ELCE 705 Simulation Project 2

M-B5-5515-2 TANG QI

Problem 1: Linear Constant Coefficient Difference Equations

A causal system is given by:

y[n]-0.4y[n-1]+0.75y[n-2]+0.2y[n-3] = 2.2403x[n] +2.4908x[n-1]+2.2403x[n-2]

The impulse response of the causal discrete-time system can be calculated by matlab function impz, as following:

```
num=[2.2403 2.4908 2.2403];
den=[1 -0.4 0.75];
h =impz(num, den, N); % compute N samples of the impulse response
```

The following program can be used to calculate the output y[n]:

```
num=[2.2403 2.4908 2.2403];
den=[1 -0.4 0.75];
ic=[0 0 0]; %set zero initial conditions, number of zero is determined by the order of the difference equation
y=filter(num, den, x, ic); % x is the input sequence
```

a) **Calculate the output of the discrete-time system by using matlab function filter, when input x1[n] =cos(2*pi*0.1*n) for n=0~30. Show the input and output sequences in your report.**

Solution:

```
>> num=[2.2403,2.4908,2.2403,0];
den=[1,-0.4,0.75,0.2];
ic=[0,0,0];
n=0:30;
x=cos(2*pi*0.1*n) %input
y=filter(num, den, x, ic)%output
figure
stem(n,y);
title('P1a1-filter function');
```

x =

Columns 1 through 10

1.0000 0.8090 0.3090 -0.3090 -0.8090 -1.0000
-0.8090 -0.3090 0.3090 0.8090

Columns 11 through 20

1.0000 0.8090 0.3090 -0.3090 -0.8090 -1.0000
-0.8090 -0.3090 0.3090 0.8090

Columns 21 through 30

1.0000 0.8090 0.3090 -0.3090 -0.8090 -1.0000
-0.8090 -0.3090 0.3090 0.8090

Column 31

1.0000

y =

Columns 1 through 10

2.2403 5.1994 5.3472 -0.3188 -7.0677 -8.6051
-4.1932 1.2424 3.4730 3.1859

Columns 11 through 20

3.3688 4.3792 3.5356 -0.6541 -5.6790 -7.4358
-4.7000 -0.1150 3.0763 4.1466

Columns 21 through 30

4.3221 4.1193 2.5245 -1.0543 -5.0288 -6.6734
-4.8026 -0.8579 2.7036 4.5752

Column 31

4.9217

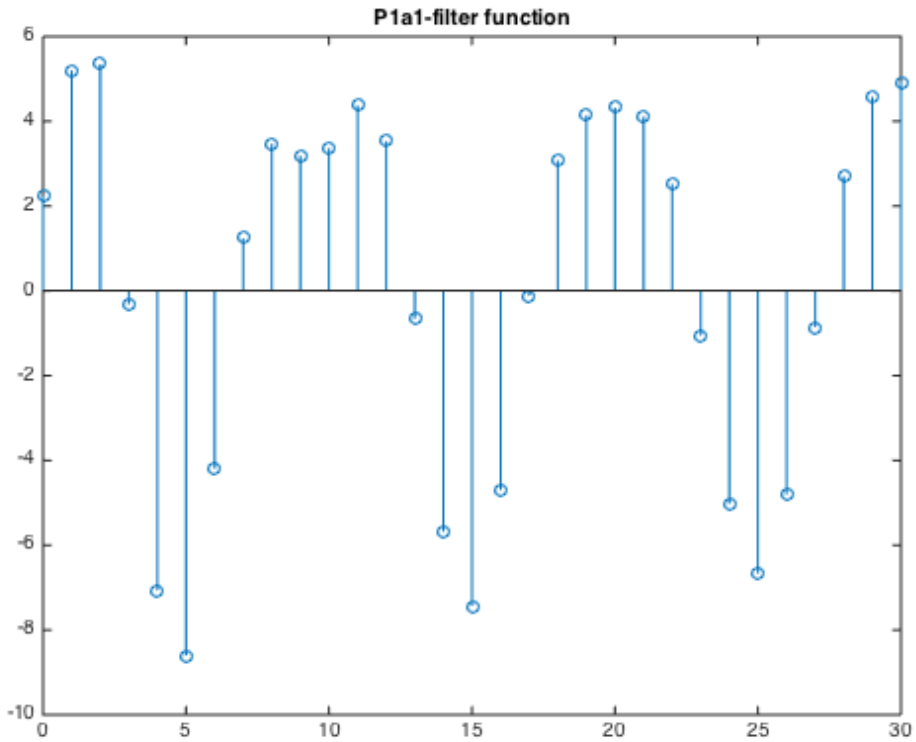

Fig.01 Filter function output of x1=cos(2*pi*0.1*n)

b) Calculate the impulse response h[n] with the length of 10, 20 and 30 samples respectively, by matlab function impz. Show them in the report and compare the results.

Solution:

```
num=[2.2403,2.4908,2.2403];
den=[1,-0.4,0.75,0.2];
for k=1:3
    n=0:1:(10*k);
    h=impz(num,den,n)
    figure
    stem(n,h);


end
```

For n=10:

h =

2.2403
3.3869
1.9148
-2.2223
-3.0024
0.0828
2.7294
1.6302
-1.4116
-2.3331
-0.2006

For n=20:

h =

2.2403
3.3869
1.9148
-2.2223
-3.0024
0.0828
2.7294
1.6302
-1.4116
-2.3331
-0.2006
1.9519
1.3978
-0.8647
-1.7846
-0.3449
1.3734
1.1650
-0.4951
-1.3465
-0.4003

For n=30:

h =

2.2403
3.3869
1.9148
-2.2223
-3.0024
0.0828
2.7294
1.6302
-1.4116
-2.3331
-0.2006
1.9519
1.3978
-0.8647
-1.7846
-0.3449
1.3734
1.1650
-0.4951
-1.3465
-0.4003
0.9488
0.9490
-0.2519
-1.0023
-0.4018
0.6414
0.7583
-0.0973
-0.7360
-0.3730

