Visual Demonstration of the

Nyquist Sampling Theorem

The Theorem

- *Nyquist sampling theorem*: a continuous-time signal must be sampled at a rate that is at least twice that of its fundamental frequency in order for that signal to be fully reconstructed from its samples (*Nyquist frequency*).

- Full reconstruction refers to the interpolation of the signal between its samples with total accuracy relative to the original continuous-time signal.

Sampling, Reconstruction, and Aliasing

- Undersampling, oversampling, and critical sampling refer to the sampling of a continuous-time signal above, below, and at the Nyquist frequency, respectively

- The sampled signal will suffer from aliasing when undersampled. Aliasing refers to the phenomenon that occurs when the interpolation between samples does not yield the original continuous-time signal as a result of the samples being too far apart in time

- When observed in the frequency domain, a sampled signal appears as a periodic version of its own frequency spectrum. When aliasing occurs, each of these periods overlaps with another

Overview of Experiments

- The goal of this project is to demonstrate the Nyquist sampling theorem for a continuous-time sinusoidal signal of unit amplitude

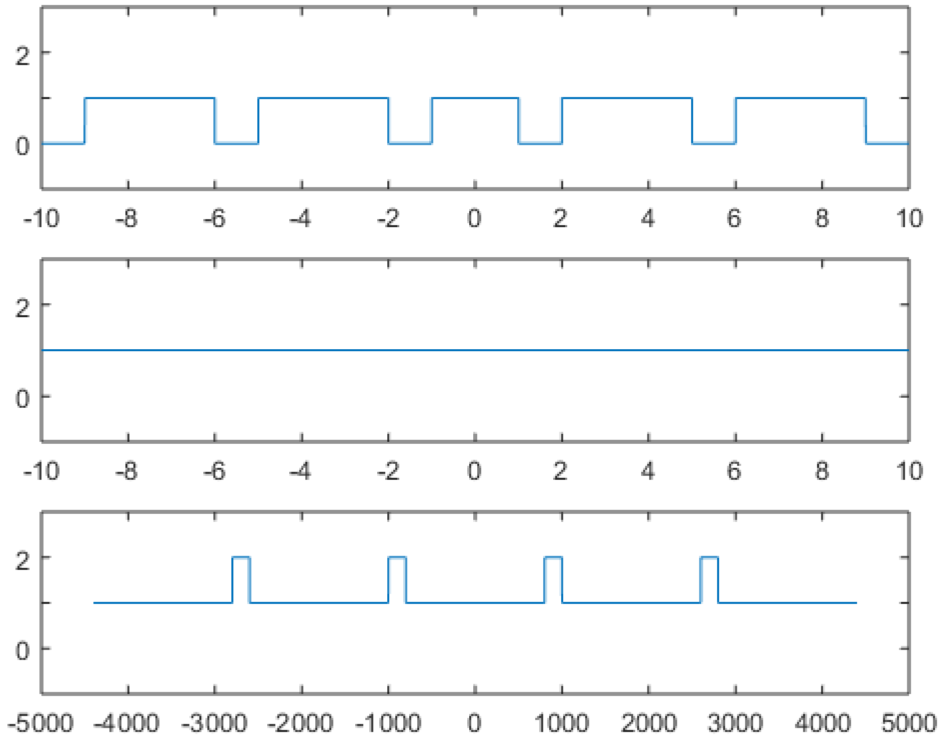
- The function, **nyquist_demo(function_type, bandwidth)**, accepts two arguments:

  - *function_type* (string) specifies the type of time-domain function for which the Nyquist rate is demonstrated; can be 'sine' or 'sinc’

