JHU Communication Systems

Lab 1

Hunter Mills

Introduction

Matlab is an important tool to master in learning communication systems. It has many features such as FFT, IFFT, plotting and matrix operations that are helpful in analyzing signals. In this lab I will plot some simple functions, their spectrum then some more complected spectrum. The results will be discussed along side of the figures.

Results

Figure 1 is showing the three signals that were created, a 1Hz sine wave, a rectangular pulse and a triangular pulse.

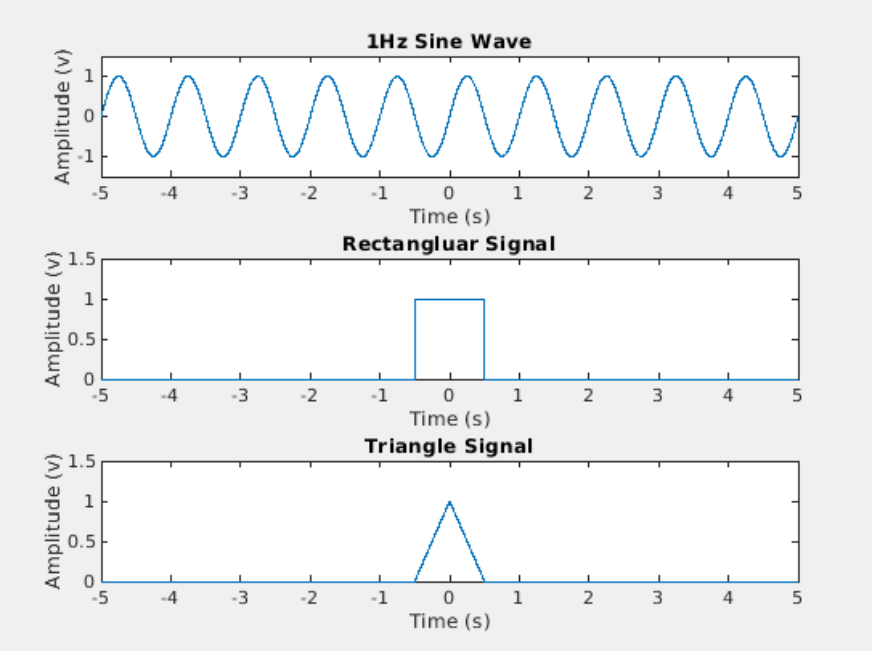
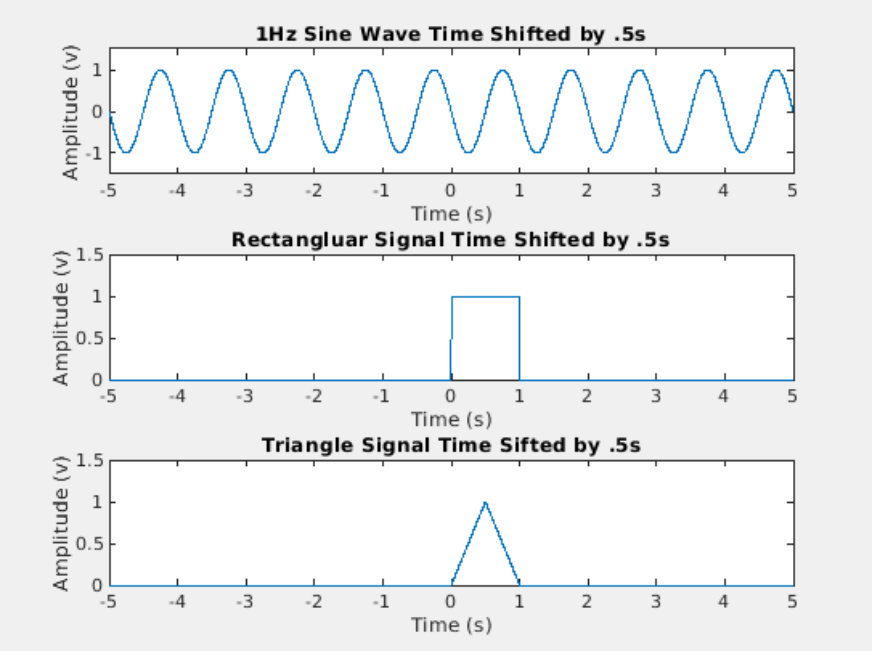
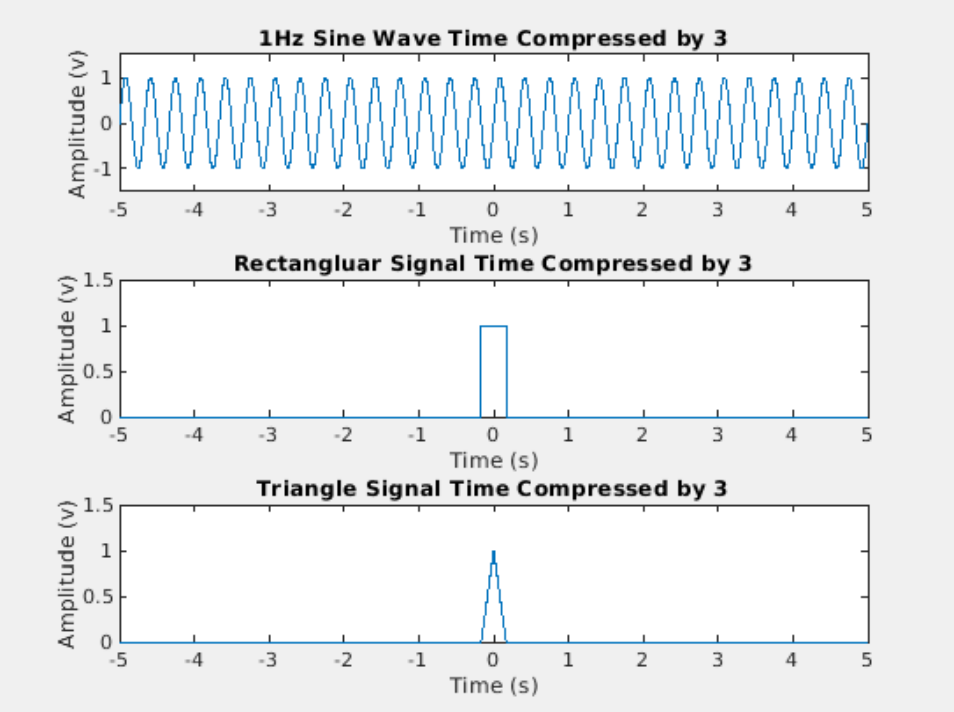


Figure two is going to show the same signals shown in figure 1 time shifted by .5 seconds.