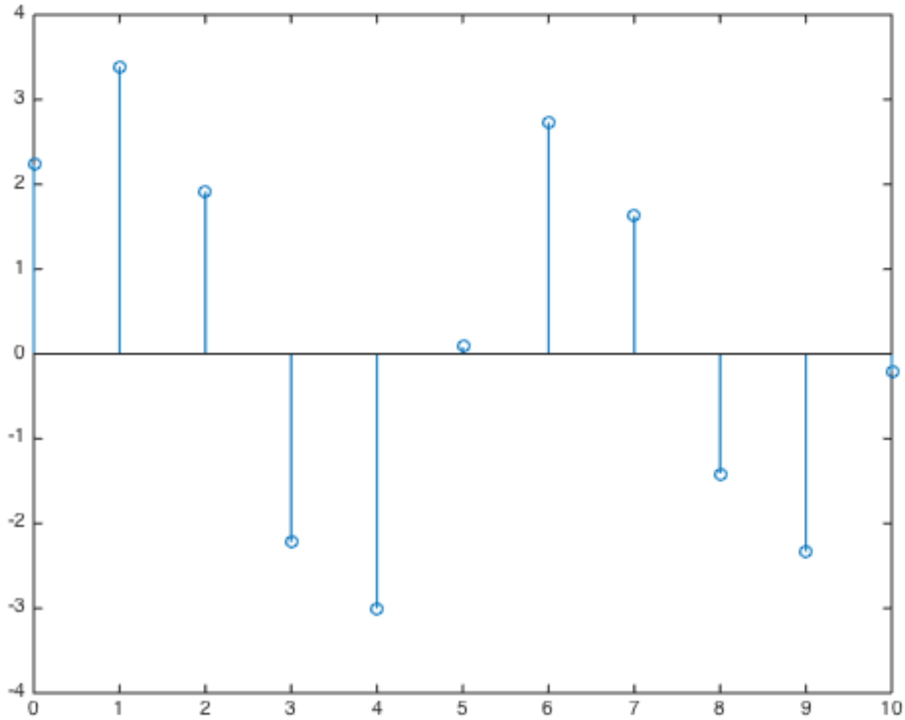

Fig.02 Impulse response h[n] in length of n=10

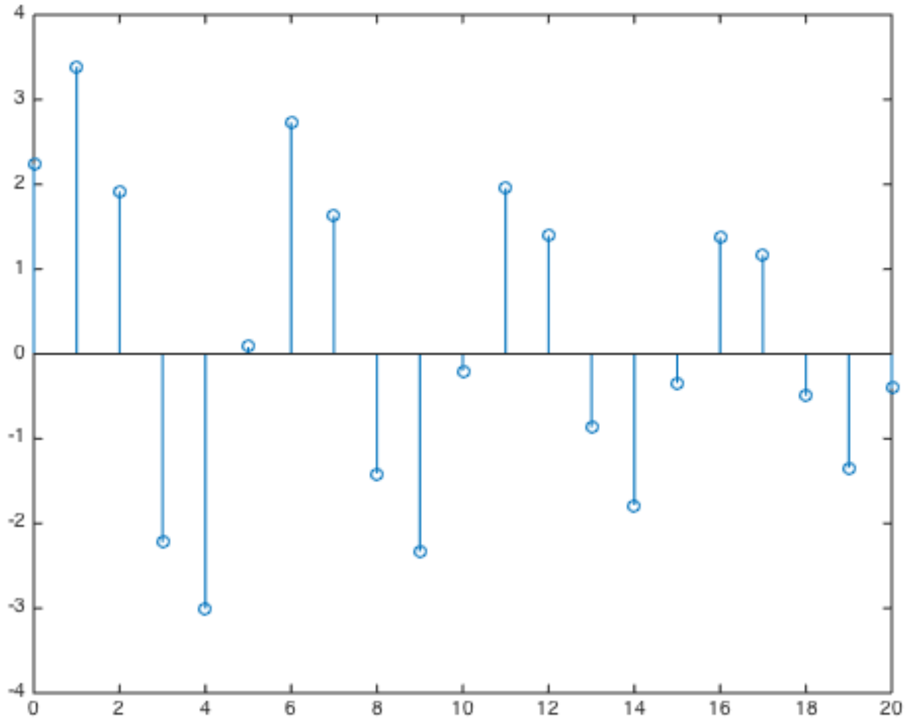

Fig.03 Impulse response h[n] in length of n=20

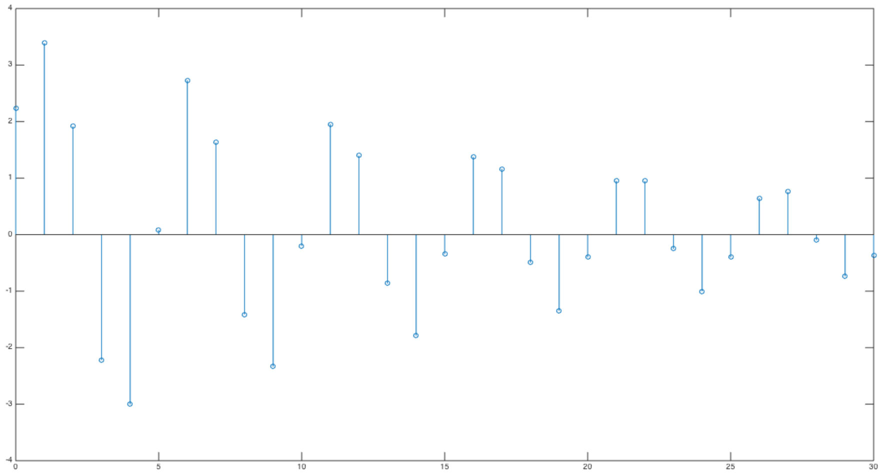

Fig.04 Impulse response h[n] in length of n=30

Comparing the three figures of impulse response of this system, we can find that the longer the sample lengths are, the more clear we can observe the trend of the impulse, which is, getting more stable while oscillating. In addition, because the sampling rate remains the same, which is 1, we can find that the density of the output also remains the same. Thus, we find data of Fig.02 on Fig.03 and data of Fig.02 & Fig.03 in Fig.04.

c) **Using matlab function conv to calculate the convolution sum of x1[n] and the h[n] to get the output of the given system. Use the three h[n] obtained in part (b) respectively. Show the results in your report and compare them. Comparison with the result obtained in part (a) is also necessary. Comments on your result.**

Solution:

For N=10:

```
>> num=[0.9,-0.45,0.35,0.002];
den=[1,0,0,0];
n=0:30;
x1=cos(2*pi*0.1*n)
ic=[0 0 0];
m1=0:10;
```

```
h=impz(num,den,m1);
Con=conv(x1,h)
r=[0:40];
figure
stem(r,Con);
title('p1c1-convolution');
```

```
x1 =
   Columns 1 through 10
      1.0000      0.8090      0.3090     -0.3090     -0.8090     -1.0000     -0.8090
-0.3090      0.3090      0.8090
   Columns 11 through 20
      1.0000      0.8090      0.3090     -0.3090     -0.8090     -1.0000     -0.8090
-0.3090      0.3090      0.8090
   Columns 21 through 30
      1.0000      0.8090      0.3090     -0.3090     -0.8090     -1.0000     -0.8090
-0.3090      0.3090      0.8090
   Column 31
      1.0000
Con =
   Columns 1 through 10
      0.9000      0.2781      0.2641     -0.1320     -0.4793     -0.6435     -0.5619
-0.2657      0.1320      0.4793
   Columns 11 through 20
      0.6435      0.5619      0.2657     -0.1320     -0.4793     -0.6435     -0.5619
-0.2657      0.1320      0.4793
   Columns 21 through 30
      0.6435      0.5619      0.2657     -0.1320     -0.4793     -0.6435     -0.5619
-0.2657      0.1320      0.4793
   Columns 31 through 40
      0.6435     -0.1662      0.3516      0.0020           0           0
0           0           0           0
   Column 41
           0
```

For N=20:

```
>> num=[0.9,-0.45,0.35,0.002];
den=[1,0,0,0];
n=0:30;
x1=cos(2*pi*0.1*n)
ic=[0 0 0];
m1=0:20;
h=impz(num,den,m1);
Con=conv(x1,h)
r=[0:50];
figure
stem(r,Con);
title('p1c1-convolution-length 20');
```

```
x1 =
   Columns 1 through 10
      1.0000    0.8090    0.3090   -0.3090   -0.8090   -1.0000   -0.8090
-0.3090    0.3090    0.8090
   Columns 11 through 20
      1.0000    0.8090    0.3090   -0.3090   -0.8090   -1.0000   -0.8090
-0.3090    0.3090    0.8090
   Columns 21 through 30
      1.0000    0.8090    0.3090   -0.3090   -0.8090   -1.0000   -0.8090
-0.3090    0.3090    0.8090
   Column 31
      1.0000
Con =
   Columns 1 through 10
      0.9000    0.2781    0.2641   -0.1320   -0.4793   -0.6435   -0.5619
-0.2657    0.1320    0.4793
   Columns 11 through 20
      0.6435    0.5619    0.2657   -0.1320   -0.4793   -0.6435   -0.5619
-0.2657    0.1320    0.4793
```

Columns 21 through 30

0.6435 0.5619 0.2657 -0.1320 -0.4793 -0.6435 -0.5619 -0.2657 0.1320 0.4793

Columns 31 through 40

0.6435 -0.1662 0.3516 0.0020 0 0 0 0 0 0

Columns 41 through 50

0 0 0 0 0 0 0 0 0 0

Column 51

0

For N=30:

```
>> num=[0.9,-0.45,0.35,0.002];
den=[1,0,0,0];
n=0:30;
x1=cos(2*pi*0.1*n)
ic=[0 0 0];
m1=0:30;
h=impz(num,den,m1);
Con=conv(x1,h)
r=[0:60];
figure
stem(r,Con);
title('p1c1-convolution-length 30');
```

x1 =

Columns 1 through 10

1.0000 0.8090 0.3090 -0.3090 -0.8090 -1.0000 -0.8090 -0.3090 0.3090 0.8090

Columns 11 through 20

1.0000 0.8090 0.3090 -0.3090 -0.8090 -1.0000 -0.8090 -0.3090 0.3090 0.8090

Columns 21 through 30

1.0000 0.8090 0.3090 -0.3090 -0.8090 -1.0000 -0.8090
-0.3090 0.3090 0.8090

Column 31

1.0000

Con =

0.9000	0.5619	0
0.2781	0.2657	0
0.2641	-0.1320	0
-0.1320	-0.4793	0
-0.4793	-0.6435	0
-0.6435	-0.5619	0
-0.5619	-0.2657	0
-0.2657	0.1320	0
0.1320	0.4793	0
0.4793	0.6435	0
0.6435	-0.1662	0
0.5619	0.3516	0
0.2657	0.0020	0
-0.1320	0	0
-0.4793	0	0
-0.6435	0	0
-0.5619	0	0
-0.2657	0	0
0.1320	0	0
0.4793	0	
0.6435	0	

The corresponding figures of convolution in lengths of 10, 20, 30 are as shown:

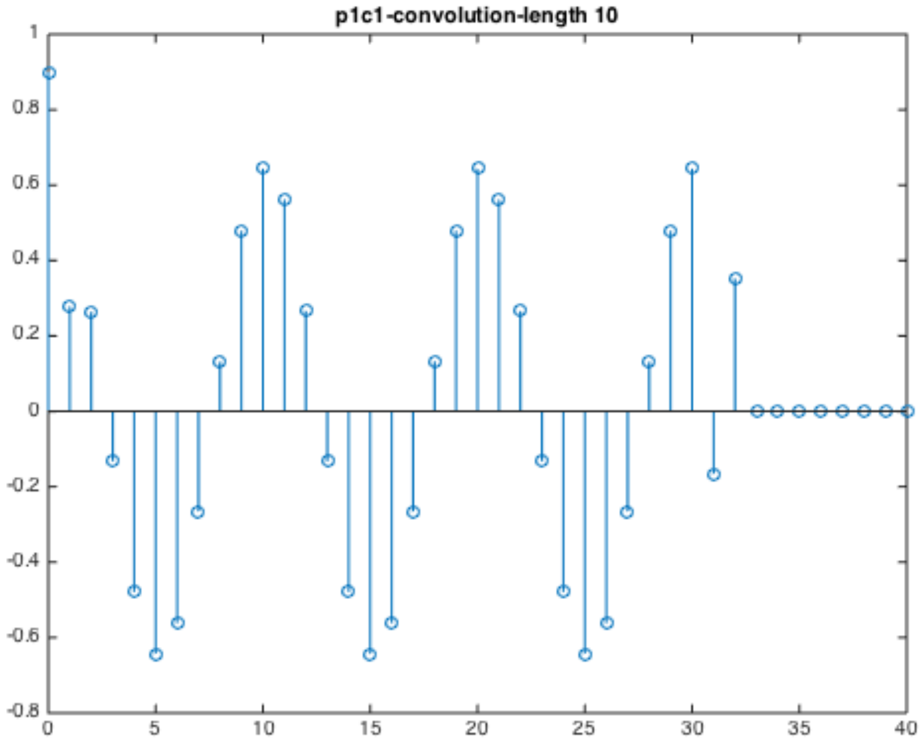

Fig.06 Convolution of x1(n) and h(n) in length of 10

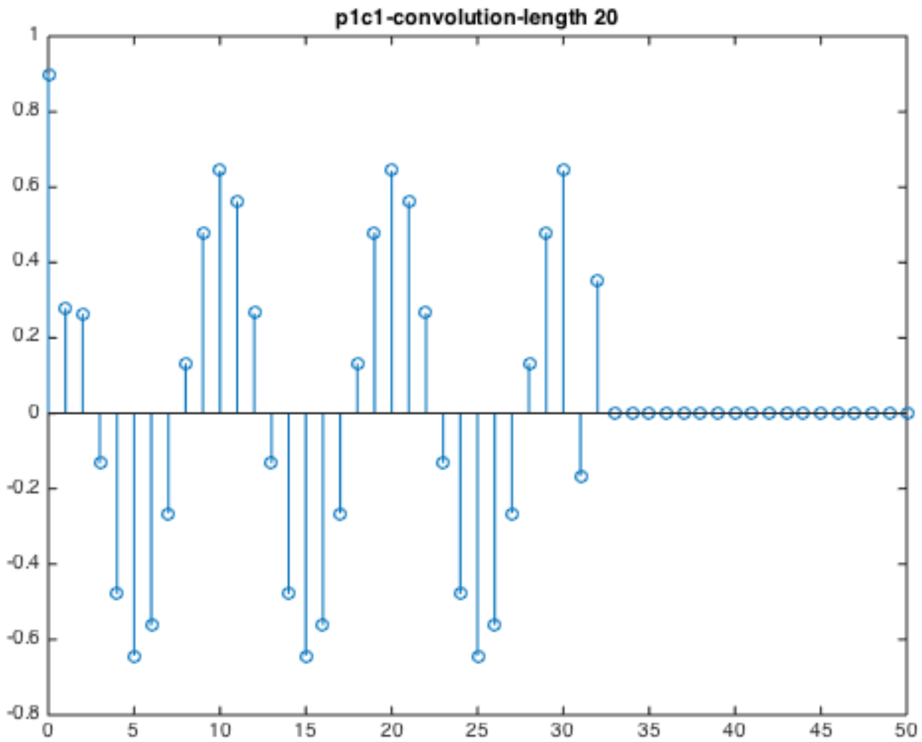

Fig.07 Convolution of x1(n) and h(n) in length of 20

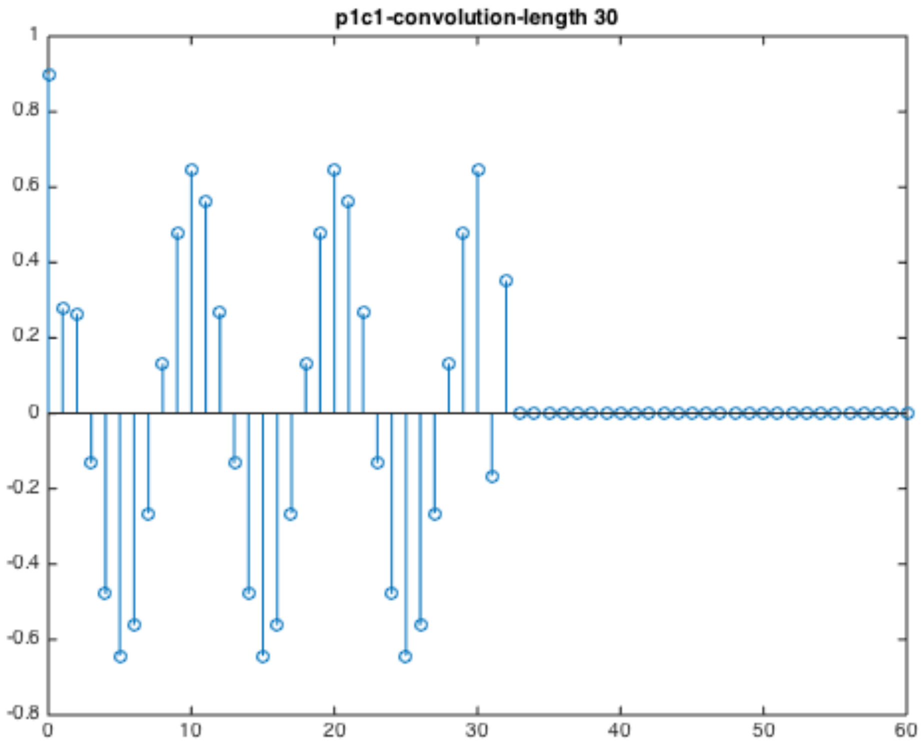

Fig.08 Convolution of x1(n) and h(n) in length of 30

Comparing with results from part a, we can find that data of the filter function and convolution from 0 to 30 are exactly the same. However, filter can handle FIR and IIR systems, while conv takes two inputs and returns their convolution. So here they give the same result.

Problem 2： FIR System

A causal system is given by: y[n] = 0.9x[n] -0.45x[n-1]+0.35x[n-2]+0.002x[n-3]

(a) Calculate the output of the discrete-time system by using matlab function filter, when input x1[n] =cos(2*pi*0.1*n) for n=0~30. Show the input and output sequences in your report.

```
>> num=[0.9,-0.45,0.35,0.002];
den=[1,0,0,0];
n=0:30;
x1=cos(2*pi*0.1*n)%input
ic=[0 0 0];
```

```
y=filter(num,den,x1,ic)%output
stem(n,y);
title('P2a1');
```

```
x1 =
  Columns 1 through 10
    1.0000    0.8090    0.3090   -0.3090   -0.8090   -1.0000   -0.8090
-0.3090    0.3090    0.8090
  Columns 11 through 20
    1.0000    0.8090    0.3090   -0.3090   -0.8090   -1.0000   -0.8090
-0.3090    0.3090    0.8090
  Columns 21 through 30
    1.0000    0.8090    0.3090   -0.3090   -0.8090   -1.0000   -0.8090
-0.3090    0.3090    0.8090
  Column 31
    1.0000
y =
  Columns 1 through 10
    0.9000    0.2781    0.2641   -0.1320   -0.4793   -0.6435   -0.5619
-0.2657    0.1320    0.4793
  Columns 11 through 20
    0.6435    0.5619    0.2657   -0.1320   -0.4793   -0.6435   -0.5619
-0.2657    0.1320    0.4793
  Columns 21 through 30
    0.6435    0.5619    0.2657   -0.1320   -0.4793   -0.6435   -0.5619
-0.2657    0.1320    0.4793
  Column 31
    0.6435
```