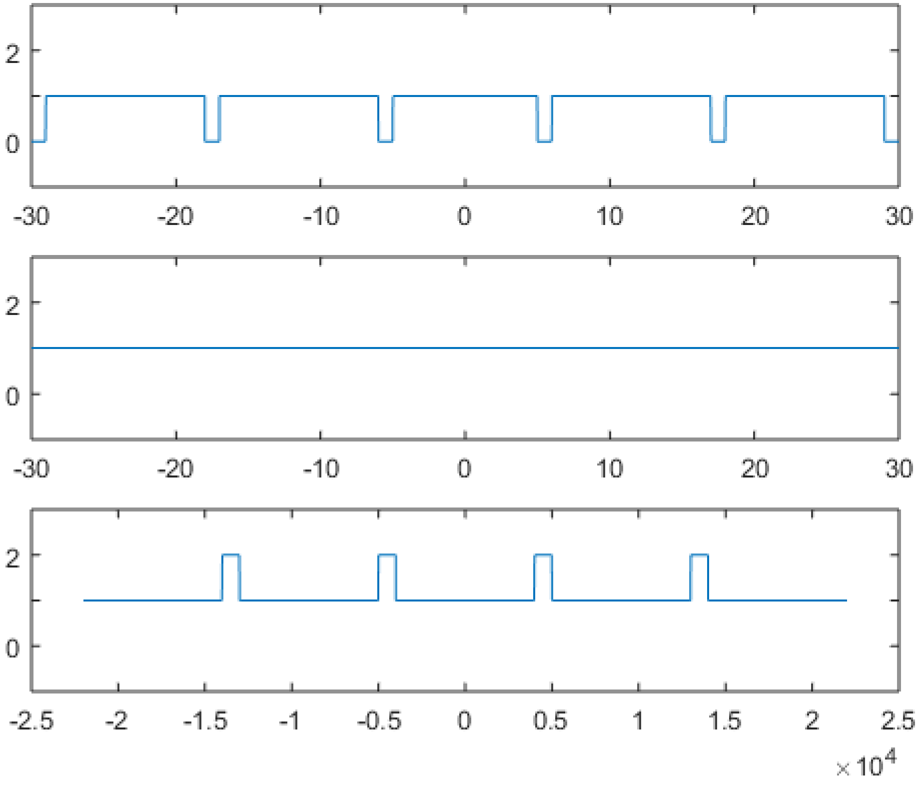
  - *bandwidth* (integer) specifies the bandwidth of the time-domain function in kHz

- The first experiment will demonstrate the cases of oversampling, critical sampling, and undersampling for a sine function. The second experiment will repeat the first for a sinc function.