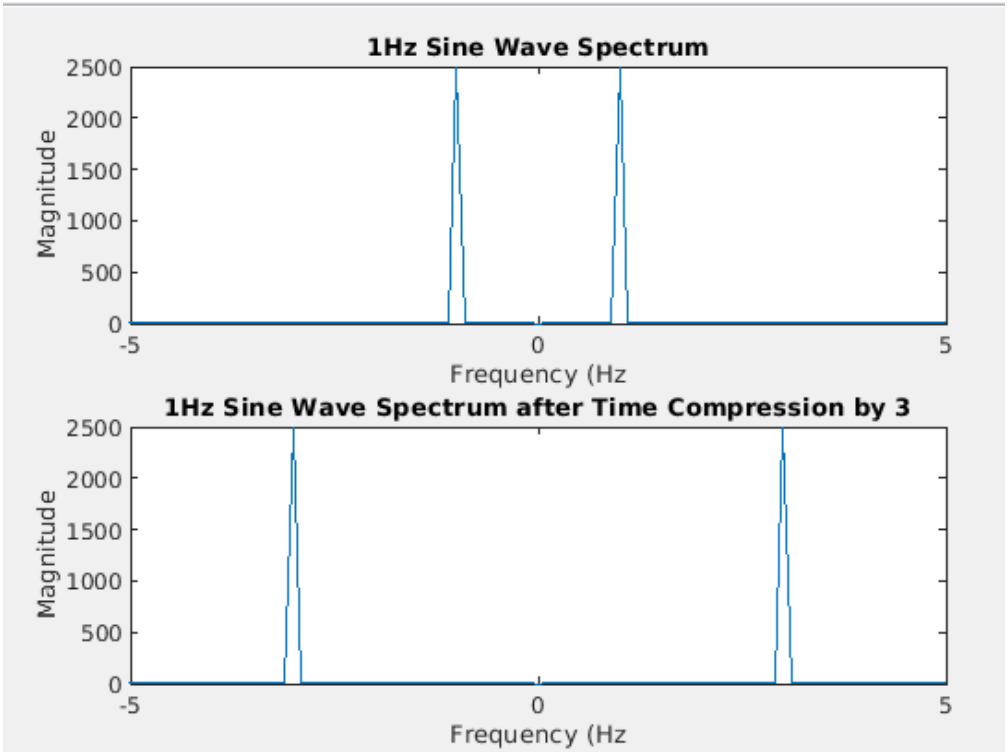


The time shifting operation on the sine wave created a phase shift of the initial wave. This causes a 180degree phase shift since the delay was .5s and the frequency is 1Hz. The rectangular and triangle signals shifted by .5 seconds moves it from being centered around the origin to being a causal signal. Figure 3 is showing the signals shown in figure 1 time compressed by a factor of 3.

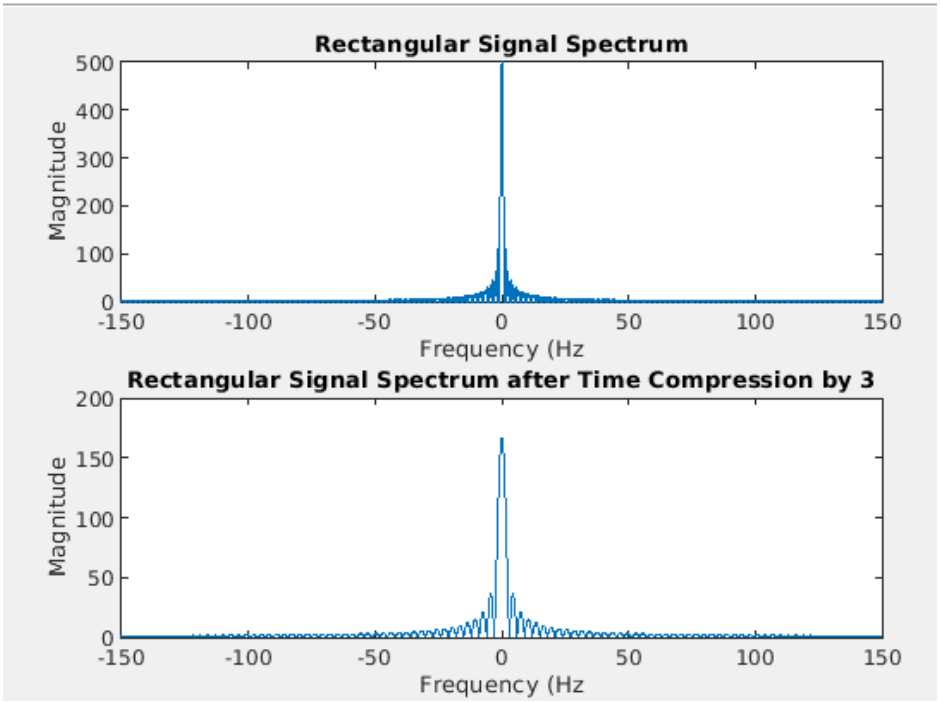


These results make sense because when you convert from x(t) to x(2t) the time axis is compressed by a factor of 2. So in the figure the sine wave has 3x more oscillations and the rectangular/triangular signals are compressed by a factor of 3.

Figure 4 is showing the spectrum of the 1Hz sine wave and the time compressed 1Hz sine wave.



This result shows that when you compress the time of a waveform the spectrum is expanded by that same factor. The next two figures show the same result. Figure 5 shows the spectrum of the rectangular spectrum (normal and compressed time) and the spectrum of the triangular spectrum (normal and compressed time).



