DSP2 Week 4 experiment Report

2019023436 Kim, Hyunsoo

EXERCISE 1 a~d

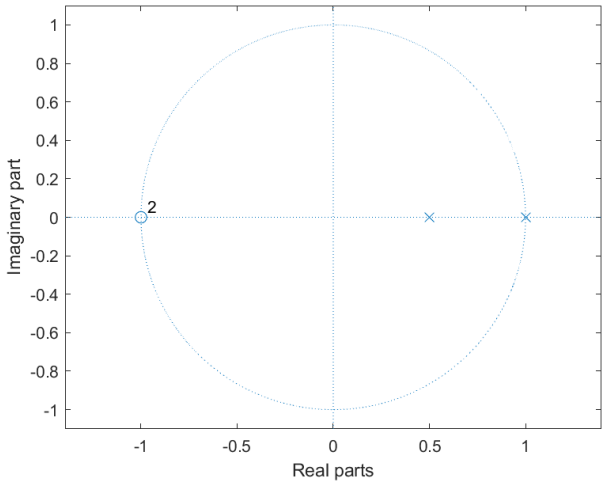
(Source Code)

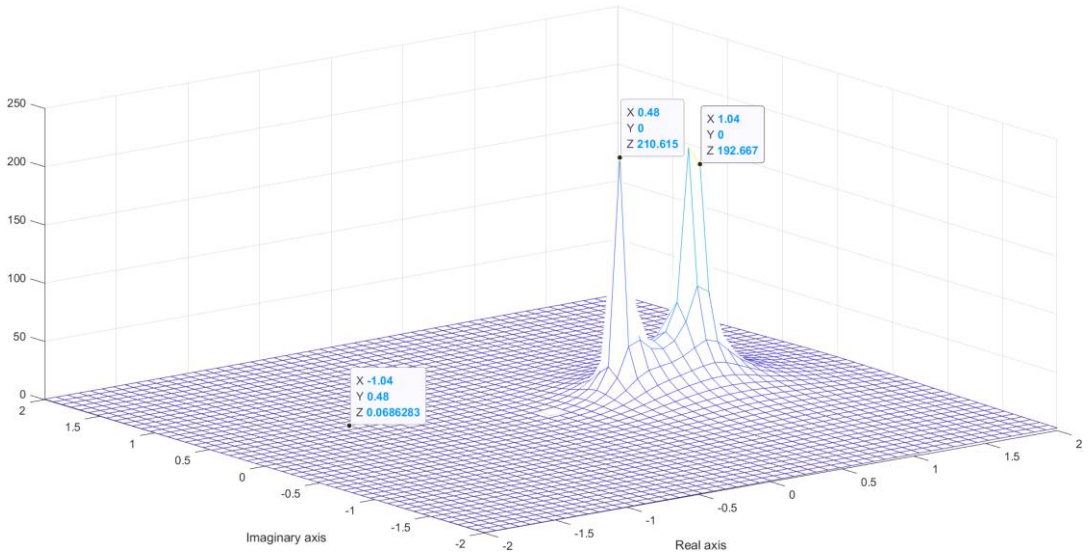
```
R = [-9; 8]
p = [0.5; 1]
C = 2
[num, den] = residuez(R, p, C )

zero = roots(num')
pole = roots(den')

zplane(zero, pole)
xlabel('Real parts')
ylabel('Imaginary part')

x = linspace(-2, 2, 51)
y = linspace(-2, 2, 51)
[X Y] = meshgrid(x, y);

z = X + j*Y;

f = @(co, x) co(1) + x .* (co(2) + x .* co(3));

H = f(num, 1./z) ./ f(den, 1./z)

mesh(X, Y, abs(H) )
xlabel('Real axis');
ylabel('Imaginary axis');
```

f is my anonymous function that receives a coefficient vector *co* and a complex number *x*, and it makes a polynomial of *x* with Horner's method. I made H(z) with this function *f*.

(Result)

```
R = 2×1
    -9
     8

p = 2×1
    0.5000
    1.0000

C = 2
num = 1×3
     1     2     1

den = 1×3
    1.0000   -1.5000    0.5000

zero = 2×1
    -1
    -1

pole = 2×1
    1.0000
    0.5000
```





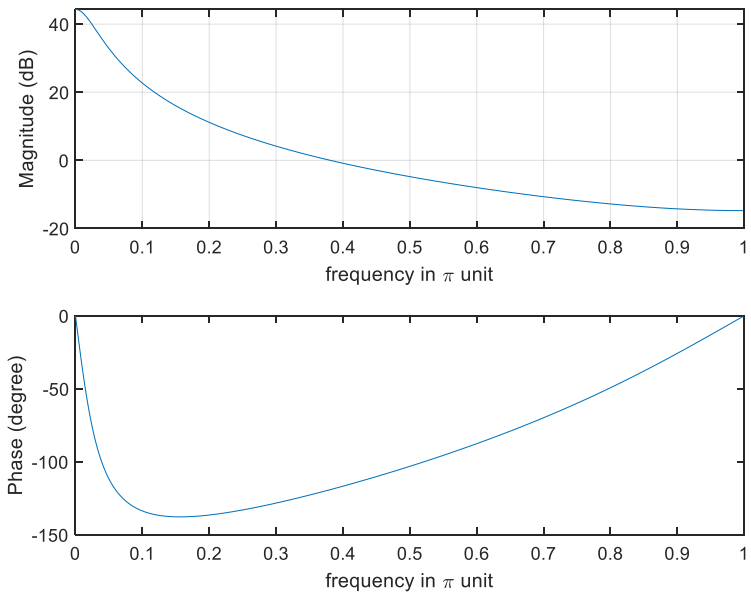
EXERCISE 2

a) (Source Code)

```
num = [1 1/3];
den = [1 -1.85*cos(pi/18) 0.83];
N = 512;

[h w ] = freqz(num, den, N)

mag_h = abs(h)
mag_h = 20 * log10(mag_h);
ang_h = angle(h)

subplot(2,1,1);
plot(w/pi, mag_h)
xlabel('frequency in \pi unit');
ylabel('Magnitude (dB)');
grid on

subplot(2,1,2);
plot(w/pi, ang_h*180/pi);
xlabel('frequency in \pi unit');
ylabel('Phase (degree)');
grid off
```

num is a vector of coefficients of numerator of H and den is a vector of coefficients of denominator of H. We can get a frequency vector w and complex frequency response vector H of this system, and plot them on the log-scaled coordinate.

(Result)



b) As you can see in these graph, magnitude of frequency response decreases when the frequency goes from 0 to 1. Because it's a log-scaled graph, low magnitude of high

frequency becomes nearly 0. So this system is the type of a low-pass filter.