23 January 2021; Annapolis, Maryland

Lab 1 Part 2

Signal Analysis in MATLAB

MIDN McKenzie Eshleman
United States Naval Academy
Cyber Operations
Annapolis, Maryland
221938

Abstract— *For this lab we plotted a 440 Hz sinusoid in both time and frequency. Then we plotted MATLABS's chirp, gong, and splat for both time and frequency.*

Keywords—Frequency; Signals; Bandwidth

Introduction

We first begin our Lab 1 Part 2 to generate a 440 Hz sine wave. Then we use the provided function getSpectrum to determine the frequency content of the sine wave. The next procedure we do is display the spectrum of our signal. The next type of signal frequency we analyze is real-time voice signal analysis. Using the MATLAB included sounds of chirp, gong, and splat we plot the signals as a function of time.

I. Fourier Analysis

A. What is Fourier Analysis

Fourier Analysis is a concept that any signal is made up of components at various frequencies, and each component is sinusoidal. By adding enough signals together, the appropriate frequency, amplitude, and phase any electromagnetic signal can be constructed.

B. 440 Hz Sinusoid

To begin plotting out our 440 Hz sinusoid we input the initial data that is given to us in MATLAB. The next step was to display the 440 Hz Sinusoid signal as a function of time. Using the plot function the sinusoid is displayed. The findings relating to time were from zero to ten milliseconds. The amplitude ranges from negative one to one volts. The 440Hz sinusoid creates the bandwidth of two.

C. 440 Hz Sinusoid Spectrum

Using the getSpectrum function that was provided we can determine and display the spectrum of our signal. The frequency goes from zero to one thousand, and peeks its amplitude at 440 Hz, reaching a voltage of one. This makes the bandwidth for the spectrum one.

D. Spectrum of Signals with Multiple Frequencies

- For sound clip chirp, the time ranges from zero to one and a half seconds. The frequency domain is from zero to one, making the bandwidth one as well. When using the getSpectrum function the chirp sounds gives a frequency up to 4000 Hz, and the amplitude ranges from zero to 0.0175 with a peak at 2500 Hz..
- For sound gong, the time ranges from zero to five seconds. The frequency domain is from zero to one, making the bandwidth one as well. When using the getSpectrum function the gong sounds gives a frequency up to 4000 Hz, and the amplitude ranges from zero to 0.0175 with a peak at 1000 Hz.
- For sound splat, the time ranges from zero to one seconds. The frequency domain is from zero to one, making the bandwidth one as well. When using the getSpectrum function the splat sounds gives a frequency up to 4000 Hz, and the amplitude ranges from zero to 0.006 with a peak at 1000 Hz.

E. Appendix

```
fs = 10e5;            % "fs" is sampling frequency
f0 = 440;             % "f0" is the frequency (in Hz) of the sinusoid
A = 1;                % "A" is the amplitude of the sinusoid, 1.0 Volts
Ts = 1/fs;            % "Ts" is the time between samples
dur = 1/f0 * 1e1;     % "dur"ation for 1000 periods of f0
N = fs*dur;           % "N" is the number of samples of the sinusoid for duration
t = 0:Ts:(N-1)*Ts;    % "t" is the time vector (StartTime:StepSize:StopTime)
s1 = A.*sin(2*pi*f0*t) ; % "s1" is our signal vector (a 440 Hz sinusoid)
plot(t,s1,'m-','linewidth',2) % These plot options produce lines of different
% colors/styles, as well as a thicker line that's
% easier to see.
axis([0 10e-3 -1 1]) % Sets the amount of the waveform that's visible
% in the figure window so that you don't get a
% big blue (or pink, in this case) blob. Suggest
% setting xmax to 10 ?s or less.
xlabel("Time (ms)") % All figures should have meaningful titles
ylabel("Voltage (V)") % and axes labels.
title("440 Hz Sinusoid")
grid on;
%[freq, amp] = getSpectrum(A, fs);
```

