

Digital Signal Processing and Analysis of Signals in the Power Spectrum

by

Kevin Moran

Lab Partner : Cade Hermeston

Date of Experiment : Thursday, October 29th, 2020

University of Southern California

Aerospace and Mechanical Engineering Department

AME 341A : Mechoptronics

Abstract

Utilizing the power spectra for digital signal processing provides a efficient method for managing, interpreting, and transmitting data. By using a function generator to control various types of signals and a National Instruments data acquisition system with a custom virtual interface to collect data, signals were successfully measured and represented in the frequency domain in a concise manner. Using a sampling rate of 2500 Hz, a sine wave with frequency of 1500 Hz was purposely aliased and represented in the power spectrum with an observable frequency of 1001 Hz. Additionally, an unknown signal was replicated by analyzing fundamental frequencies and harmonics in the power spectrum. This unknown signal was determined to be comprised of a 777 Hz sine wave embedded in a 33 Hz triangle wave with 50% symmetry.

1 Introduction

Digital Signal Processing (DSP) is essential to making digital measurements of real-world signals, and components such as a analog to digital converter (ADC) allows for the measurement of natural signals via electronic devices. Furthermore, DSP is essential for communication applications and is a fundamental component of data acquisition systems found throughout almost every area of engineering. Processing, filtering, and transmitting an enormous amount of data as a time trace of voltage, however, is extremely difficult due to limitations set by computing power. Continuous signals can be transformed into a power spectrum via Fourier transformation, thus making data easier to manageable. For example, a sine wave of an AC current can represented by a large set of data depending on the sampling frequency and number of data point or duration of the measurement, however, this information can be represented in the power spectrum as a single point. The compact-ability of the signal simplifies data management and the information can be more easily transmitted, relative to the signal in the time domain.

1.1 Fourier Transformations

As mentioned, the relationship between the frequency and time domain can be described by Fourier transformation,

$$y(t) = A_0 + \sum_{n=1}^{\infty} A_n \cos(nwt) + B_n \sin(nwt) \quad (\text{Eq. 1})$$

where w and t represent the natural frequency and time, respectively. The variable n represents the harmonics associated with the signal and the special case where $n = 1$ is known as the fundamental frequency. The variables A_0 , A_n , and B_n can be expressed as the subset of

equations

$$\begin{aligned} A_0 &= \frac{1}{T} \int_{-T/2}^{T/2} y(t) dt \\ A_n &= \frac{2}{T} \int_{-T/2}^{T/2} y(t) \cos(n\omega t) dt \\ B_n &= \frac{1}{T} \int_{-T/2}^{T/2} y(t) \sin(n\omega t) dt \end{aligned}$$

where T is the period of the signal and A_0 is often referred to as the offset of a signal. The purpose of a Fourier transformation is to represent a continuous function as an infinite sum and, in the case of square and triangle waves, the systems harmonics.

In the scope of this paper, the Fourier transformation of a square wave can be described as

$$y(t) = \sum_{n=1,3,5,\dots}^{\infty} \frac{4}{n\pi} \sin(n\omega t) \quad (\text{Eq. 2})$$

and is strictly in terms of odd harmonics. Similarly, the Fourier transformation of a triangle wave can be described by the equation

$$y(t) = \sum_{n=1,3,5,\dots}^{\infty} \frac{8}{n^2\pi^2} \cos(n\omega t) \quad (\text{Eq. 3})$$

1.2 Nyquist Frequency

The Nyquist Frequency, defined by the relationship

$$f_{Ny} = \frac{f_s}{2} \quad (\text{Eq. 4})$$

is the largest observable frequency that can be measured before aliasing occurs. Attempting to resolve any signal beyond the Nyquist frequency in the time domain will result in a loss of shape and amplitude and *folds back*, as shown in Figure 2, on itself in the power spectrum. If a signal is aliased, the observable frequency can be represented as

$$f_{obs} = f_s - f \quad (\text{Eq. 5})$$

2 Methods and Materials

By using a National Instrument VirtualBench VB-8012 (VBench) to control input signals and a National Instrument data acquisition (NI-DAQ) system with a custom virtual interface

to collect data, various signals were able to be processed in the frequency and time domain. The experiment focused on three primary modes; analyzing a well known signal in the frequency domain, examining the response for a signal being aliased in the frequency domain and time domain, and determining the characteristics of an unknown signal in the frequency and time domain.

