EE387 – SIGNAL PROCESSING

LAB 3

WIMALASIRI KPGP

E/14/403

SEMESTER 6

01/01/2019

EE 387 – Signal Processing
Lab 3: System Functions and Frequency Response

1. Objectives

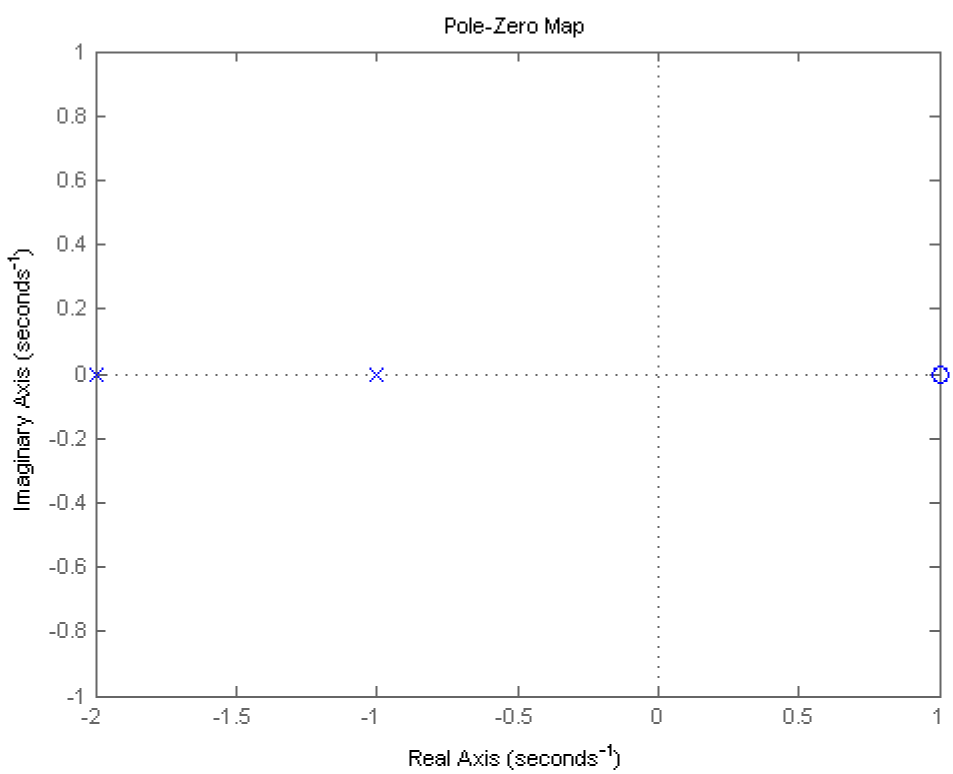
1. Learn MATLAB basics to use system functions.
2. Use MATLAB to determine the frequency response of a given LTI system.

2. Procedure

PART 1: Pole-Zero Diagrams in MATLAB.

Part 1 code

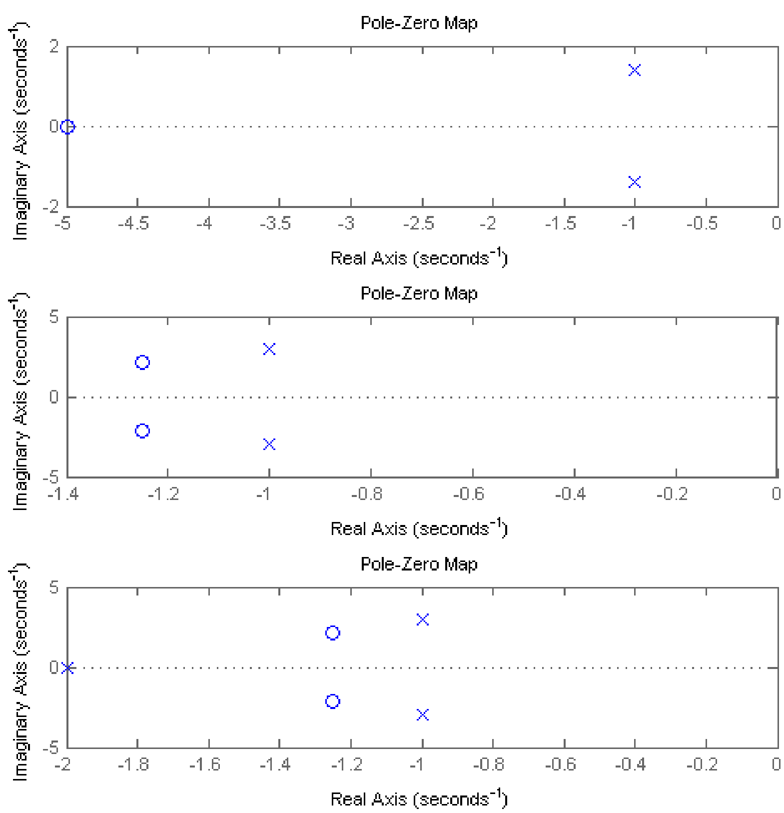
```
clear all;
close all;
b = [1 -1]; % Numerator coefficients
a = [1 3 2]; % Demoninator coefficients
zs = roots(b); % Generetes Zeros
ps = roots(a); % Generetes poles
pzmap(ps,zs); % generates pole-zero diagram
```