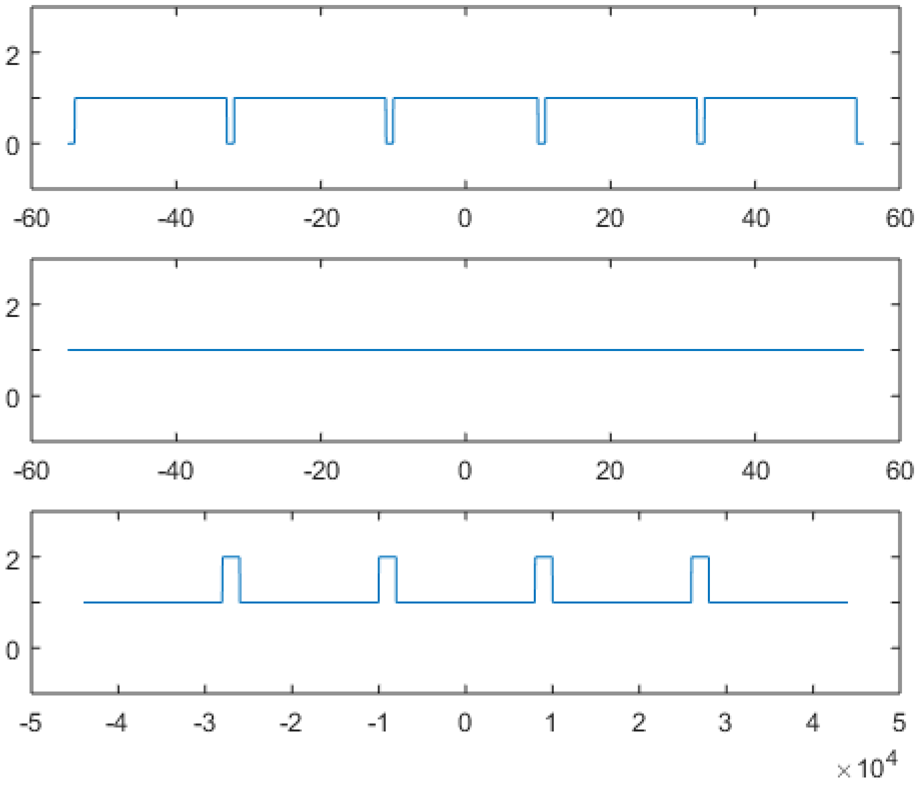
Experiment 1a: Sine Wave at 1 kHz



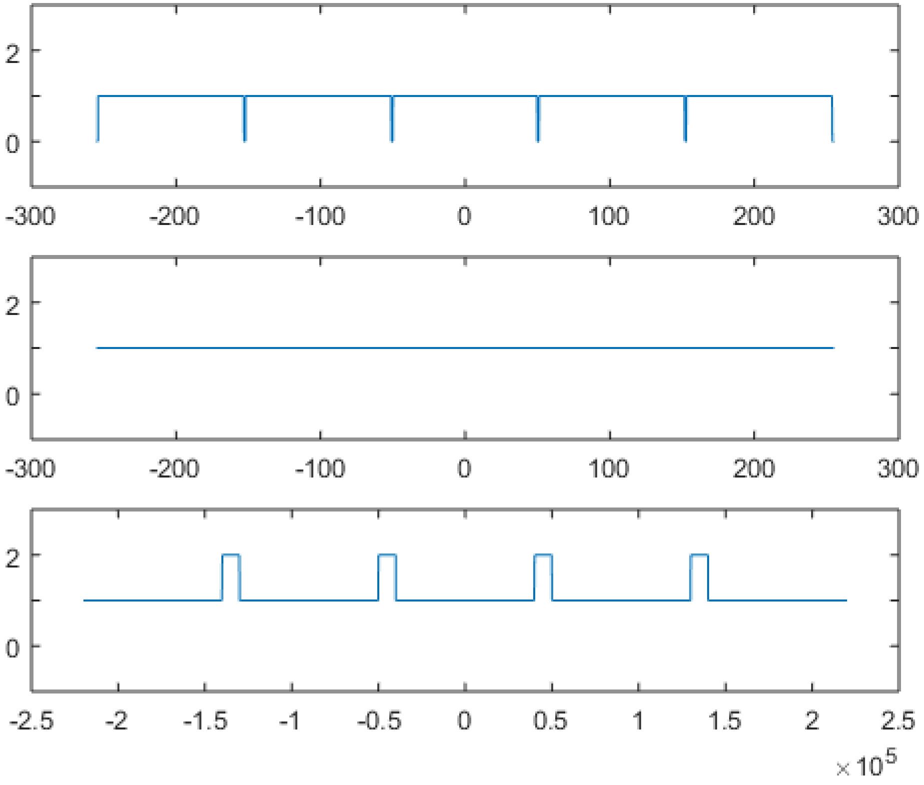
Experiment 1b: Sine Wave at 5 kHz



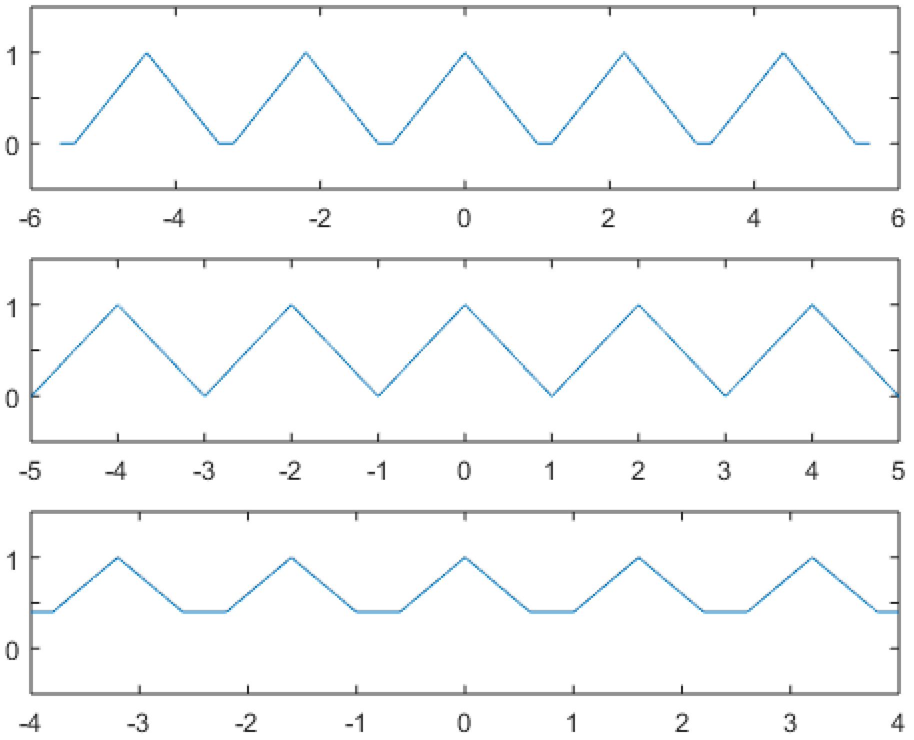
Experiment 1c: Sine Wave at 10 kHz



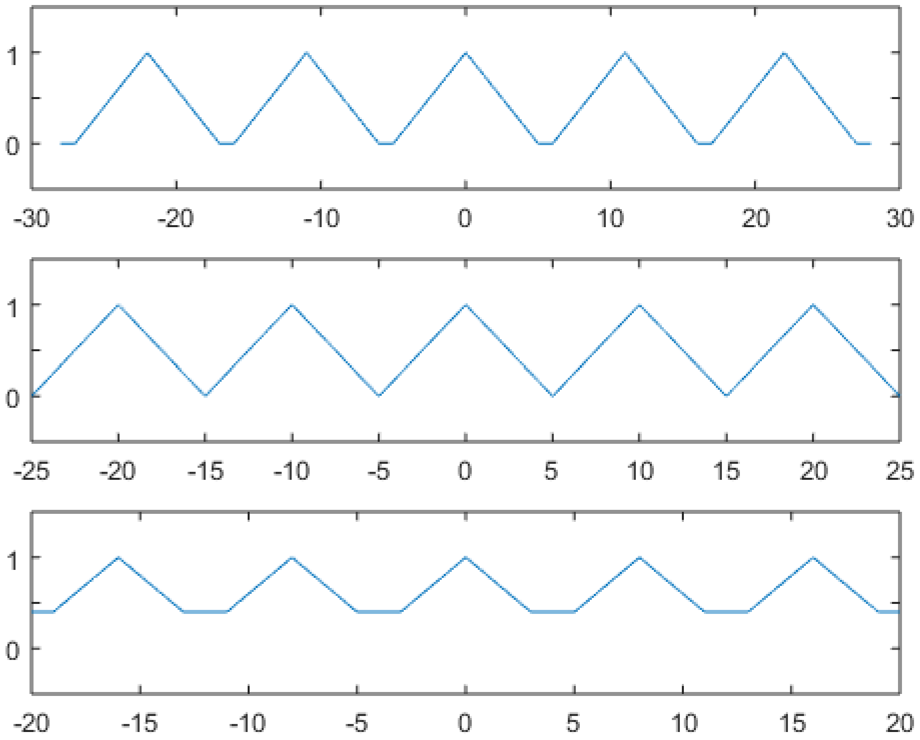
Experiment 1d: Sine Wave at 50 kHz



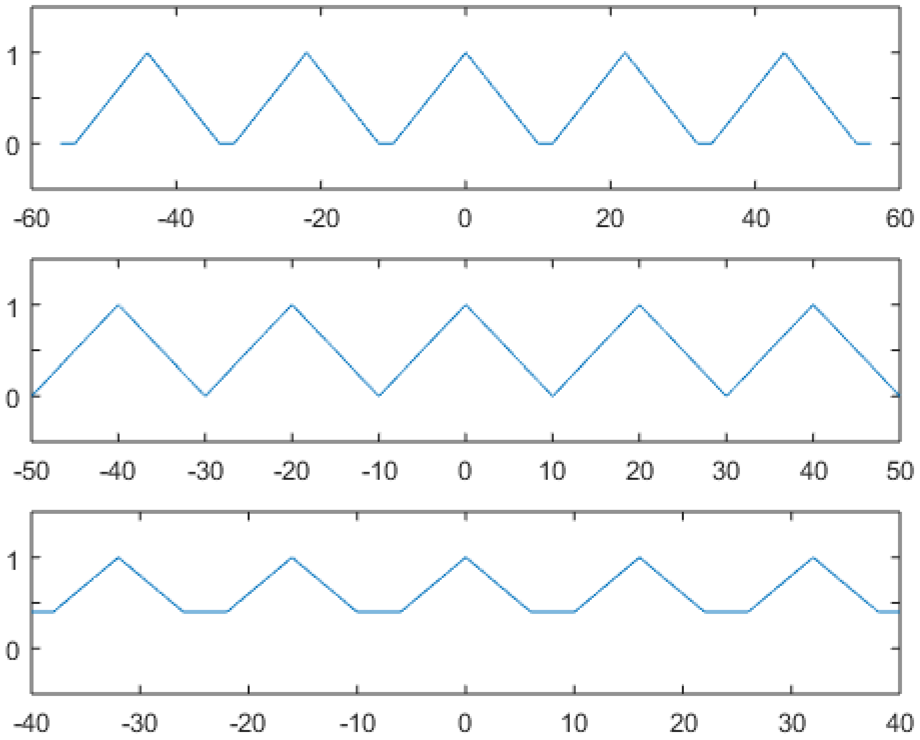
Experiment 2a: Sinc Wave at 1 kHz



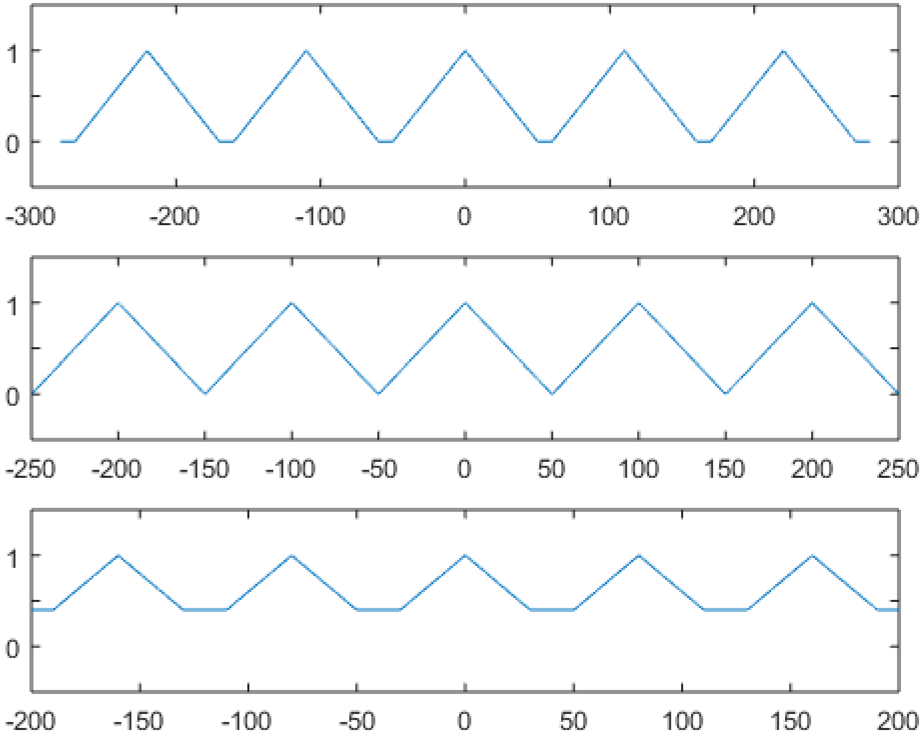
Experiment 2b: Sinc Wave at 5 kHz



Experiment 2c: Sinc Wave at 10 kHz



Experiment 2d: Sinc Wave at 50 kHz



MATLAB Scripts

```
function [] = nyquist_demo(function_type, bandwidth)
    % function_type (string): the type of time-domain function for which the
    % Nyquist rate is demonstrated; can be 'sine' or 'sinc'

    % bandwidth (integer): the bandwidth of the time-domain function in kHz

    nyquist_rate = 2*bandwidth;

    if strcmp(function_type, 'sine')
        f_range = 5*bandwidth + 5;
        f = -1*f_range:0.001:f_range;

        square = ones(1, 1000*(nyquist_rate + 1));