2.1 Power Spectrum Representation

To establish a foundational understanding for interpreting signals in the frequency domain, the VBench function generator was used to create a sine and square wave. Both of these signals were created using the parameters, $2 V_{rms}$ amplitude and a frequency of 100 Hz. For the sine wave, a sampling rate of 400Hz was chosen to ensure the no aliasing of the signal occurred. Collecting enough data for the square wave, however, required a higher sampling frequency of 10 kHz due to the harmonics expressed in Equation 2.

2.2 Aliasing of a Known Signal

To properly demonstrate the *fold back* type of behavior associated with aliasing a sine wave signal requires a proper measurement, therefore, a sampling frequency of 10 kHz was chosen to measure a 1500 Hz signal. This ensured that the maximum observable frequency was well beyond the Nyquist frequency described in Equation 4. Additionally, the same signal was also analyzed using the sampling frequency of 2.5 kHz. These two sampling frequencies ensured that data was collected under aliased and non-aliased conditions.

To further analyze a sine wave under aliasing conditions, the signal was analyzed in the time domain of the virtual interface using sampling frequencies of 2.5 kHz and 1.5 kHz. To conclude the experimental observations of an aliased sine wave in the time domain, the frequency on the function generator was changed in increments of 1 Hz on the interval from 1,500 Hz to 1,490 Hz.

2.3 Characterizing an Unknown Signal

Characterizing the unknown signal was done by setting the sampling rate to 20 kHz and examining the time trace for continuity and a well defined shaped, thus ensuring the signal was not being aliased. Additionally, the sampling frequency was set to 2,000 kHz and than 1,000 kHz and data sets were collected using the NI-DAQ in the frequency domain.

3 Results and Discussion

3.1 Analysis of Known Signal

Figure 1 shows the results for sine and square waves generated with the VBench in the frequency domain. Both signals satisfy the condition $f_s > f_{Ny}$ and thus ensure no aliasing is occurring. Furthermore, the fundamental frequency in Figure 1(a) corresponds to the same frequency on the function generator and harmonic frequencies in Figure 1(b) match the values associated with the Fourier transformation depicted in the Equation 2. That is the, harmonic frequencies exist as $3f$, $5f$, $7f$,

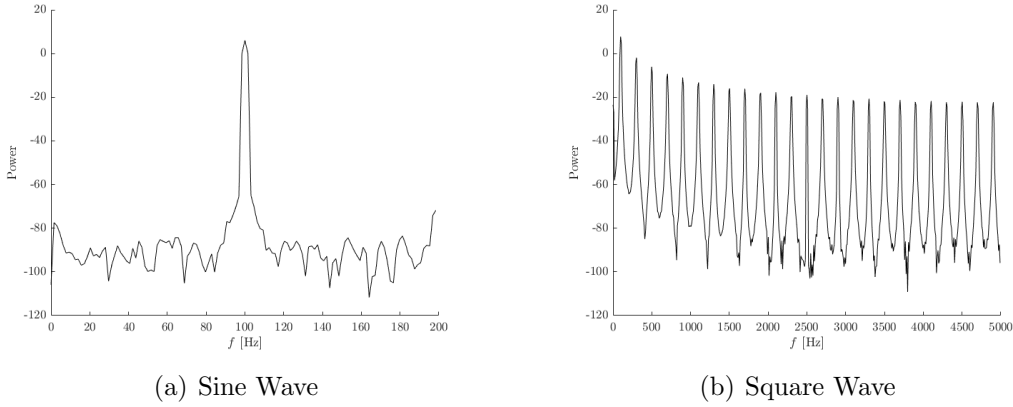


Figure 1: Sine and square waves with V_{rms} of 2 V at 100Hz shown in the frequency domain. Both of the signals are measured under non-aliasing conditions

Examining key characteristics of graphs shown in Figure 1 and Figure 2 reveals an agreement between theoretical predictions and experimental results for signals represented on a power spectra. In either case, signals generated via the VBench function generator were shown to have either single peaks or harmonics for sine and square waves, respectively.