Exercise

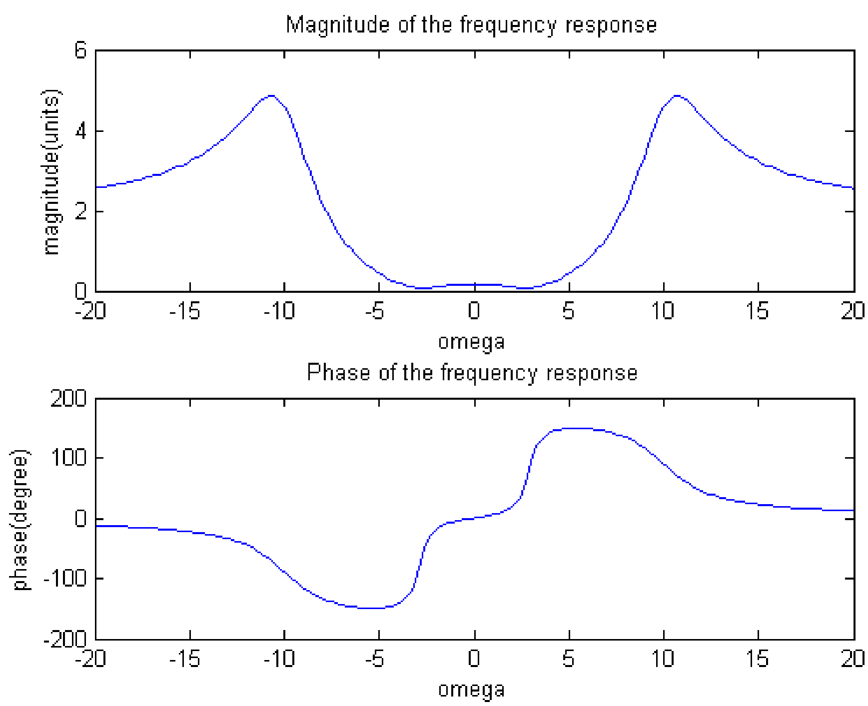
Part 1 Exercise code

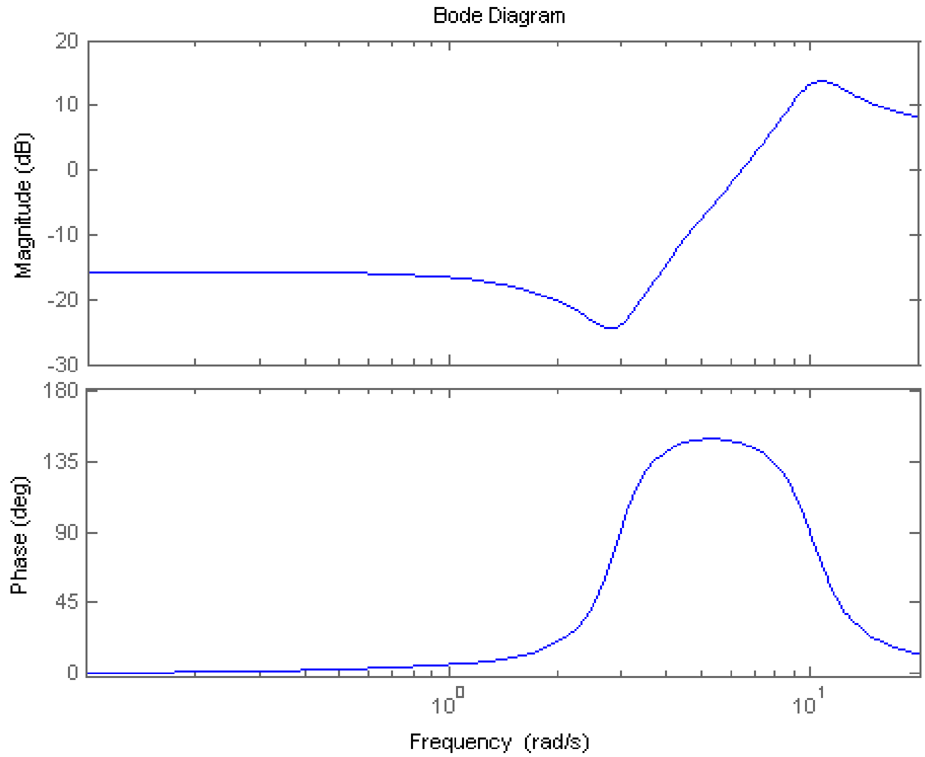
```
clear all;
close all;

b1 = [1 5]; % Numerator coefficients
a1 = [1 2 3]; % Demoninator coefficients
zs1 = roots(b1); % Generetes Zeros
ps1 = roots(a1); % Generetes poles
subplot(3,1,1)  % Subplotting as the 1st plot
pzmap(ps1,zs1); % generates pole-zero diagram

b2 = [2 5 12]; % Numerator coefficients
a2 = [1 2 10]; % Demoninator coefficients
zs2 = roots(b2); % Generetes Zeros
ps2 = roots(a2); % Generetes poles
subplot(3,1,2)  % Subplotting as the 2nd plot
pzmap(ps2,zs2); % generates pole-zero diagram

b3 = [2 5 12]; % Numerator coefficients
a3 = [1 4 14 20]; % Demoninator coefficients
zs3 = roots(b3); % Generetes Zeros
ps3 = roots(a3); % Generetes poles
subplot(3,1,3)  % Subplotting as the 3rd plot
pzmap(ps3,zs3); % generates pole-zero diagram
```



PART 2: Frequency Response and Bode Plots in MATLAB

```
Part 2 code

clear all;
close all;

b = [2 2 17];    % Numerator coefficients
a = [1 4 104];   % Demoninator coefficients
omega = linspace(-20,20,200); % taking 200 points between -20 and 20
H = freqs(b,a,omega);    %evaluate the frequency response of a Laplace
transform.

magnitude = abs(H); % taking the absolute value
phase = angle(H);    % taking the phase
phaseDegree = phase*180/pi; % converting phase from radians to degree

subplot(2,1,1);
plot(omega,magnitude);      % plotting the magnitude
title('Magnitude of the frequency response')
xlabel('omega')
ylabel('magnitude(units)')

subplot(2,1,2);
plot(omega, phaseDegree);    % plotting the phase
title('Phase of the frequency response')
xlabel('omega')
ylabel('phase(degree)')

figure
h = tf(b,a);     % taking denominator and the nominator for bode function
bode(h,omega)    % plotting the bode diagram
```