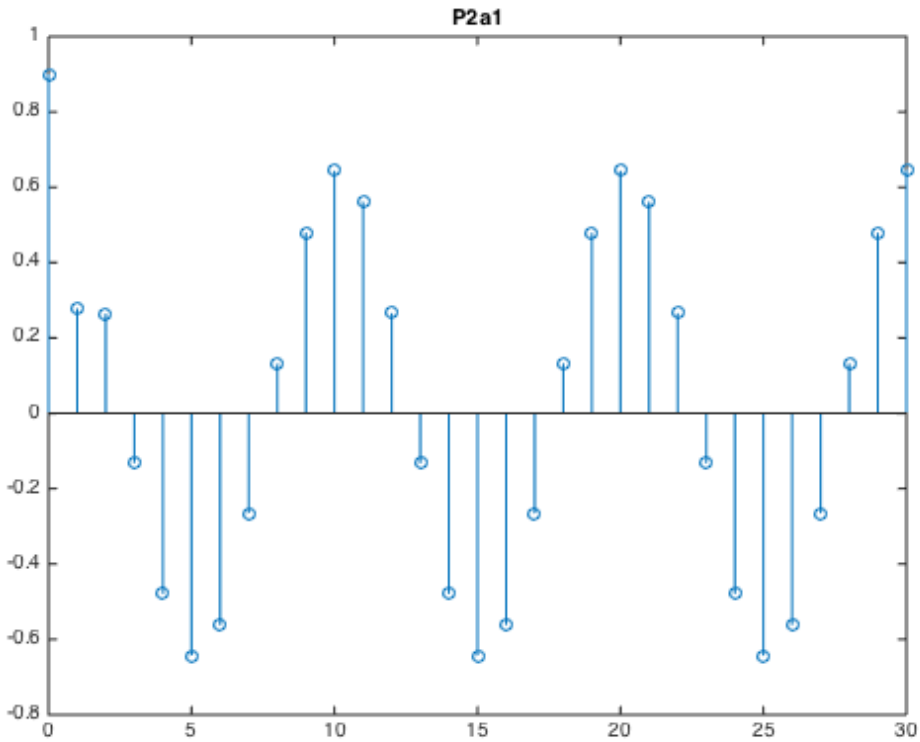

Fig.08 Filter function output of x1=cos(2*pi*0.1*n)

(b) Generate the impulse response h[n] of the given causal LTI system and provide the results in your report. In this case, the parameter N is not necessary for using the function impz. Could you explain why? Is the given system a stable system and why? Please provide your answer in the report.

Solution:

```
>> num=[0.9,-0.45,0.35,0.002];
den=[1,0,0,0];
n=0:30;
x1=cos(2*pi*0.1*n)%input
ic=[0 0 0];
h1=impz(num,den,n)
h2=impz(num,den)
```

x1 =

Columns 1 through 10

1.0000 0.8090 0.3090 -0.3090 -0.8090 -1.0000 -0.8090 -0.3090 0.3090 0.8090

Columns 11 through 20

1.0000 0.8090 0.3090 -0.3090 -0.8090 -1.0000 -0.8090 -0.3090 0.3090 0.8090

Columns 21 through 30

1.0000 0.8090 0.3090 -0.3090 -0.8090 -1.0000 -0.8090 -0.3090 0.3090 0.8090

Column 31

1.0000

h1 =

0.9000	0	0
-0.4500	0	0
0.3500	0	0
0.0020	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	
0	0	

h2 =

0.9000

-0.4500

0.3500

0.0020

Comparing values of h1 & h2 generated from filter function with & with out parameter N, we can find h1 and h2 are exactly the same when the system is still oscillating, From h1, we can tell that the system will finally reach the state of stable, which is exact 0, thus, this system is a finite impulse response system.

c) **Using matlab function conv to calculate the convolution sum of x1[n] and h[n],**

compare the results with that obtained in part (a).

```
>> num=[0.9,-0.45,0.35,0.002];
den=[1,0,0,0];
n=0:30;
x1=cos(2*pi*0.1*n);%input
ic=[0 0 0];
h1=impz(num,den,n)
%h=impz(num,den);
Con=conv(x1,h1)
r=0:(30+30);
stem(r,Con);
```

```
h1 =

    0.9000
   -0.4500
    0.3500
    0.0020
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0
         0

Con =

    0.9000
    0.2781
    0.2641
   -0.1320
   -0.4793
   -0.6435
   -0.5619
   -0.2657
    0.1320
    0.4793
    0.6435
    0.5619
    0.2657
   -0.1320
   -0.4793
   -0.6435
   -0.5619
   -0.2657
    0.1320
    0.4793
    0.6435
    0.5619
    0.2657
   -0.1320
   -0.4793
   -0.6435
   -0.5619
   -0.2657
    0.1320
    0.4793
```

0.6435	0	0
-0.1662	0	0
0.3516	0	0
0.0020	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	0
0	0	
0	0	