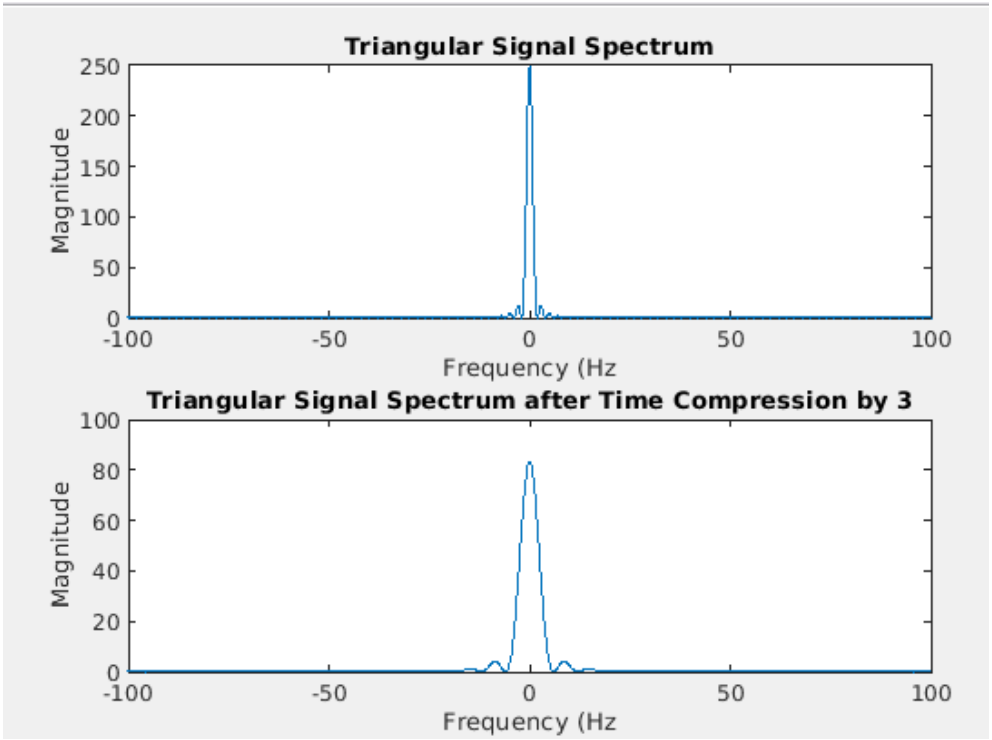
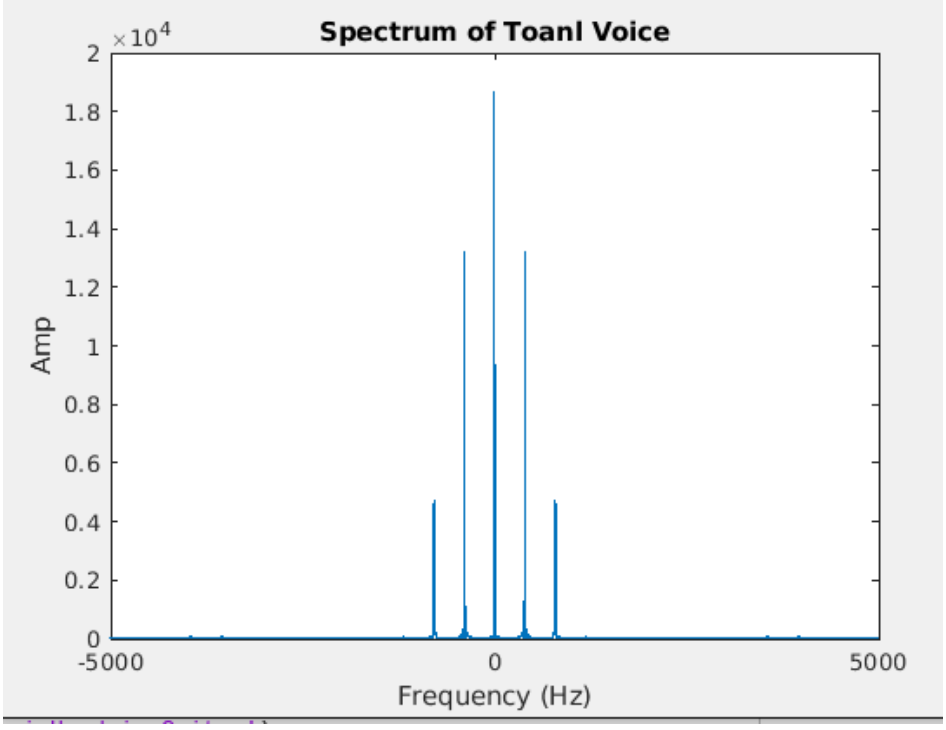
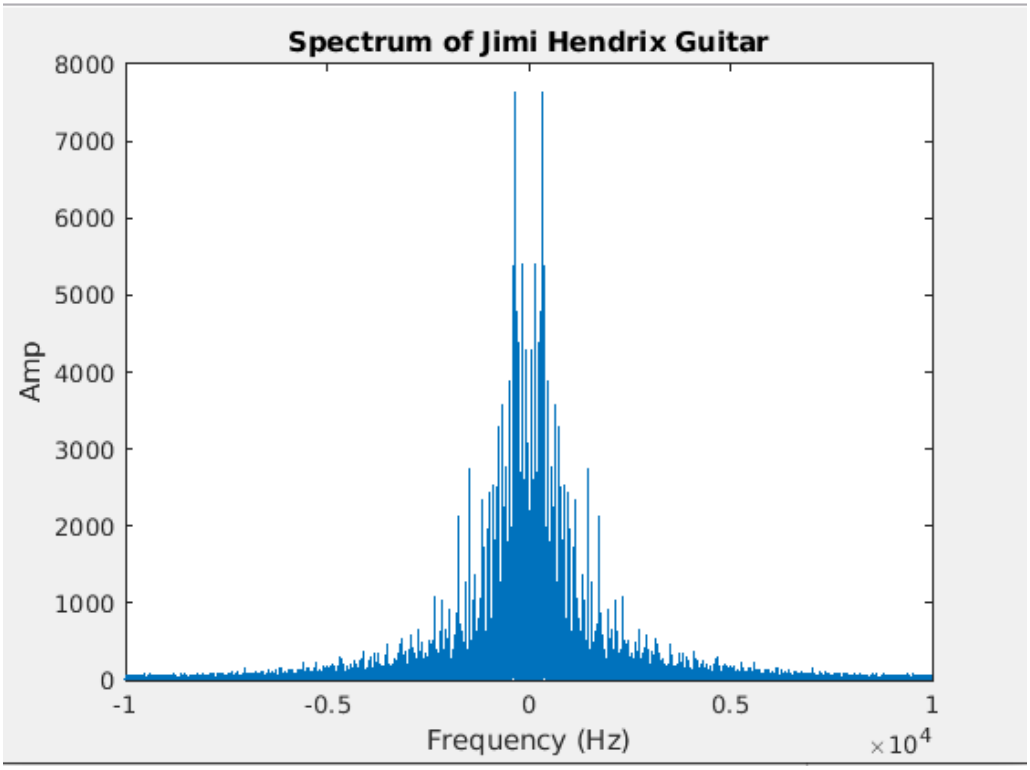