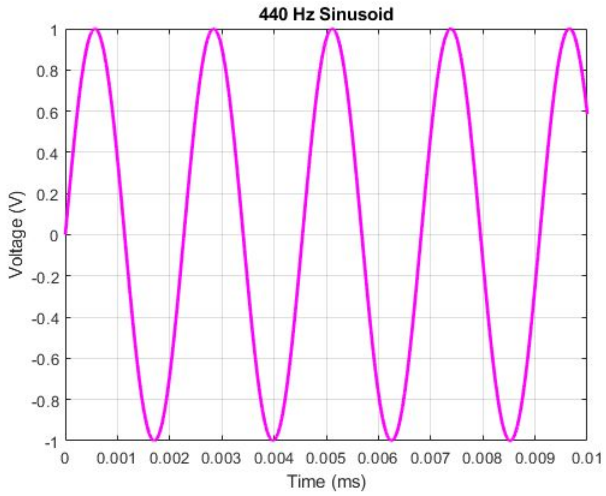

Fig. 1. *440 Hz Sinusoid*

```
fs = 10e5;            % "fs" is sampling frequency
f0 = 440;             % "f0" is the frequency (in Hz) of the sinusoid
A = 1;                % "A" is the amplitude of the sinusoid, 1.0 Volts
Ts = 1/fs;            % "Ts" is the time between samples
dur = 1/f0 * 1e1;     % "dur"ation for 1000 periods of f0
N = fs*dur;           % "N" is the number of samples of the sinusoid for duration
t = 0:Ts:(N-1)*Ts;  % "t" is the time vector (StartTime:StepSize:StopTime)
s1 = A.*sin(2*pi*f0*t); % "s1" is our signal vector (a 440 Hz sinusoid)
[freq, amp] = getSpectrum(s1, fs);
plot(freq,amp,'-m','linewidth',2) % These plot options produce lines of different
% colors/styles, as well as a thicker line that's
% easier to see.
axis([0 1000 0 1]) % Sets the amount of the waveform that's visible
% in the figure window so that you don't get a
% big blue (or pink, in this case) blob. Suggest
% setting xmax to 10 ?s or less.
xlabel("Time (seconds)") % All figures should have meaningful titles
ylabel("Voltage (V)") % and axes labels.
title("Spectrum fo a 440 Hz Sinusoid")
grid on;
```

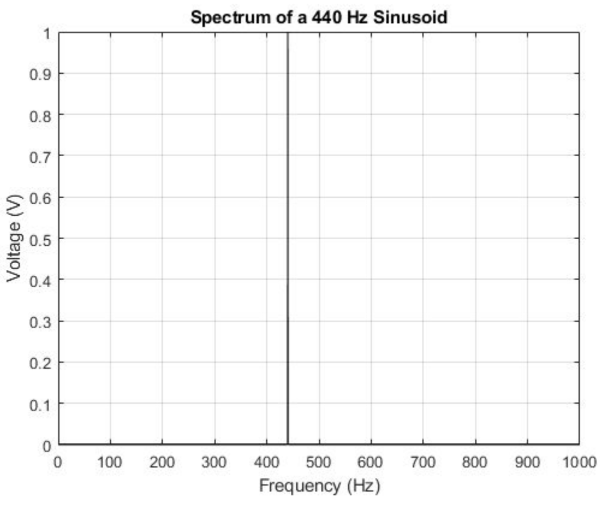

Fig. 2. Spectrum of a 440 Hz Sinusoid

```
fs = 10e5;            % "fs" is sampling frequency
f0 = 440;             % "f0" is the frequency (in Hz) of the sinusoid
A = 1;                % "A" is the amplitude of the sinusoid, 1.0 Volts
Ts = 1/fs;            % "Ts" is the time between samples
dur = 1/f0 * 1e1;     % "dur"ation for 1000 periods of f0
N = fs*dur;           % "N" is the number of samples of the sinusoid for duration
t = 0:Ts:(N-1)*Ts;  % "t" is the time vector (StartTime:StepSize:StopTime)
s1 = A.*sin(2*pi*f0*t);

%%Chirp
figure()
subplot(2,3,1);

load chirp;
Fs_chirp= 8192;
s1_chirp=y;
soundsc(s1_chirp,Fs_chirp);

sizechirp = length(s1_chirp);
xc = linspace(0,1.5,sizechirp);
plot(xc,s1_chirp,"m-","linewidth",1)
axis([0 1.5 -1 1]);
xlabel("Time (seconds)");
ylabel("Amplitude");
title("Signal 1: Chirp");
grid off;

%%Chirp Spectrum
subplot(2,3,4);
[freq, amp] = getSpectrum(s1_chirp, Fs_chirp);
plot(freq,amp, "m-", "linewidth",1);
axis([0 4000 0 0.02]);
xlabel("Frequency (Hz)");
ylabel("Amplitude");
title("Spectrum");

%%GONG
subplot(2,3,2);
load gong;
Fs_gong= 8192;
s1_gong=y;
soundsc(s1_gong,Fs_gong);

sizegong = length(s1_gong);
xg = linspace(0,5,sizegong);
plot(xg,s1_gong,"m-","linewidth",1)
axis([0 5 -1 1]);
xlabel("Time (seconds)");
ylabel("Amplitude");
title("Signal 1: Gong");
grid off;

%Gong Spectrum
subplot(2,3,5);
[freq, amp] = getSpectrum(s1_gong, Fs_gong);
```

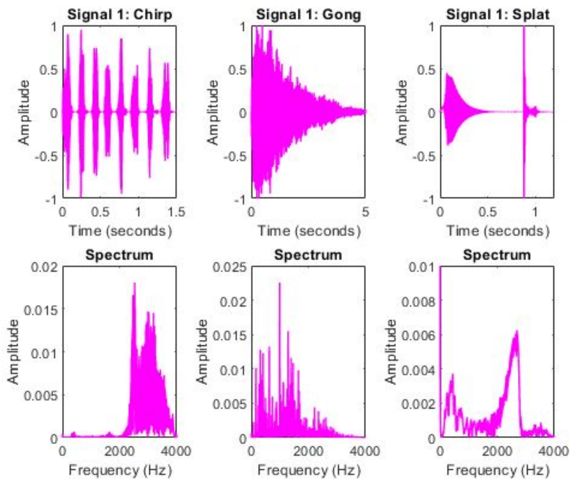

Fig. 3. Spectrum of Signals with Multiple Frequencies