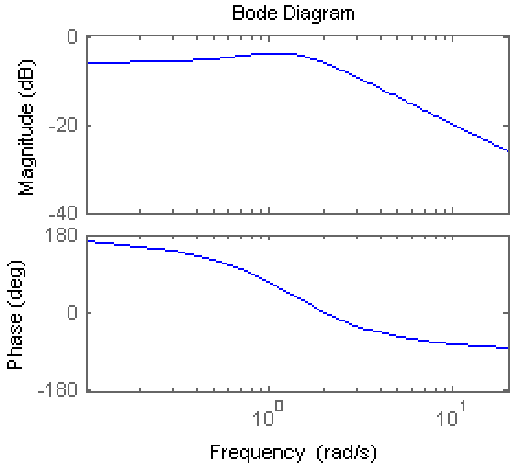


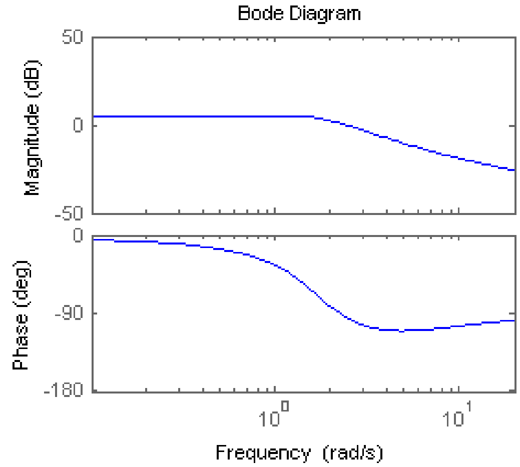

Bode Diagram
Magnitude (dB)
20
10
0
-10
-20
-30
Phase (deg)
180
135
90
45
0
10^0
10^1
Frequency (rad/s)

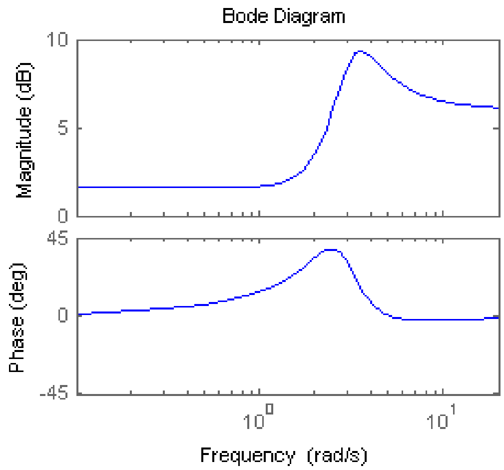
Exercise

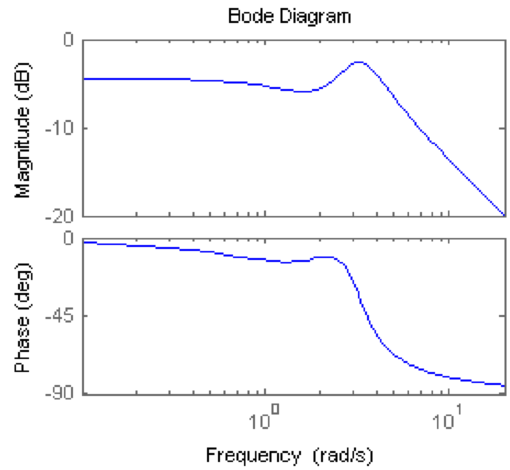
1.

```
Part 2 exercise 1 code

clear all;
close all;

omega = linspace(-20,20,200); % taking 200 points between -20 and 20

b0 = [1 -1]; % Numerator coefficients
a0 = [1 2 2]; % Demoninator coefficients

b1 = [1 5]; % Numerator coefficients
a1 = [1 2 3]; % Demoninator coefficients

b2 = [2 5 12]; % Numerator coefficients
a2 = [1 2 10]; % Demoninator coefficients

b3 = [2 5 12]; % Numerator coefficients
a3 = [1 4 14 20]; % Demoninator coefficients

h0 = tf(b0,a0); % taking denominator and the nominator for bode
function
subplot(2,2,1);
bode(h0,omega)

h1 = tf(b1,a1); % taking denominator and the nominator for bode
function
subplot(2,2,2);
bode(h1,omega)

h2 = tf(b2,a2); % taking denominator and the nominator for bode
function
subplot(2,2,3);
bode(h2,omega)

h3 = tf(b3,a3); % taking denominator and the nominator for bode
function
subplot(2,2,4);
bode(h3,omega)
```


Bode Diagram
Magnitude (dB)
Phase (deg)
Frequency (rad/s)


Bode Diagram
Magnitude (dB)
Phase (deg)
Frequency (rad/s)


Bode Diagram
Magnitude (dB)
Phase (deg)
Frequency (rad/s)


Bode Diagram
Magnitude (dB)
Phase (deg)
Frequency (rad/s)