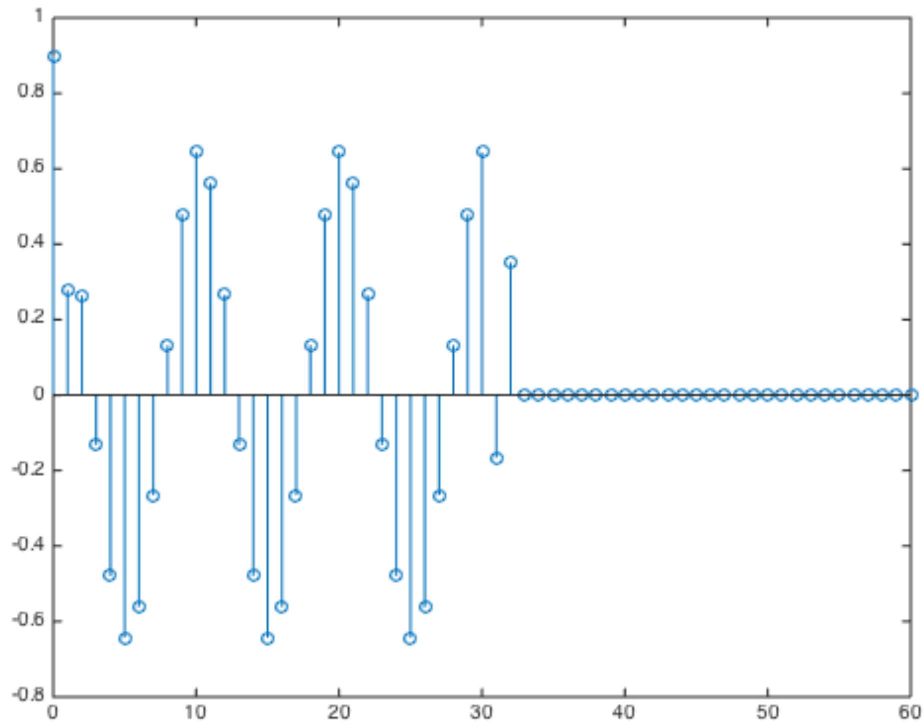

Fig.09 Convolution of x1(n) and h(n)

Comparing with the filter function in part a when n=60, we can find the results of filter function and convolution are the same. We assume that, in FIR system, filter can be considered equivalent to convolution.

Problem 3: DTFT Computation

The DTFT of a sequence x[n] of the form of Eq. (1) can be computed easily by using the Matlab function freqz.

$$X(e^{j\omega}) = \frac{b_0 + b_1 e^{-j\omega} + \cdots + b_M e^{-j\omega M}}{a_0 + a_1 e^{-j\omega} + \cdots + a_N e^{-j\omega N}} \quad (1)$$

There are several methods to use the function freqz, one example is given as follows:

```
w= - pi:0.01 : pi; % Assign the frequency spectrum range and step size
h=freqz(num,den,w);
```

(a) Using Matlab function freqz to obtain the discrete-time Fourier transform of Eq. 1. Show its real part, imaginary part, magnitude response and phase response in the range -2pi< ω< 2pi in the report.

$$X\left(e^{j\omega}\right) = \frac{0.0181 + 0.0543e^{-j\omega} + 0.0543e^{-j2\omega} + 0.0181e^{-j3\omega}}{1 - 1.76e^{-j\omega} + 1.1829e^{-j2\omega} - 0.2781e^{-j3\omega}} \quad (2)$$

Is the DTFT a periodic function of ω ? If it is, what is the period? Explain the type of symmetries exhibited by the four figures.

Solution:

```
>> num=[0.0181,0.0543,0.0543,0.0181];
den=[1,-1.76,1.1829,-0.2781];
w=-2*pi:0.01:2*pi;
h=freqz(num,den,w);
subplot(2,2,1);
plot(w/pi,real(h));grid
title('Real');
subplot(2,2,2);
plot(w/pi,imag(h));grid
title('Imaginary');
```

```
subplot(2,2,3);
plot(w/pi,abs(h));grid
title('Magnitude Response');
subplot(2,2,4);
plot(w/pi,angle(h));grid
title('Phase');
```

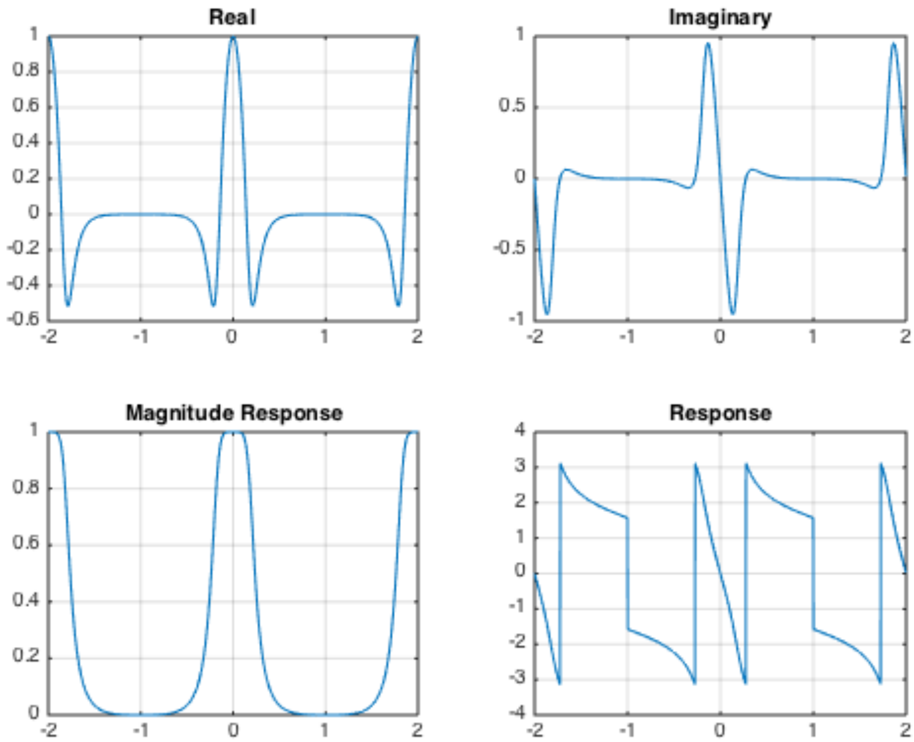

Fig.10 Real part, imaginary part, magnitude response and phase response of DTFT X

Yes, the DTFT is a periodic function of w. In this system X, we can observe the period is half of the domain, which is 2*pi. Type of symmetries of real part, imaginary part, magnitude response and phase response are correspondingly axial symmetry (to y axis), axial symmetry (to y axis), central symmetry (to origin), central symmetry (to origin).

