Solution Set for Homework #7

By Prof. Brian Evans

November 11, 2025

PROBLEM 1: TRANSFER FUNCTIONS IN THE z DOMAIN

For each of the following linear time-invariant (LTI) systems, derive the transfer function, compute the poles and zeros, and plot the poles and zeros using `zplane`:

a) First-order unnormalized averaging filter (lowpass filter): $y[n] = x[n] + x[n-1]$ for $n \geq 0$ and the initial condition $x[-1] = 0$ to satisfy LTI properties.

b) First-order difference filter (highpass filter): $y[n] = x[n] - x[n-1]$ for $n \geq 0$ and the initial condition $x[-1] = 0$ to satisfy LTI properties

c) Second-order difference filter (highpass filter): $y[n] = x[n] - 2\,x[n-1] + x[n-2]$ for $n \geq 0$ and the initial condition $x[-1] = 0$ and $x[-2] = 0$ to satisfy LTI properties

Solution for part (a): $y[n] = x[n] + x[n-1]$ for $n \geq 0$ and $x[-1] = 0$ as a necessary condition for the system to be at rest. The impulse response is:

$$h[n] = \delta[n] + \delta[n-1]$$

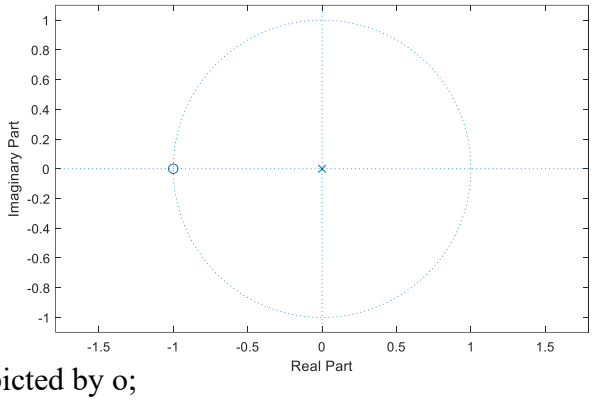
By performing the z-transform of the impulse response, we can calculate the transfer function:

$$H(z) = 1 + z^{-1} = \frac{z+1}{z}$$

The pole (root of the denominator) is at $z = 0$, and the zero (root of the nominator) is at $z = -1$.

Using zplane, we can plot zeros and poles:

```
zplane([1 1])
```



In the plot above, a pole is shown by × and a zero is depicted by o; hence, the system has one pole at $z = 0$ and one zero at $z = -1$

Solution for part (b): With

$$y[n] = x[n] - x[n-1]$$

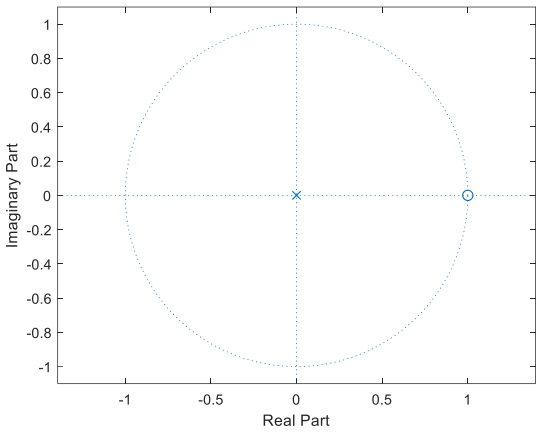
the impulse response is

$$h[n] = \delta[n] - \delta[n-1].$$

The transfer function is:

$$H(z) = 1 - z^{-1} = \frac{z-1}{z}$$

Therefore, system has one pole at $z = 0$, and one zero at $z = 1$.



MATLAB code:

```
zplane([1 -1])
```

Solution for part (c): With

$$y[n] = x[n] - 2x[n-1] + x[n-2]$$

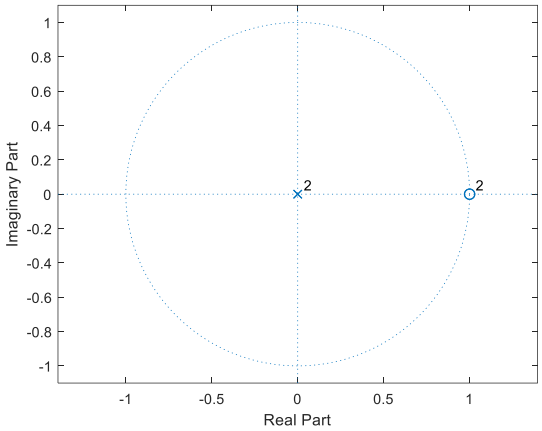
the impulse response is

$$h[n] = \delta[n] - 2\delta[n-1] + \delta[n-2]$$

and the transfer function is:

$$H(z) = 1 - 2\,z^{-1} + z^{-2} = \frac{z^2 - 2\,z + 1}{z^2}$$

System has two poles at $z = 0$, and two zeros at $z = 1$:




```
zplane([1 -2 1])
```