Part 2 exercise 2 code

```
clear all;
close all;

syms t;
syms s; % make s and t symbolic

Hs0 = (s - 1)/(s.^2 + 2*s + 2);
Hs1 = (s + 5)/(s.^2 + 2*s + 3);
Hs2 = (2*s.^2 + 5*s + 12)/(s.^2 + 2*s + 10);
Hs3 = (2*s.^2 + 5*s + 12)/(s.^3 + 4*s.^2 + 14*s + 20);

x1 = sin(2*pi*(403*1*10^3)*t);
x2 = sin(2*pi*(403*2*10^3)*t);
x3 = sin(2*pi*(403*3*10^3)*t);  % 3 inputs

xs1 = laplace(x1);
xs2 = laplace(x2);
xs3 = laplace(x3);  % taking laplace of 3 inputs

fprintf('For system 0\n');
y1 = ilaplace(xs1*Hs0)
y2 = ilaplace(xs2*Hs0)
y3 = ilaplace(xs3*Hs0)  % multiply and taking inverse laplace transform

fprintf('For system 1\n');
y1 = ilaplace(xs1*Hs1)
y2 = ilaplace(xs2*Hs1)
y3 = ilaplace(xs3*Hs1) % multiply and taking inverse laplace transform

fprintf('For system 2\n');
y1 = ilaplace(xs1*Hs2)
y2 = ilaplace(xs2*Hs2)
y3 = ilaplace(xs3*Hs2) % multiply and taking inverse laplace transform

fprintf('For system 3\n');
y1 = ilaplace(xs1*Hs3)
y2 = ilaplace(xs2*Hs3)
y3 = ilaplace(xs3*Hs3) % multiply and taking inverse laplace transform
```

```
>> part2Exercise2

For system 0

y1 =
(4090821653175306420264367565163271763965375582044 16*sin((5437694194922373*t)/2147483648))/874297266029962144282139211060749075087149924261646770298427505 -
(34528221245734865668193265893344933923166262018100 6385152*cos((5437694194922373*t)/2147483648))/874297266029962144282139211060749075087149924261646770298427505 +
(34528221245734865668193265893344933923166262018100 6385152*exp(-t)*(cos(t) -
(5913703631499417187294784667806 6*sin(t))/2956851815747402750638178639951 3))/874297266029962144282139211060749075087149924261646770298427505

 y2 =

(2045410826587812717491677972486384338829921432371 2*sin((5437694194922373*t)/1073741824))/174859453205992428856427826261413865598439510006644630695550589 -
(3452822124575102136266994770141955040105945182297 3911040*cos((5437694194922373*t)/1073741824))/174859453205992428856427826261413865598439510006644630695550589 +
(3452822124575102136266994770141955040105945182297 3911040*exp(-t)*(cos(t) -
(1182740726299745086878404111924 2*sin(t))/5913703631497572512887413712645))/174859453205992428856427826261413865598439510006644630695550589

 y3 =

(5617888622830377978499970991274172531100929228 8*sin((1019567661547945*t)/134217728))/1080598122382003037286158742298348347198539098856541497455649 -
(1422518880669187205061076266100201891608001400694 37440*cos((1019567661547945*t)/134217728))/1080598122382003037286158742298348347198539098856541497455649 +
(1422518880669187205061076266100201891608001400694 37440*exp(-t)*(cos(t) -
(2079036432948725879424086410018*sin(t))/1039518216474272867719495795089))/1080598122382003037286158742298348347198539098856541497455649

 For system 1

 y1 =

 (1150940708193136773820258861876173144736931932310 08792576*exp(-t)*(cos(2^(1/2)*t) +
(5913703631496650175683728235064 2*2^(1/2)*sin(2^(1/2)*t))/29568518157524756052584487666457))/29143242200989647427953142287136906776595674994468776056883545 1 -
(1363607217724180542011044980049210615093330821775 36*sin((5437694194922373*t)/2147483648))/291432422009896474279531422871369067765956749944687760568835451 -
(1150940708193136773820258861876173144736931932310 08792576*cos((5437694194922373*t)/2147483648))/291432422009896474279531422871369067765956749944687760568835451

 y2 =

 (5754703540960971791545595448692435917632479301795 7351424*exp(-t)*(cos(2^(1/2)*t) +
(5913703631498033681489256451435 4*2^(1/2)*sin(2^(1/2)*t))/29568518157500544700987743879961))/291432422009964654640417625252329641090379845956407125708201339 -
(3409018044312445197021289823932382248323747571302 4*sin((5437694194922373*t)/1073741824))/291432422009964654640417625252329641090379845956407125708201339 -
```