Figure 7 shows the spectrum of the tonal voice. The figure shows that there is a large DC offset in the signal. There are also tones at 400 and 800Hz. In this signal the audio read passes back a NumSamp x 2 and both columns are the same data.



The magnitude of the spectrum is even since the input signal is real. The last figure is the spectrum of the Jimmi Hendrix guitar. As expected this has many more frequency components in it than the tonal voice. Its main frequency components are within 5kHz. (Is the wav file complex?)



Conclusion

In conclusion matlab is a great tool to use when trying to create and manipulate signals. The FFT tools are handy to let you look at the spectrum and how manipulations effect the frequency of the signals.

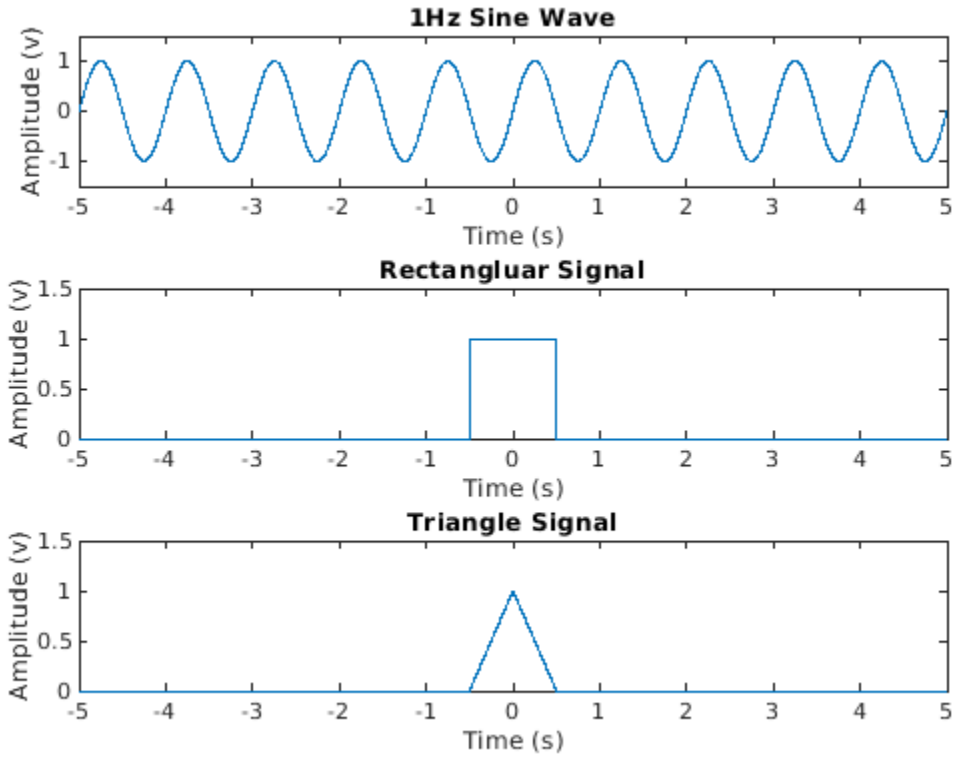
Appendix

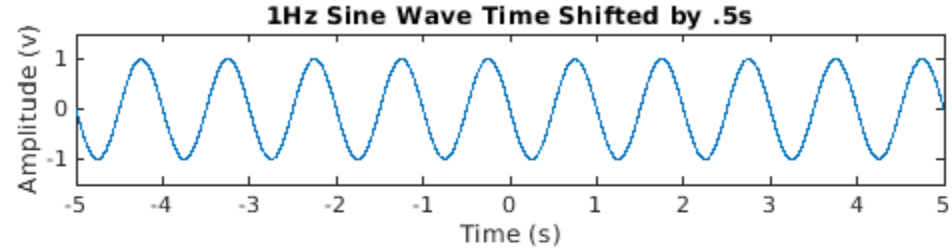
```
%!-------------------------------------------------------------------------
----------------------
%! @file      lab1.m
%! @author    Hunter Mills
%! @date      Febuary 2024
%! @brief     Steps :
%!              1 : Plot over -5:5 seconds
%!                  1 Hz sine wave
%!                  Unit rectangle
%!                  Unit triangle
%!              2 : Plot each singal with
%!                  a) Time sift of .5 seconds
%!                  b) Time Scaling with compression of 3
%!              3 : Signal Spectra
%!                  a) Plot spectrum of signals in 1 and 2b
%!                  b) Plot and observe the spectrum of tonal voice
%!                  c) Plot spectrum of another wave file
%!
%!-------------------------------------------------------------------------
----------------------

%! Adding path to functions
addpath(genpath('functions'));

%! Create Signals for step (1)
t           = linspace(-5, 5, 5000);
sin_wave    = sin(2*pi*t);
rect        = rectangle_signal(t, -.5, .5, 0);
tri         = triangle_signal(t);

%! Plot Signals from step (1)
figure(1);
subplot(3, 1, 1);
plot(t, sin_wave)
xlabel('Time (s)')
ylabel('Amplitude (v)')
ylim([-1.5, 1.5])
title('1Hz Sine Wave')
subplot(3, 1, 2);
plot(t, rect)
xlabel('Time (s)')
ylabel('Amplitude (v)')
title('Rectangluar Signal')
ylim([0, 1.5])
subplot(3, 1, 3);
plot(t, tri)
xlabel('Time (s)')
ylabel('Amplitude (v)')
title('Triangle Signal')
ylim([0, 1.5])

%! Create signals for step (2a)
```