(b) Modify your program to evaluate the DTFT of the following finite-length sequence:

g[n]= [0.1170 0.4132 0.7500 0.9698 0.9698 0.7500 0.4132 0.1170];

Draw its real part, imaginary part, magnitude response and phase response in the range $-\pi \leqslant \omega \leqslant \pi$. Is there any difference in the phase response of (b) and (a)?

```
>> num=[0.1170,0.4132,0.7500,0.9698,0.9698,0.7500,0.4132,0.1170];
den=[1];w=-pi:0.01:pi; h=freqz(num,den,w);
figure
subplot(2,2,1);plot(w/pi,real(h));grid
title('Real');
subplot(2,2,2); plot(w/pi,imag(h));grid
title('Imaginary');
subplot(2,2,3); plot(w/pi,abs(h));grid
title('Magnitude Response');
subplot(2,2,4); plot(w/pi,angle(h));grid
title('Phase');
```

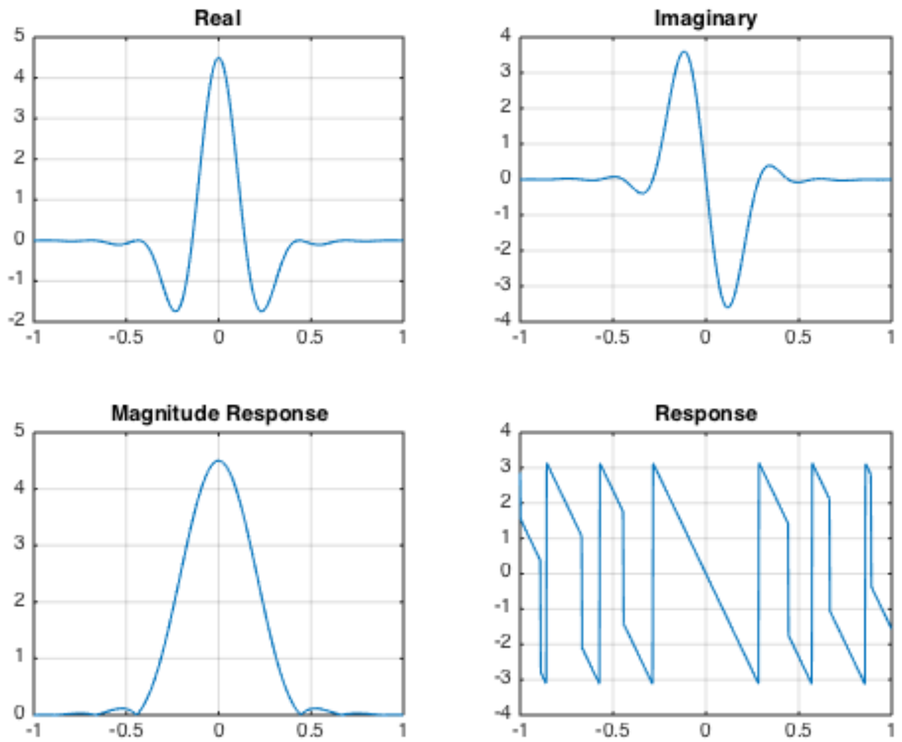

Fig.11 Real part, imaginary part, magnitude response and phase response of g[n]

Compared with plots from part a, the two results are quite different but are of the same type of symmetries. The differences in shapes are caused by the denominators of the two systems. g[n] is a finite length sequence, thus, making the real part, imaginary part and magnitude response acts like a pulse. While those of the system X are periodic due to the coefficients in y. meanwhile, the phase response, reflecting the transient change in the cycle period of an oscillation, are quite similar in shape.

(a) Draw the real part, imaginary part, magnitude and phase response of the following z-transform when its value is evaluated on the unit circle.

$$H(z) = \frac{2+5z^{-1}+9z^{-2}+5z^{-3}+3z^{-4}}{5+45z^{-1}+2z^{-2}+z^{-3}+z^{-4}} \quad (1)$$

```
>> num=[2 5 9 5 3];
den=[5 45 2 1 1];
w=-pi:0.01:pi;
h=freqz(num,den,w);
figure
subplot(2,2,1);
plot(w/pi,real(h));grid
title('Real');
subplot(2,2,2);
plot(w/pi,imag(h));grid
title('Imaginary');
subplot(2,2,3); plot(w/pi,abs(h));grid
title('Magnitude');
subplot(2,2,4); plot(w/pi,angle(h));grid
title('Phase');
```

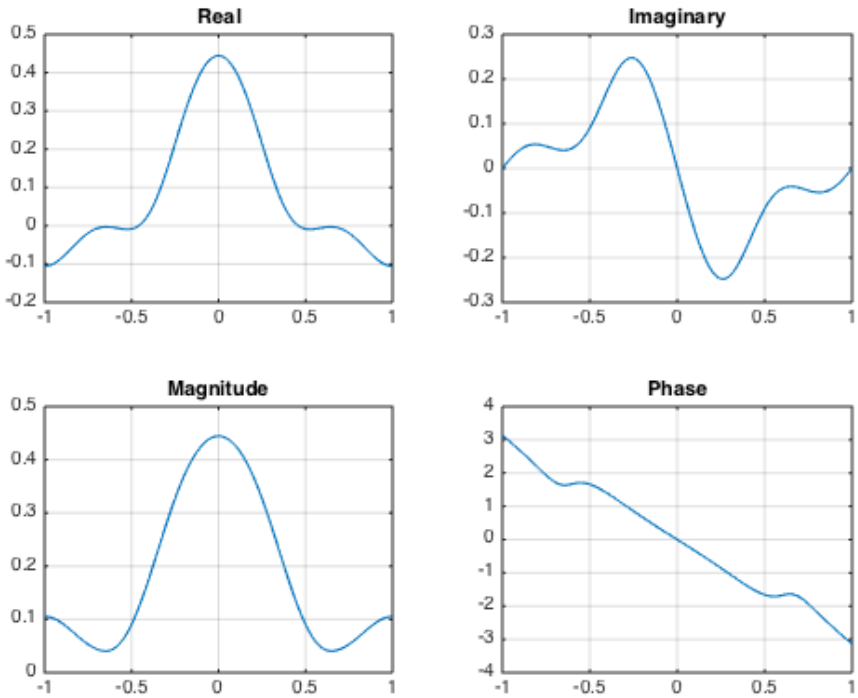

Fig.12 Real & imaginary part, magnitude& phase response of H(z)

From Fig. 12, we cannot observe the periodic property of z transform of H, thus, the range of w is expanded to be from -4*pi to 4*pi.