3.2 Aliasing

On the contrary, Figure 2 shows a 1.5 kHz sine wave purposely being aliased by setting the sampling frequency to 10 kHz and then reducing it to 2.5 kHz. Since the Nyquis frequency of the latter is less than the measured frequency, these results reflect the observed frequency expressed in Equation 5 and illustrates the *fold back* type of behavior expected in a aliased signal.

Figure 3 illustrates the same signal in the time domain. This analysis indicated that using a sample frequency of 2.5 kHz is not sufficient enough to return the amplitude or shape of the signal, rather it only returns the frequency of the signal. Furthermore, Figure 3(b) illustrates the scenario $f_s = f_{signal}$. In this situation, the sampling rate is no longer

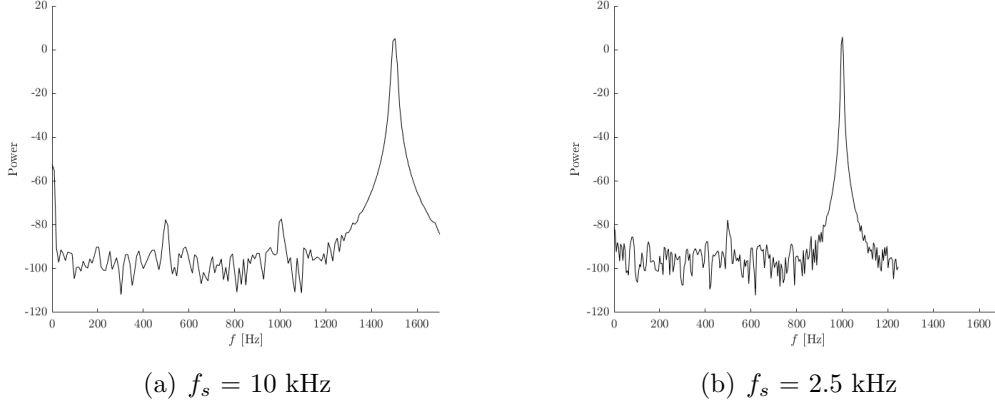


Figure 2: Demonstration of aliasing the frequency domain, where (a) and (b) are non-aliased and aliased signals, respectively. As per Equation 4, the f_{Ny} for (a) and (b) are 5 kHz and 1.25 KHz, and the observed frequency of (b) is describe by Equation 5 with a value of 1001 kHz.

sufficient enough to collect any sort of data from the signal(i.e., shape, offset, frequency, or amplitude). However, incrementally increasing the frequency at the function generator, as shown in Figure 3(c), seemed to restore the characteristics of the graph in the time domain.

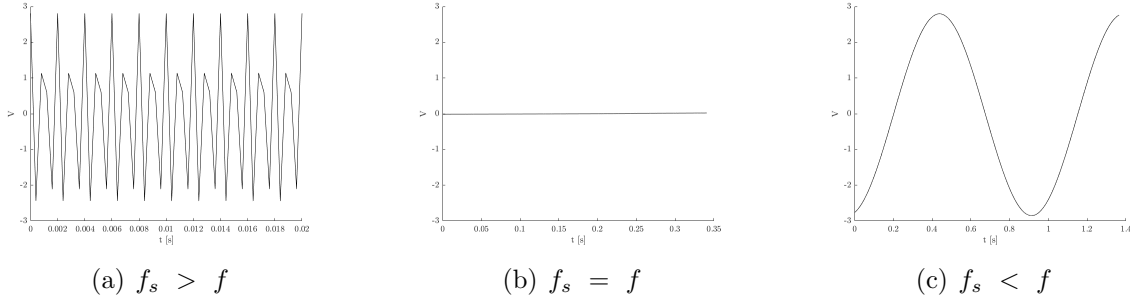


Figure 3: Sine and square waves with V_{rms} of 2V at 100Hz shown in the frequency domain