        oversampled = [zeros(1, 1000), square, zeros(1, 1000), square,...
            zeros(1, 1000), ones(1, 1000*bandwidth), 1,...
            ones(1, 1000*bandwidth), zeros(1, 1000), square,...
            zeros(1, 1000), square, zeros(1, 1000)];
        critsampled = ones(1, length(f));

        undersampled = [ones(1, 1000*(2*bandwidth*(1 - 0.2))), 2*ones(1, 1000*0.2*bandwidth),
ones(1, 1000*(2*bandwidth*(1 - 0.2))), 2*ones(1, 1000*0.2*bandwidth), ones(1, 1 +
1000*(2*bandwidth*(1 - 0.2))), 2*ones(1, 1000*0.2*bandwidth), ones(1, 1000*(2*bandwidth*(1 -
0.2))), 2*ones(1, 1000*0.2*bandwidth), ones(1, 1000*(2*bandwidth*(1 - 0.2)))];
        f_undersampled = -1000*bandwidth*(5 - 3*0.2) : 1000*bandwidth*(5 - 3*0.2);
```

MATLAB Scripts

```
        figure
        subplot(3, 1, 1)
        plot(f, oversampled)
        set(gca, 'YLim', [-1 3])
        subplot(3, 1, 2)
        plot(f, critsampled)
        set(gca, 'YLim', [-1 3])
        subplot(3, 1, 3)
        plot(f_undersampled, undersampled)
        set(gca, 'YLim', [-1 3])