(57547035409609717915455954486924359176324793017957351424*cos((5437694194922373*t)/1073741824))/29143242200996465464041762525232964109037984595640712570820133 9

y3 =

(14225188806694583730286719385546004719341373172482048 0*exp(-t)*(cos(2^(1/2)*t) + (20790364329486177930330295181 14*2^(1/2)*sin(2^(1/2)*t))/10395182164744710261031000969 13))/10805981223819655846953398743017263774728286025779062 33373729 - (56178886228299561043802150365293544282741800960*sin((1019567661547945*t)/134217728))/1080598122381965584695339874301726377472828602577906233373729 - (142251888066945837302867193855460047193413731724820480*cos((1019567661547945*t)/134217728))/1080598122381965584695339874301726377472828602577906233373729

For system 2

y1 =

(349718906411384870537057053492873943267099903618382828351995130*sin((5437694194922373*t)/2147483648))/174859453205556074546756172495179664811506930130420974550359677 - (690564424912327811507285319349288261705657402313932800 00*cos((5437694194922373*t)/2147483648))/174859453205556074546756172495179664811506930130420974550359677 + (69056442491232781150728531934928826170565740231393280000*exp(-t)*(cos(3*t) - (1774111089447888248060695897422 3*sin(3*t))/5913703631474514082795276773125))/174859453205556074546756172495179664811506930130420974550359677

y2 =

(1748594532059174304594529527329249963792961721779429203050508514*sin((5437694194922373*t)/1073741824))/8742972660294167013950491613431938541790223353344043183518577 77 - (17264110622860701294731348859974710095794301923686455705 6*cos((5437694194922373*t)/1073741824))/874297266029416701395049161343193854179022335334404318351857777 + (172641106228607012947313488599747100957943019236864557056*exp(-t)*(cos(3*t) - (8870555447245667016428356460781 9*sin(3*t))/29568518157462498291335717929753))/874297266029416701395049161343193854179022335334404318351857777

y3 =

(4322392489527188192146619910262825171889172330575976525713 54*sin((1019567661547945*t)/134217728))/216119624476340683311921563299682318852950365696839307414125 - (284503776133728973825176984238280946191145913517342 72*cos((1019567661547945*t)/134217728))/216119624476340683311921563299682318852950365696839307414125 + (28450377613372897382517698423828094619114591351734272*exp(-t)*(cos(3*t) - (31185546494227105167674304933 63*sin(3*t))/10395182164738765509522871914 41))/216119624476340683311921563299682318852950365696839307414125

For system 3

y1 =

(11677359366420320646856704*exp(-2*t))/29568518157510920994529205502745 -
(816758669479578715443618271320731198786976422803919820952683389926816411277759877218304*cos((5437694194922373*t)/2147483648))/10340669834241832018504042425298145562681788197842016806063484729663189556971549762374721626730 +
(483838137323507249862282877900717366232314773267554077168777781551846439421739008*sin((5437694194922373*t)/2147483648))/1034066983424183201850404242529814556268178819784201680606348472966318955697154976237472162673 +
(34528221245713324742224356815912111952642008944936275148 8*exp(-t)*(cos(3*t) -
(27670116110564327424*sin(3*t))/29568518157455580762308076847897))/87429726602778037273378086247589832405753465065210487275179838 5