Analysis of known signals in the time domain also qualitative matched theoretical predictions. From Figure 3(a), it is shown that the observable frequency is equivalent to the signal via the function generator. The shape and amplitude, however, cannot be recovered and is a consequence of the Nyquins frequency being insufficient to properly measure the incoming signal. When adjusting the sampling rate to match the signal frequency, it is shown in Figure 3(b) that the time trace will consist of a straight line. This phenomenon was due to the NI-DAQ measuring the same point on the curve every cycle and, in this experiment, the measured point also aligned to the point where the amplitude of the sine wave was equal to zero. Adjusting the signal frequency such that $f_s < f$, however, did not align with the anticipated results in the time domain. The anticipated result was a time trace with no particular shape, amplitude, nor frequency but the results shown in Figure 3(c) shows an entirely different outcome. Experimental data shows a sinusoidal wave with a frequency of approximately 1 Hz.

3.3 Analysis of Unknown Signal

Plotting the time trace for the unknown signal revealed a high frequency component embedded in a low frequency signal.

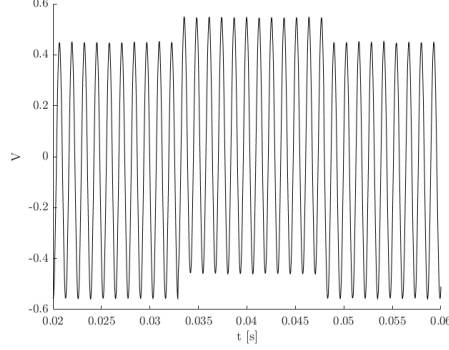


Figure 4: Time trace of unknown signal

The power spectra of the unknown signal was examined at a sampling frequencies of 1 kHz and 2 kHz. Comparing the two samples reveals that the former of the sampling frequencies aliases the signal and is most apparent by the shift in the largest peak.

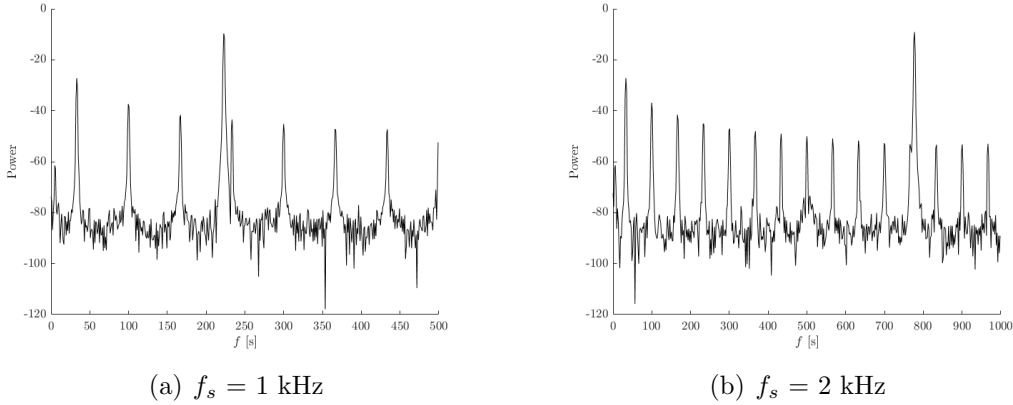


Figure 5: Power Spectrum for unknown signal

The time trace shown in Figure 4 shows a minimum of two combined signal patterns, with one at greater frequency than the other. Viewing the signal in the power spectra confirmed the existence of a low frequency signal with harmonics (i.e., square or triangle) and a high frequency signal with no harmonics (i.e., sinusoidal wave). Measuring the signal at a sampling rate of 1 kHz and 2 kHz showed that the signal was being aliased at 1 kHz and matched the *fold back* behavior for a sine wave at 777Hz and a Nyquist frequency of 500 Hz. In Figure 5(a), the observed frequency of the sine wave occurs at 223 Hz \pm df and this observation matches the theoretical outcome described in Equation 5.