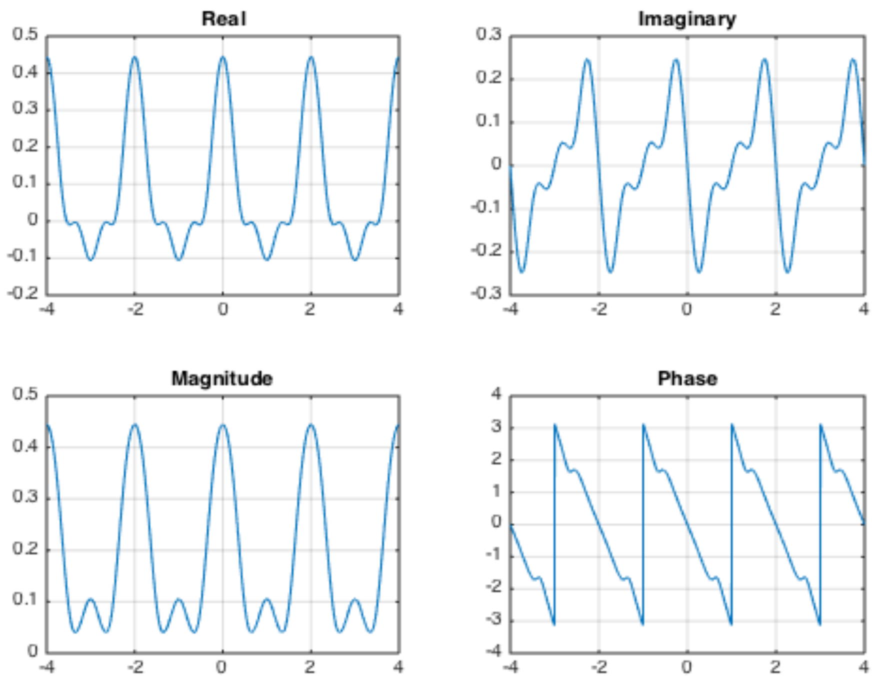

Fig.13 Real & imaginary part, magnitude& phase response of H(z)

b) Computer the poles and zeros of the z-transform in (1). Express (1) in factored form with the obtained poles and zeros, and generate the pole-zero plot of (1).

```
>> num=[2 5 9 5 3];
den=[5 45 2 1 1];
zplane(num,den);
[z,p,k]=tf2zp(num,den);
z
p
k


z =
   -1.0000 + 1.4142i
   -1.0000 - 1.4142i
   -0.2500 + 0.6614i
   -0.2500 - 0.6614i


p =
   -8.9576 + 0.0000i
   -0.2718 + 0.0000i
    0.1147 + 0.2627i
    0.1147 - 0.2627i


k =
      0.4000
```

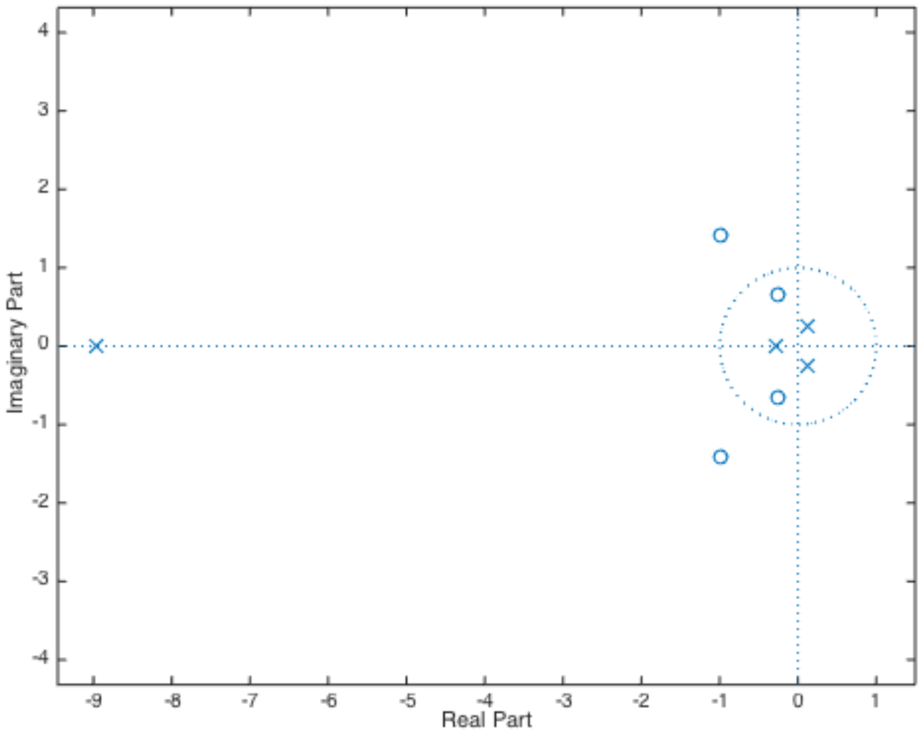

Fig.15 Pole-Zero plot of the z transform

c) From the pole-zero plot generated in part(b), determine the number of regions of convergence (ROC) of H(z). Show explicitly all possible ROCs. Can you tell from the pole-zero plot whether or not the DTFT exits?

As we get the poles of H, we can calculate the ranges parameter**:**

R1=|-8.9576| ;

R2=|-0.2718|;

R3=|0.1147|+|0.1147|;

R4=|0.1147|+|-0.1147|;

R3=R4, Thus, there are three points and four ranges of regions.

d) Determine the partial fraction expansion using residuez.

```
>> clear
num=[2 5 9 5 3];
den=[5 45 2 1 1];
```

```
w=-2*pi:0.01:2*pi;
[r,p,k]=residuez(num,den);
r
p
k

r =
    0.3109 + 0.0000i
   -1.0254 - 0.3547i
   -1.0254 + 0.3547i
   -0.8601 + 0.0000i

p =
   -8.9576 + 0.0000i
    0.1147 + 0.2627i
    0.1147 - 0.2627i
   -0.2718 + 0.0000i

k =
     3.0000
```

Getting the r and p values as numerator and demoniator, k as the constant value, we can rewrite H(z) in new form.

(e) Determine the rational form of a z-transform whose zeros are at z1=0.3, z2=2.5, z3=-0.2+0.4j, z4=-0.2-0.4j; the poles are at p1=0.5, p2=-0.75, p3=0.6+0.7j, p4=0.6-0.7j; and the gain constant k is 3.9. Generate the corresponding pole-zero plot.

```
>> clear
>> z=[0.3;2.5;-0.2+0.4j;-0.2-0.4j];
p=[0.5;0.75;0.6+0.7j;0.6-0.7j];
k=3.9;
[num,den]=zp2tf(z,p,k)
```

```
zplane(num,den);

num =

    3.9000   -9.3600   -0.6630   -1.0140    0.5850


den =

    1.0000   -2.4500    2.7250   -1.5125    0.3187

>>
```