**EECS 360**

**Lab 12**

**11/29/16**

**Chen Long**

1. **Objective**

This lab is about how to use MATLAB to find poles and zeros, and use this technique to evaluate the frequency response.

1. **Description**

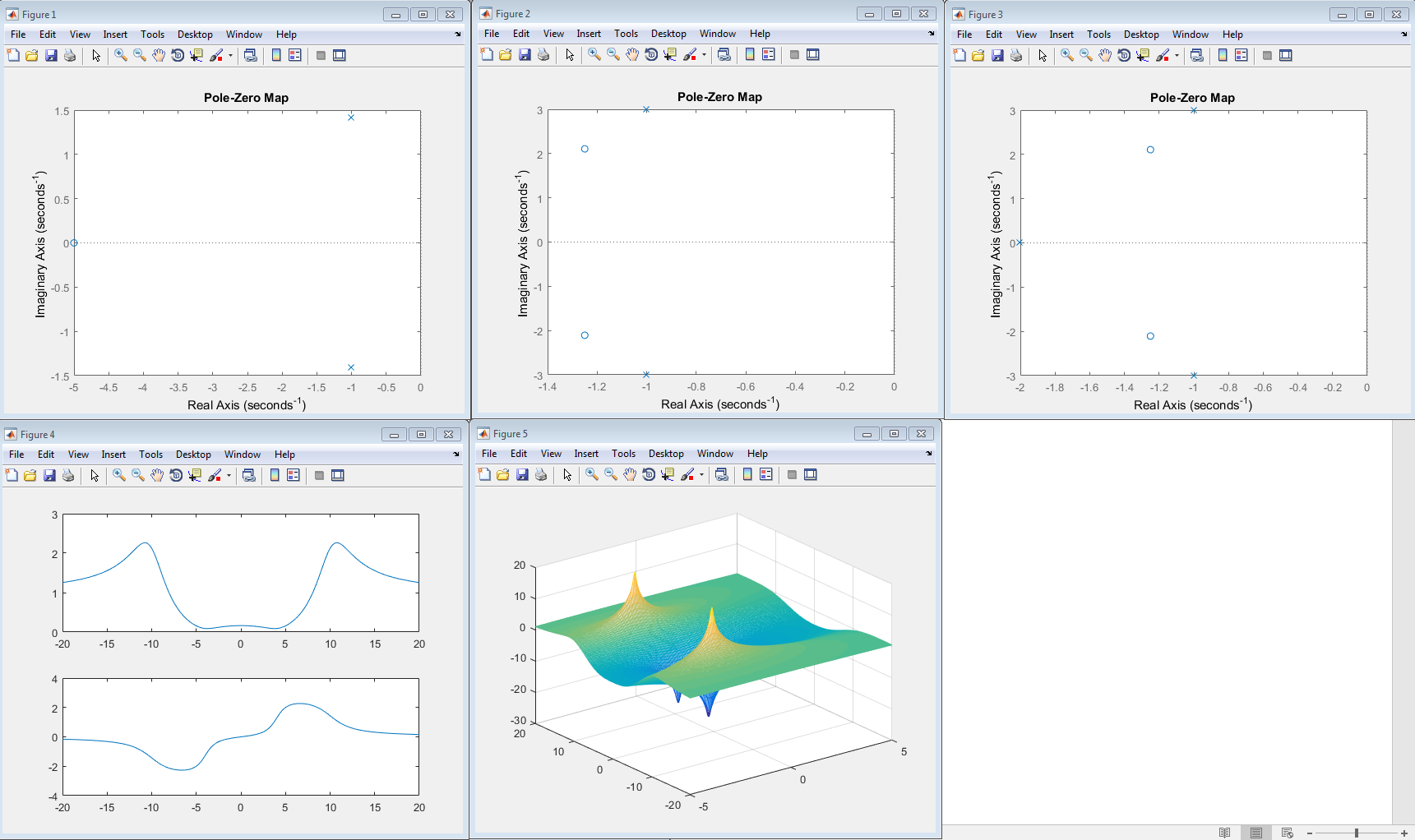
When we have the equation, we input the value of numerator and denominator into the MATLAB, the format is x = [a b c], after we input the value, we then use the command roots to calculate the root and plot the graph by using “pzmap”. The next part is to find the frequency response, the same way as we did before, we input the value, and we define the sample rate by using linspace, in this lab, we sample it from -20 to 20 with sample rate of 5rad/s; We then plot it with absolute value and angle. The last part is graphing a 3D graph, we use the command “mesh” to accomplish it.

1. **Result**

The result of the poles and zeros are the combination of real part and imaginary part. The meaning of poles are the roots of the denominator, and zeros mean the root of the nominator. Therefore, there is no surprise that some of the equations have no zeros because there is no root for the nominator. The graph of figure 5 shows the same result as figure 4 if we rotate the 3D graph, because figure 4 is the graph of the frequency, and figure 5 is actually the bode plot, which is really similar to the frequency plot but just do a log function.

1. **Result**

This lab gives us a new idea of using MATLAB, Laplace is another way of doing a Fourier Transform, we proved it by plotting the frequency graph. It also gives us a really useful tool to just simply calculate the root of any quadratic equation.

  
MATLAB CODE

clear all;

% part 1

figure(1);

a = [1 5];

b = [1 2 3];

xs = roots(b);

ys = roots(a);

pzmap(xs, ys);

% 2

figure(2);

a = [2 5 12];

b = [1 2 10];

xs = roots(b);

ys = roots(a);

pzmap(xs, ys);

% 3

figure(3);

a = [2 5 12];

b = [1 4 14 20];

xs = roots(b);

ys = roots(a);

pzmap(xs, ys);

% part 2

figure(4);

a = [1 2 17];

b = [1 4 104];

w = linspace(-20, 20, 200);

H = freqs(a, b, w);

subplot(211);

plot(w, abs(H));

subplot(212);

plot(w, angle(H));

figure(5);

sigma = linspace(-5, 5, 200);

[sigma, w] = meshgrid(sigma, w);

s = sigma + j\*w;

H1 = polyval(a, s)./polyval(b, s);

mesh(sigma, w, 10\*log10(abs(H1)));