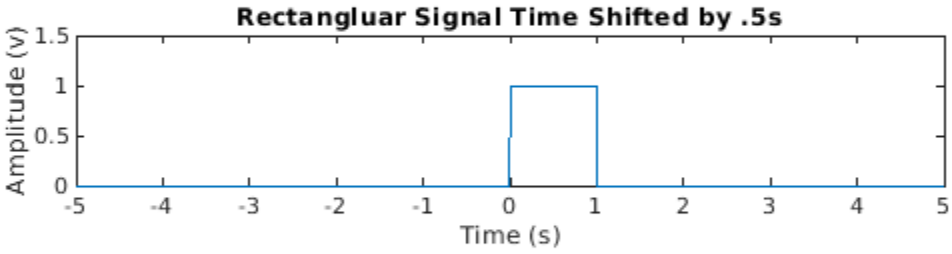
```
sin_delay  = sin(2*pi*(t - .5));
rect_delay = rectangle_signal(t, -.5, .5, .5);
tri_delay  = triangle_signal(t-.5);

%! Plot Signals from step (2a)
figure(2);
subplot(3, 1, 1);
plot(t, sin_delay)
xlabel('Time (s)')
ylabel('Amplitude (v)')
ylim([-1.5, 1.5])
title('1Hz Sine Wave Time Shifted by .5s')
subplot(3, 1, 2);
plot(t, rect_delay)
xlabel('Time (s)')
ylabel('Amplitude (v)')
title('Rectangluar Signal Time Shifted by .5s')
ylim([0, 1.5])
subplot(3, 1, 3);
plot(t, tri_delay)
xlabel('Time (s)')
ylabel('Amplitude (v)')
title('Triangle Signal Time Sifted by .5s')
ylim([0, 1.5])

%! Create signals for step (2b)
sin_com  = sin(2*pi*(t*3));
rect_com = rectangle_signal(t*3, -.5, .5, 0);
tri_com = triangle_signal(t*3);

%! Plot Signals from step (2b)
figure(3);
subplot(3, 1, 1);
plot(t, sin_com)
xlabel('Time (s)')
ylabel('Amplitude (v)')
ylim([-1.5, 1.5])
title('1Hz Sine Wave Time Compressed by 3')
subplot(3, 1, 2);
plot(t, rect_com)
xlabel('Time (s)')
ylabel('Amplitude (v)')
title('Rectangluar Signal Time Compressed by 3')
ylim([0, 1.5])
subplot(3, 1, 3);
plot(t, tri_com)
xlabel('Time (s)')
ylabel('Amplitude (v)')
title('Triangle Signal Time Compressed by 3')
ylim([0, 1.5])

%! Create signals from step (3a)
fs              = 500;
L               = 5000;
```

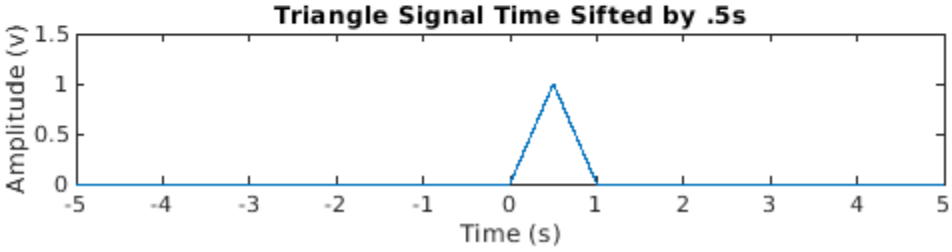
```
f               = fs * (-L/2:L/2-1)/L;
sin_fft         = fftshift(abs(fft(sin_wave)));
rect_fft        = fftshift(abs(fft(rect)));
tri_fft         = fftshift(abs(fft(tri)));
sin_fft_comp    = fftshift(abs(fft(sin_com)));
rect_fft_comp   = fftshift(abs(fft(rect_com)));
tri_fft_comp    = fftshift(abs(fft(tri_com)));

%! Plot Signals from step (3a)
figure(4);
subplot(2, 1, 1);
plot(f, sin_fft)
xlabel('Frequency (Hz')
ylabel('Magnitude')
xlim([-5, 5])
title('1Hz Sine Wave Spectrum')
subplot(2, 1, 2);
plot(f, sin_fft_comp)
xlabel('Frequency (Hz')
ylabel('Magnitude')
title('1Hz Sine Wave Spectrum after Time Compression by 3')
xlim([-5, 5])
figure(5);
subplot(2, 1, 1);
plot(f, rect_fft)
xlabel('Frequency (Hz')
ylabel('Magnitude')
xlim([-150, 150])
title('Rectangular Signal Spectrum')
subplot(2, 1, 2);
plot(f, rect_fft_comp)
xlabel('Frequency (Hz')
ylabel('Magnitude')
xlim([-150, 150])
title('Rectangular Signal Spectrum after Time Compression by 3')
figure(6);
subplot(2, 1, 1);
plot(f, tri_fft)
xlabel('Frequency (Hz')
ylabel('Magnitude')
title('Triangular Signal Spectrum')
xlim([-100, 100])
subplot(2, 1, 2);
plot(f, tri_fft_comp)
xlabel('Frequency (Hz')
ylabel('Magnitude')
title('Triangular Signal Spectrum after Time Compression by 3')
xlim([-100, 100])

%! Plot Tonal voice (3b)
[tonal, tonal_fs] = audioread("sounds/tonal_voice.wav");
figure(7)
f = tonal_fs * (-length(tonal)/2:length(tonal)/2-1)/length(tonal);
plot(f, fftshift(abs(fft(tonal(:, 1)))))
```