The harmonics observed in the unknown signal matched to known signals via trial and

error. It was assumed that either a triangle or square wave were the underlying type of wave associated with the measurements obtained in the power spectra. Figure 6 shows the measured signal in blue and a replicated signal in black. The replicated signal was created using the function generated and combining two sets of data. The first set of data was obtained by creating a sine wave with a frequency of 777 Hz and the second data set was obtained by generating a triangle wave at 33 Hz and 50% symmetry. A side by side qualitatively shows that replicated signal matches harmonic frequency of the unknown signal. The amplitude of the signals, however, differ in magnitude and the replicated signal carries more energy than the unknown signal.

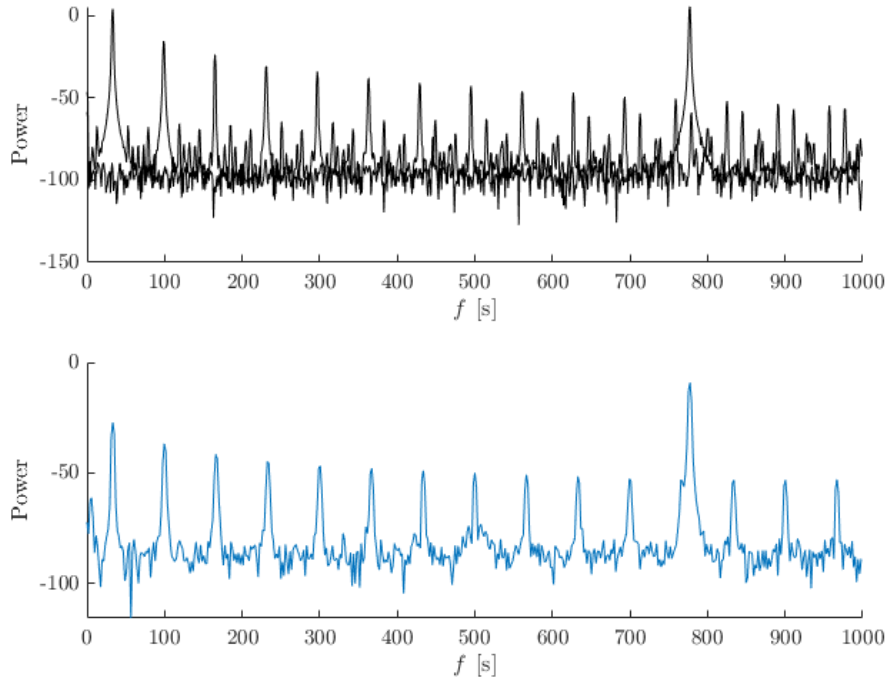


Figure 6: Guess and Actual

4 Conclusion

Representing signals in the frequency domain is an efficient method for managing large sets of data. Rather than representing a signal with a relatively large set of data points, a signal can be represented by a fundamental frequency and, if applicable, a series of harmonic frequencies. Additionally, analyzing unknown frequencies in the power spectrum provides a simplified approach to identifying various types of waves embedded in a signal.