    elseif strcmp(function_type, 'sinc')
        down_ramp = 1 : -1/1000 : 0;
        up_ramp = 0 : 1/1000 : 1;
        triangle = [up_ramp(1:1000), down_ramp(1:1000)];

        x_oversampled = -1*bandwidth*(5 + 3*0.2) : bandwidth/1000 : bandwidth*(5 + 3*0.2);
        x_critsampled = -1*bandwidth*(5) : bandwidth/1000 : bandwidth*(5);
        x_undersampled = -1*bandwidth*(5 - 5*0.2) : bandwidth/1000 : bandwidth*(5 - 5*0.2);

        oversampled = [zeros(1, 200), triangle, zeros(1, 200), triangle, zeros(1, 200),
up_ramp(1:1000), 1, down_ramp(1:1000), zeros(1, 200), triangle, zeros(1, 200), triangle, zeros(1,
```

MATLAB Scripts

```
200)];
        critsampled = [triangle, triangle, up_ramp(1:1000), 1, down_ramp(1:1000), triangle,
triangle];
        undersampled = [0.4*ones(1, 200), triangle(401:1600), 0.4*ones(1, 400),
triangle(401:1600), 0.4*ones(1, 400), up_ramp(401:1000), 1, down_ramp(1:600), 0.4*ones(1, 400),
triangle(401:1600), 0.4*ones(1, 400), triangle(401:1600), 0.4*ones(1, 200)];

        figure
        subplot(3, 1, 1);
        plot(x_oversampled, oversampled);
        set(gca, 'YLim', [-0.5 1.5])
        subplot(3, 1, 2);
        plot(x_critsampled, critsampled);
        set(gca, 'YLim', [-0.5 1.5])
        subplot(3, 1, 3);
        plot(x_undersampled, undersampled);
        set(gca, 'YLim', [-0.5 1.5])
    end
end
```

MATLAB Scripts

```
nyquist_demo('sine', 1)
nyquist_demo('sine', 5)
nyquist_demo('sine', 10)
nyquist_demo('sine', 50)
nyquist_demo('sinc', 1)
nyquist_demo('sinc', 5)
nyquist_demo('sinc', 10)
nyquist_demo('sinc', 50)
```

Summary & Conclusion

- In this project, the Nyquist sampling theorem was demonstrated for continuous-time sine waves and sinc waves of unity amplitude, at frequencies of 1 kHz, 5 kHz, 10 kHz, and 50 kHz

- The frequency spectrum of a sampled signal is the spectrum of that signal repeated at intervals equal to the sampling rate. For this reason, the minimum rate at which a signal must be sampled is twice that of its fundamental frequency in order to avoid the overlap of repeated instances of the spectrum