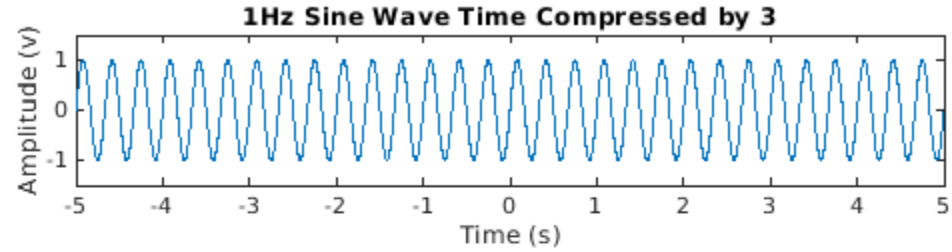
```
title('Spectrum of Toanl Voice')
xlabel('Frequency (Hz)')
ylabel('Amp')
xlim([-5000, 5000])

%! Plot (3c)
[song, fs] = audioread("sounds/jimi_hendrix_guitar.wav");
figure(8)
f = fs * (-length(song)/2:length(song)/2-1)/length(song);
plot(f, fftshift(abs(fft(song(:, 1)))))
title('Spectrum of Jimi Hendrix Guitar')
xlabel('Frequency (Hz)')
ylabel('Amp')
xlim([-10000, 10000])
```

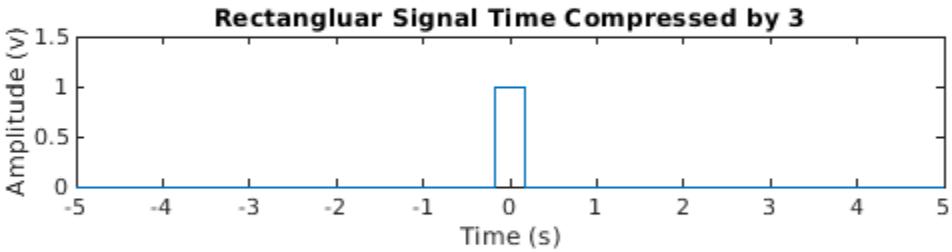


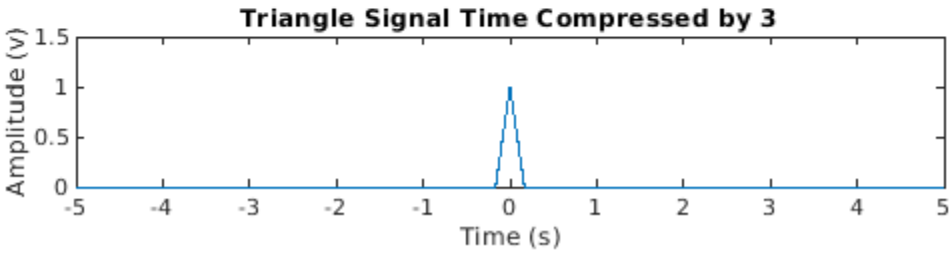

1Hz Sine Wave Time Shifted by .5s
Amplitude (v)
Time (s)

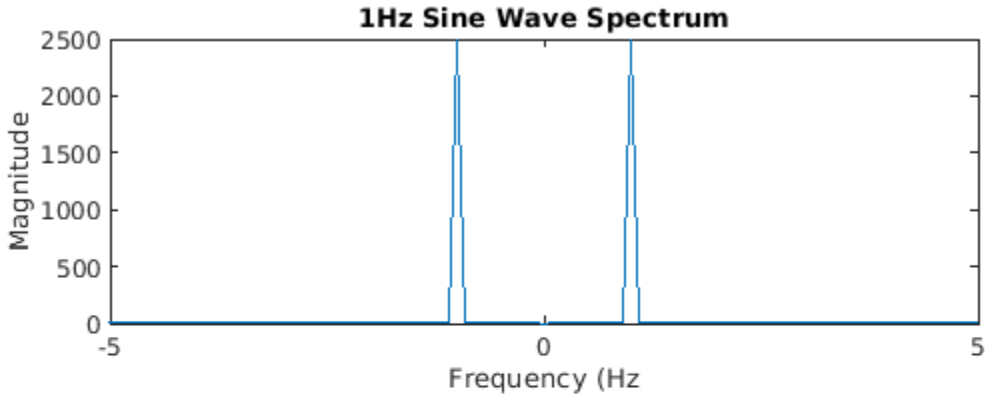

Rectangluar Signal Time Shifted by .5s
Amplitude (v)
Time (s)

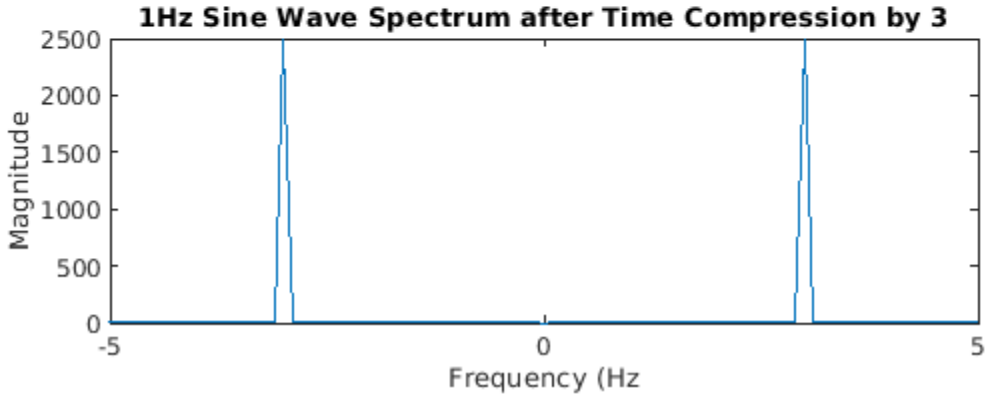

Triangle Signal Time Sifted by .5s
Amplitude (v)
Time (s)