A MATLAB Script

```
1 clear;clc;close all
2 % ----- Import Data
   -----
3 % data(:,1) = time/freq
4 % data(:,2) = power/voltage
5 part1sin = importdata('Data Lab 7/Part1_sin.txt').data;
6 part1square = importdata('Data Lab 7/Part1_square.txt').data;
7 part2sin = importdata('Data Lab 7/Part2_sin.txt').data;
8 part2sinAliasing = importdata('Data Lab 7/Part2_sinAliasing.txt
   ').data;
9 part2Final = importdata('Data Lab 7/Part2_timeDomainFinal.txt')
   .data;
10 part2Initial = importdata('Data Lab 7/Part2_timeDomainInitial.
   txt').data;
11 part2Freq = importdata('Data Lab 7/Part2_timeDomainFreqChange.
   txt').data;
12 part31000 = importdata('Data Lab 7/Part3_1000Hz.txt').data;
13 part32000 = importdata('Data Lab 7/Part3_2000Hz.txt').data;
14 part3time = importdata('Data Lab 7/Part3_timeDomain.txt').data;
15 sinwave777 = importdata('Data Lab 7/sinwave_777Hz.txt').data;
16 triangle33 = importdata('Data Lab 7/triangle_33Hz.txt').data;
17
18 %% Figures From Part 1
19 % ----- Part 1 Sine Wave -----
20 figure()
21 hold on;
22 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
   , 'latex');
23 plot(part1sin(:,1),part1sin(:,2),'k')
24 xlabel('$f$ [Hz]');
25 ylabel('Power');
26
27 % ----- Part 1 Square Wave -----
28 figure()
29 hold on;
30 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
   , 'latex');
31 plot(part1square(:,1),part1square(:,2),'k')
32 xlabel('$f$ [Hz]');
33 ylabel('Power');
34
35 %% Figures From Part 2
```

```

36 % ----- Sine Wave at 10kHz Sampling Rate -----
37 figure()
38 hold on;
39 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
    , 'latex');
40 plot(part2sin(:,1),part2sin(:,2),'k')
41 xlabel('$f$ [Hz]');
42 xlim([0 1700])
43 ylabel('Power');
44
45 % ----- Sine Wave at 2.5kHz Sampling Rate -----
46 figure()
47 hold on;
48 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
    , 'latex');
49 plot(part2sinAliasing(:,1),part2sinAliasing(:,2),'k')
50 xlabel('$f$ [Hz]');
51 xlim([0 1700])
52 ylabel('Power');
53
54 % ----- Sine Wave time domain initial -----
55 figure()
56 hold on;
57 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
    , 'latex');
58 plot(part2Initial(:,1),part2Initial(:,2),'k')
59 xlabel('t [s]');
60 xlim([0 .02])
61 ylabel('V');
62 ylim([-3 3])
63
64 % ----- Sine Wave time domain final -----
65 figure()
66 hold on;
67 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
    , 'latex');
68 plot(part2Final(:,1),part2Final(:,2),'k')
69 xlabel('t [s]');
70 ylabel('V');
71 ylim([-3 3])
72 % ----- Sine wave in time domain with increase Freq -----
73 figure()
74 hold on;
75 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
    , 'latex');

```

```

76 plot(part2Freq(:,1),part2Freq(:,2),'k')
77 xlabel('t [s]');
78 ylabel('V');
79 ylim([-3 3])
80
81 %% Figures for part 3
82 % ----- Mystery Signal in time domain -----
83 figure()
84 hold on;
85 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
    , 'latex');
86 plot(part3time(:,1),part3time(:,2),'k')
87 xlim([.02 .06])
88 xlabel('t [s]');
89 ylabel('V');
90
91 % ----- Mystery Signal f_s at 2000 Hz -----
92 figure()
93 hold on;
94 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
    , 'latex');
95 plot(part32000(:,1),part32000(:,2),'k')
96 xlabel('$f$ [s]');
97 ylabel('Power');
98
99 % ----- Mystery Signal f_s at 1000 Hz -----
100 figure()
101 hold on;
102 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
    , 'latex');
103 plot(part31000(:,1),part31000(:,2),'k')
104 xlabel('$f$ [s]');
105 ylabel('Power');
106
107 %% The Extra Graphs
108 figure()
109
110 subplot(2,1,1)
111 hold on;
112 set(gca,'defaulttextinterpreter','latex','TickLabelInterpreter'
    , 'latex');
113 plot(sinwave777(:,1),sinwave777(:,2),'k')
114 plot(triangle33(:,1),triangle33(:,2),'k')
115 xlabel('$f$ [s]');
116 ylabel('Power');

```

```
117
118 subplot(2,1,2)
119 hold on;
120 set(gca, 'defaulttextinterpreter', 'latex', 'TickLabelInterpreter'
    , 'latex');
121 plot(part32000(:,1), part32000(:,2))
122 xlabel('$f$ [s]');
123 ylabel('Power');
```