np.fft.fftshift(np.fft.fft(impulse_train) / N))
fft_sampled = np.abs(np.fft.fftshift(np.fft.fft(sampled_signal) / N))

# Plotting
plt.figure(figsize=(10, 8))

plt.subplot(3, 1, 1)
plt.stem(t, impulse_train)
plt.title('Impulse Train')
plt.xlabel('Time [s]')
plt.ylabel('Amplitude')

plt.subplot(3, 1, 2)
plt.plot(t, message_signal)
```

```
plt.title('Message Signal')
plt.xlabel('Time [s]')
plt.ylabel('Amplitude')

plt.subplot(3, 1, 3)
plt.stem(t, sampled_signal)
plt.title('Sampled Signal')
plt.xlabel('Time [s]')
plt.ylabel('Amplitude')

plt.tight_layout()
plt.show()

plt.figure(figsize=(10, 8))

plt.subplot(3, 1, 1)
plt.plot(f_axis, fft_message, 'C3')
plt.title('Spectrum of Message Signal')
plt.xlabel('Frequency [Hz]')
plt.ylabel('Magnitude')

plt.subplot(3, 1, 2)
plt.plot(f_axis, fft_impulse, 'C4')
plt.title('Spectrum of Impulse Train')
plt.xlabel('Frequency [Hz]')
plt.ylabel('Magnitude')

plt.subplot(3, 1, 3)
plt.plot(f_axis, fft_sampled, 'C5')
plt.title('Spectrum of Sampled Signal')
plt.xlabel('Frequency [Hz]')
plt.ylabel('Magnitude')