y2 =

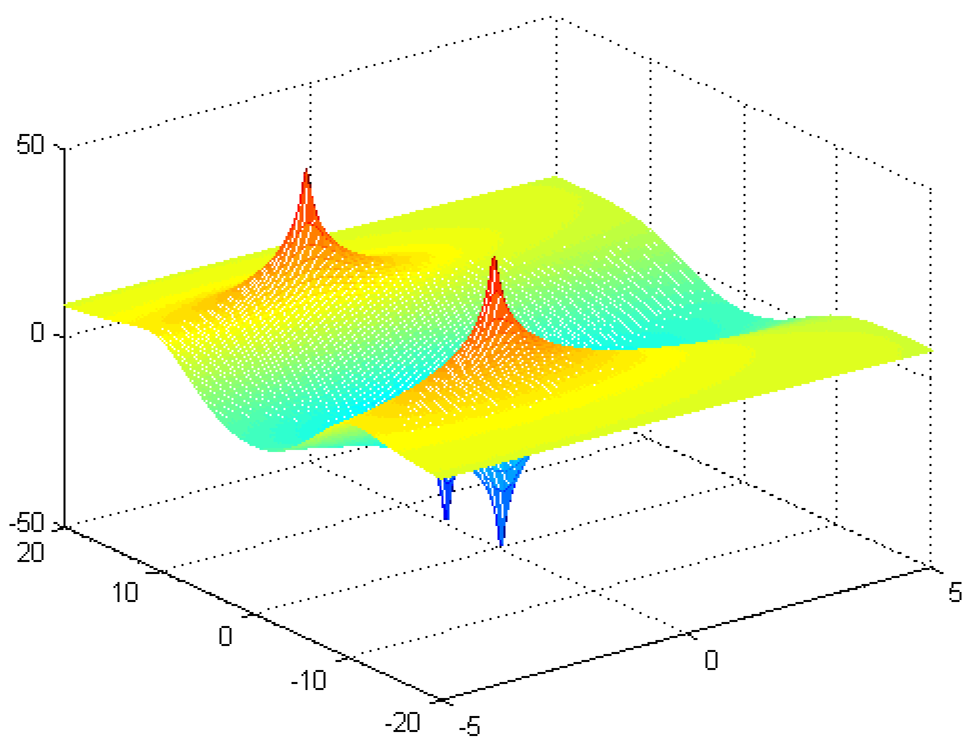
(5838679683210160323428352*exp(-2*t))/29568518157497085936473923339033 -
(1020948336850667644168223714832126373458575038981833159327246132151024957143437147111 4240*cos((5437694194922373*t)/1073741824))/258516745856408679065593818412549247489220071928368563617118500036016985026142910290345687096 41 +
(30239883582740426899542184051063778692435042997142017295426142593636619501342556 16*sin((5437694194922373*t)/1073741824))/258516745856408679065593818412549247489220071928368563617118500036016985026142910290345687096 41 +
(17264110622872818065588860216030672454214194277341410099 2*exp(-t)*(cos(3*t) -
(6917529027641081856*sin(3*t))/29568518157483250878418641175321))/87429726602941670139504916134319385417902233533440431835185777 7

y3 =

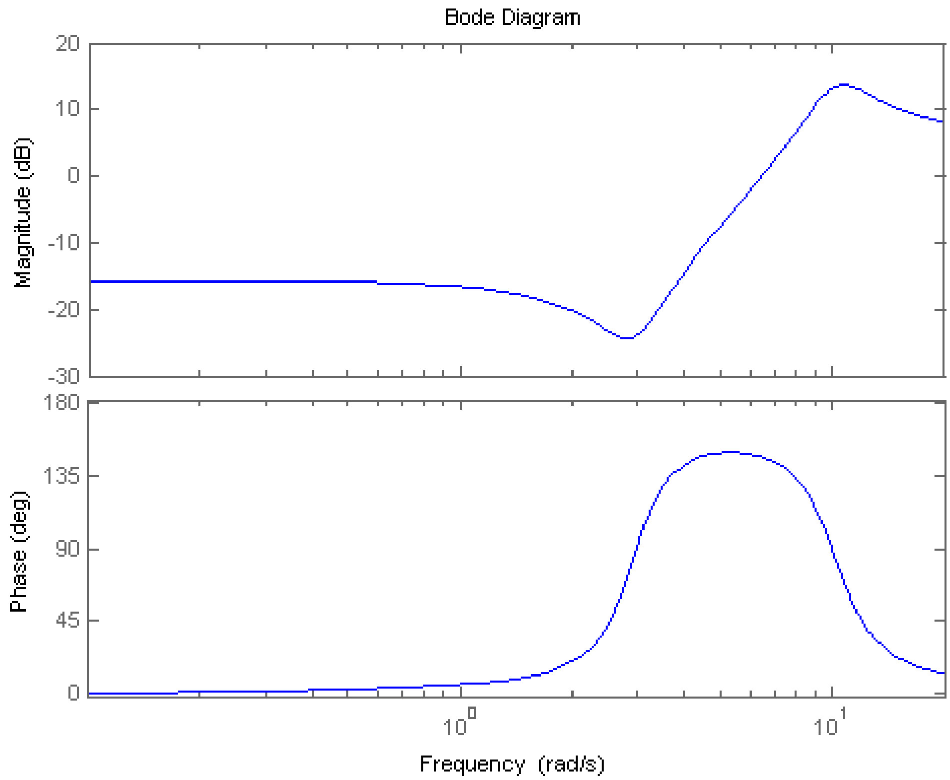
(1368440550752381409689 60*exp(-2*t))/10395182164744169829075716509 61 -
(59149371589366449074048461960999711708995154435396852419274068042777271826577883136*cos((1019567661547945*t)/134217728))/224660286580766421515812596784940289179410892291933156899690603544648558120107233981224125 +
(11679795123111217194630447719577341720229790680686041569267858485665754775552*sin((1019567661547945*t)/134217728))/224660286580766421515812596784940289179410892291933156899690603544648558120107233981224125 +
(28450377613381771970548102249608411793424494075314176*exp(-t)*(cos(3*t) -
(108086391056891904*sin(3*t))/10395182164742008101254578671 53))/21611962447634068331192156329968231885295036569683930741412 5