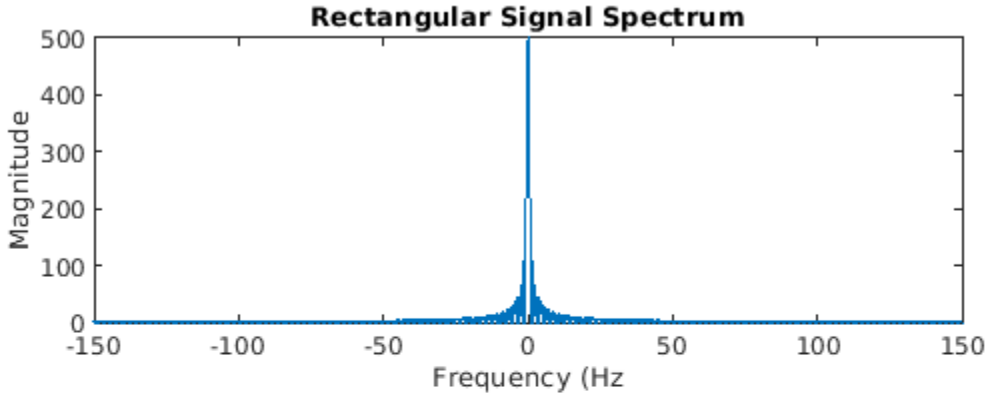

1Hz Sine Wave Time Compressed by 3
Amplitude (v)
Time (s)

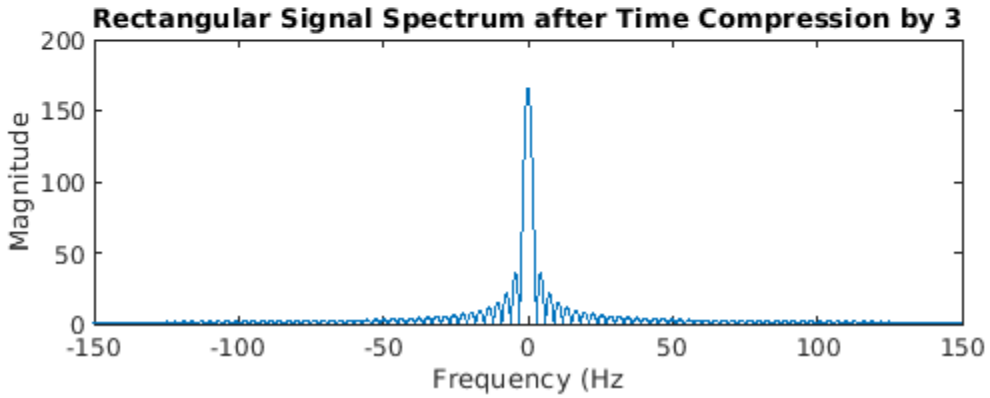

Rectangluar Signal Time Compressed by 3
Amplitude (v)
Time (s)

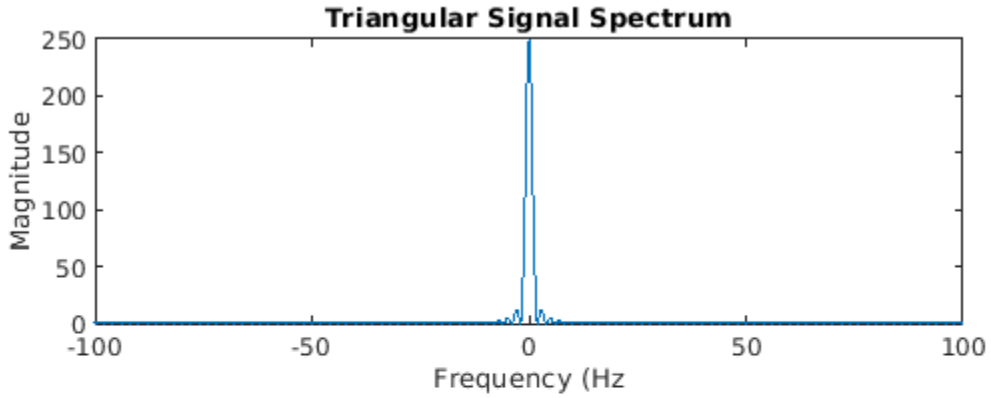

Triangle Signal Time Compressed by 3
Amplitude (v)
Time (s)