plt.tight_layout()
plt.show()
```

4 Results

The figure below shows the plots of the impulse train, the message signal, and the sampled signal in the time domain, as well as their corresponding frequency spectra in the frequency domain.

5 Conclusion

In conclusion, the experiment successfully demonstrated the process of sampling a sinusoidal signal using an impulse train. We verified that the sampling frequency was adequate for accurately reconstructing the original signal, adhering to the Nyquist theorem. Future experiments could explore different signal types and sampling frequencies to further understand the effects on signal fidelity.

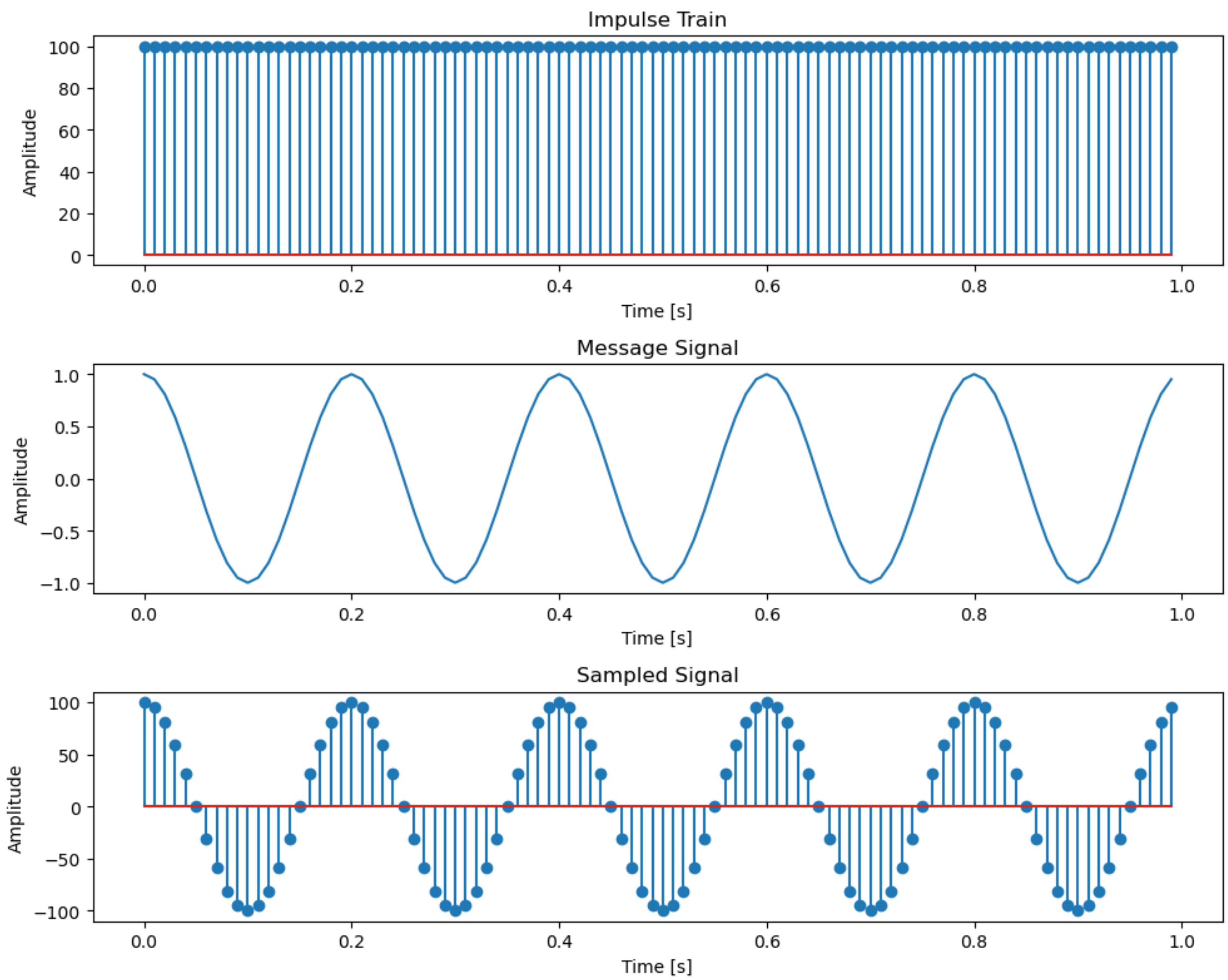

Figure 1: Impulse Train, Message Signal, and Sampled Signal in Time and Frequency Domains

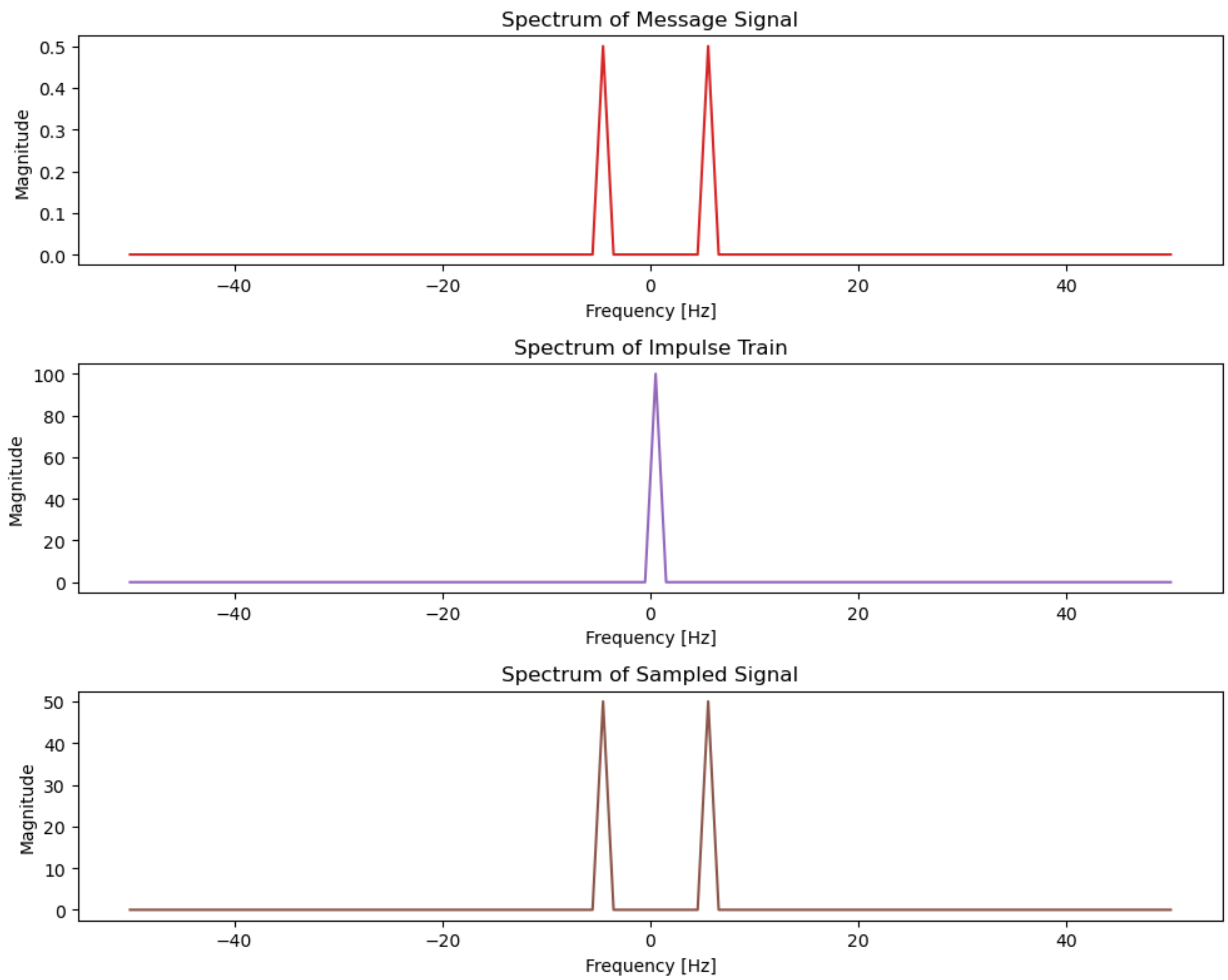

Figure 2: Impulse Train, Message Signal, and Sampled Signal in Time and Frequency Domains