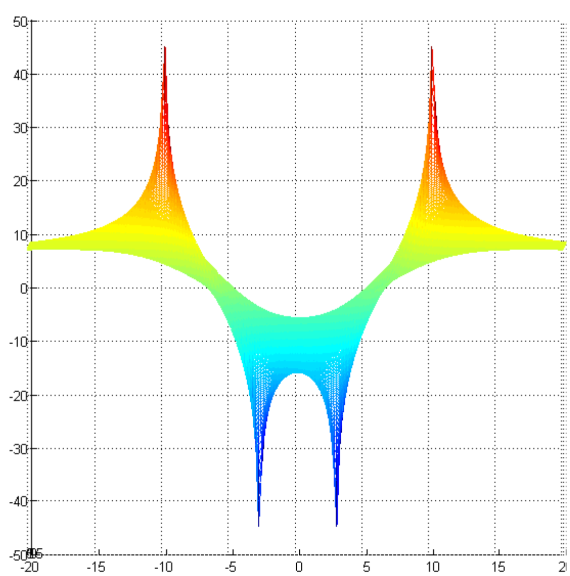
Part 3 code

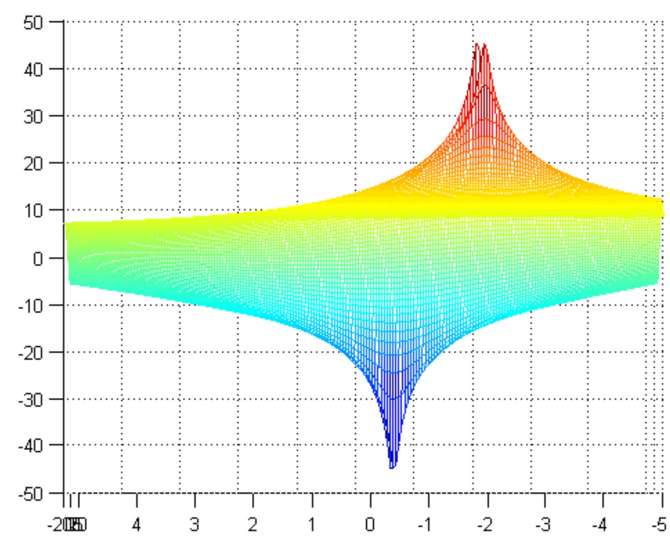
```
clear all;
close all;

omega = linspace(-20,20,200);
sigma = linspace(-5,5,200);
b = [2 2 17];    % Numerator coefficients from part 2
a = [1 4 104];   % Demoninator coefficients from part 2

[sigmagrid,omegagrid] = meshgrid(sigma,omega);

sgrid = sigmagrid+j*omegagrid;

H1 = polyval(b,sgrid)./polyval(a,sgrid);
mesh(sigma,omega,20*log10(abs(H1)));
```



Reddish upward pointed shape represents the poles and bluish downward pointed shapes represents zeros.







As the sigma and omega represents the real and imaginary parts respectively, surface diagram is an addition of phase and magnitude. However, surface diagram is symmetric around imaginary axis.