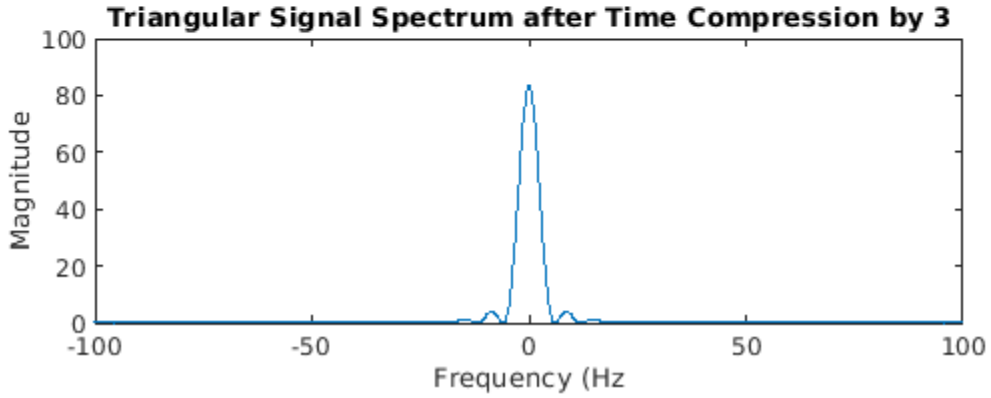

1Hz Sine Wave Spectrum
Magnitude
2500
2000
1500
1000
500
0
-5
0
5
Frequency (Hz

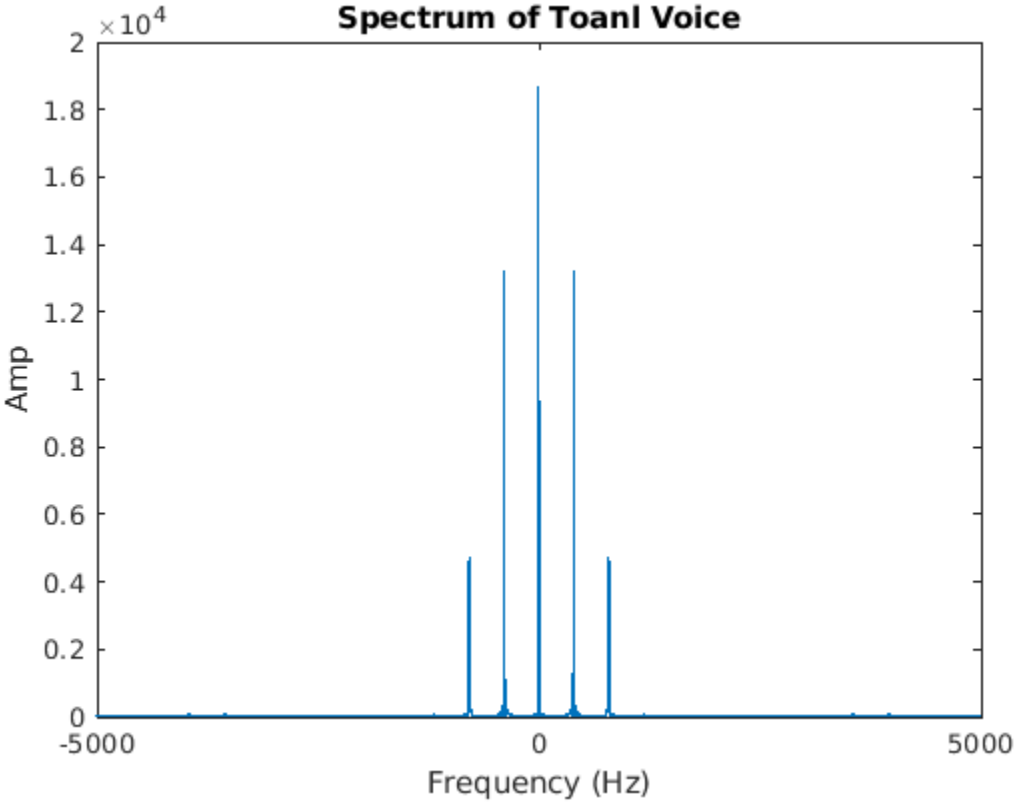

1Hz Sine Wave Spectrum after Time Compression by 3
Magnitude
2500
2000
1500
1000
500
0
-5
0
5
Frequency (Hz

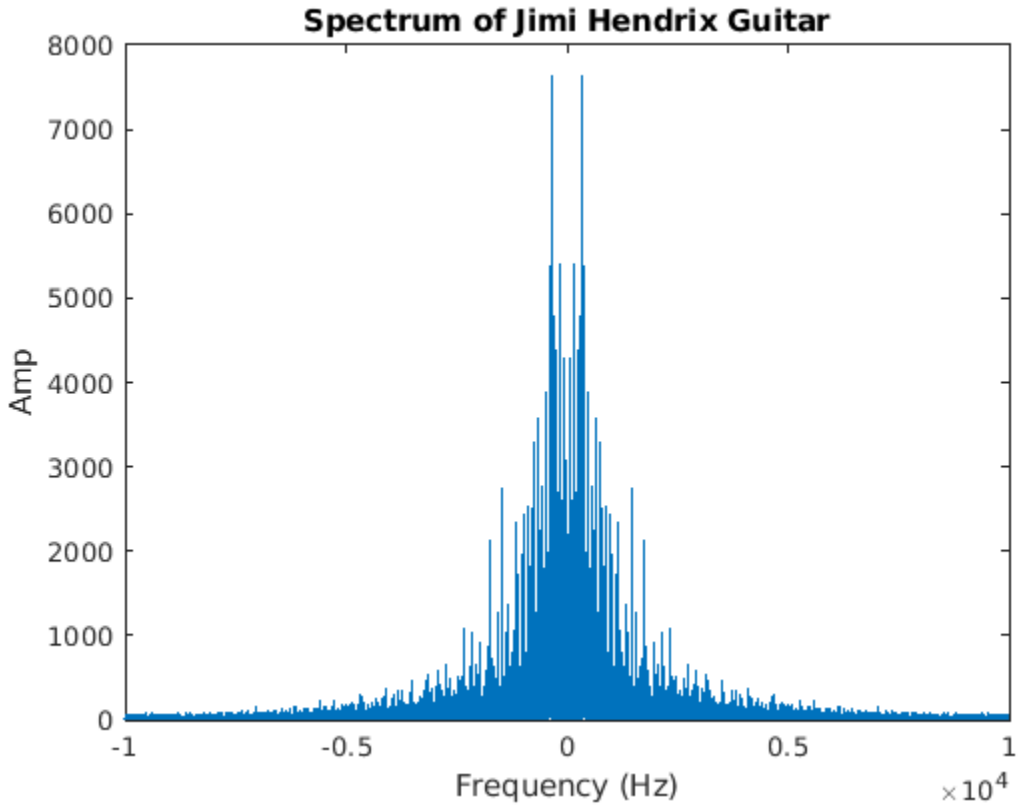

Rectangular Signal Spectrum
Magnitude
500
400
300
200
100
0
-150
-100
-50
0
50
100
150
Frequency (Hz


Rectangular Signal Spectrum after Time Compression by 3
Magnitude
200
150
100
50
0
-150
-100
-50
0
50
100
150
Frequency (Hz


Triangular Signal Spectrum
Magnitude
Frequency (Hz
250
200
150
100
50
0
-100
-50
0
50
100


Triangular Signal Spectrum after Time Compression by 3
Magnitude
Frequency (Hz
100
80
60
40
20
0
-100
-50
0
50
100


Spectrum of Toanl Voice
×10^4
Amp
Frequency (Hz)
2
1.8
1.6
1.4
1.2
1
0.8
0.6
0.4
0.2
0
-5000
0
5000



Published with MATLAB® R2023b