Epilogue: Armed with the z-transform, we'll take another look at the connection between convolution and polynomial multiplication mentioned in the Epilogue in Problem 2. From Problem 2, let's compute the convolution of $h_1[n]$ and $h_2[n]$ using z-domain techniques:

$$Z\{h_1[n] * h_2[n]\} = H_1(z)\,H_2(z)$$

$$H_1(z) = Z\{\delta[n] - \delta[n-1]\} = 1 - z^{-1}$$

$$H_2(z) = Z\{\delta[n] + \delta[n-2]\} = 1 + z^{-2}$$

$$H_1(z)\,H_2(z) = (1 - z^{-1})(1 + z^{-2}) = 1 - z^{-1} + z^{-2} - z^{-3}$$

$$h_1[n] * h_2[n] = Z^{-1}\{H_1(z)\,H_2(z)\} = Z^{-1}\{1 - z^{-1} + z^{-2} - z^{-3}\}$$

$$h_1[n] * h_2[n] = \delta[n] - \delta[n-1] + \delta[n-2] - \delta[n-3]$$

PROBLEM 2: DISCRETE-TIME AVERAGING FILTERS. ***34 points.***

For a discrete-time LTI system with input signal $x[n]$ and impulse response $h[n]$, the output signal $y[n]$ is the convolution of $h[n]$ and $x[n]$:

$$y[n] = h[n] * x[n] = \sum_{k=-\infty}^{\infty} h[k]\, x[n-k]$$

(a) Compute the output $y[n]$ when the input $x[n]$ is a rectangular pulse of amplitude 1 for $n = 0, 1, \ldots N_x - 1$ and amplitude 0 otherwise, and $x[n]$ is filtered by an LTI unnormalized averaging filter whose impulse response $h[n]$ is a rectangular pulse of amplitude 1 for $n = 0, 1, \ldots N_h - 1$ and amplitude 0 otherwise. Assume $N_x \neq N_h$.

i. Write the difference equation relating output $y[n]$ and input $x[n]$. *3 points.*

***Solution*: The impulse response $\boldsymbol{h[n]}$ has extent $\boldsymbol{n = 0, 1, \ldots N_h - 1}$:**

$$\boldsymbol{y[n] = h[n] * x[n] = \sum_{k=-\infty}^{\infty} h[k]\, x[n-k] = \sum_{k=0}^{N_h-1} h[k]\, x[n-k]}$$

$$\boldsymbol{y[n] = h[0]\, x[n] + h[1]\, x[n-1] + \cdots + h[N_h - 1]\, x[n-(N_h-1)]}$$

Since $\boldsymbol{h[n] = 1}$ for $\boldsymbol{n = 0, 1, \ldots, N_h - 1}$,

$$\boldsymbol{y[n] = x[n] + x[n-1] + x[n-2] + \cdots + x[n-(N_h-1)]}$$

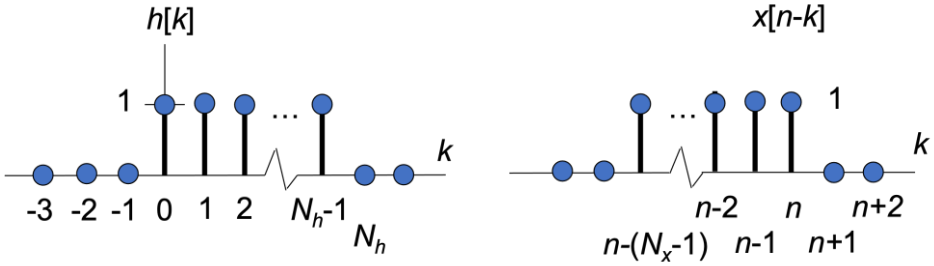
ii. What are the initial conditions and what values should they be set to? *3 points.*

***Solution:* Initial conditions can be discovered by computing the first several values of $\boldsymbol{y[n]}$ for n ≥ 0: $\boldsymbol{y[0] = x[0] + x[-1] + x[-2] + \cdots + x[-(N_h - 1)]}$.**

The initial conditions are $\boldsymbol{x[-1], x[-2], \ldots, x[-(N_h - 1)]}$. They must be set to zero as necessary conditions for LTI system properties to hold.

iii. Develop a formula for $y[n] = h[n] * x[n]$ using the convolution definition in terms of N_x and N_h. Show the intermediate steps in computing the convolution. *6 points.*

***Solution:* First, we'll define $\boldsymbol{N_{min} = \min(N_h, N_x)}$ and $\boldsymbol{N_{max} = \max(N_h, N_x)}$ and the length of the convolution result $\boldsymbol{N_y = N_h + N_x - 1}$. As we flip and slide $\boldsymbol{x[n-k]}$ across $\boldsymbol{h[k]}$, where the shift $\boldsymbol{n}$ is with respect to the convolution variable $\boldsymbol{k}$, the extent of $\boldsymbol{h[k]}$ is $\boldsymbol{0 \le k \le N_h - 1}$ and the extent of $\boldsymbol{x[n-k]}$ is $\boldsymbol{n - (N_x - 1) \le k \le n}$.**



There are five cases to consider:

1. ***No overlap.*** $\boldsymbol{n < 0}$**. Amplitude is 0.**
2. ***Partial overlap.*** $\boldsymbol{0 \le n \le N_{min} - 1}$**. Amplitude is** $\boldsymbol{(n+1)}$**.**
   Initial overlap of one sample at $\boldsymbol{n = 0}$ **with a product of one. Each shift by one in** ***n*** **adds one more overlapping sample with product of one.**

$$\boldsymbol{y[n] = \sum_{k=0}^{n} h[k]\, x[n-k] = \sum_{k=0}^{n} 1 = (n+1)}$$

3. ***Complete overlap.*** $\boldsymbol{N_{min} - 1 \le n \le N_{max} - 1}$**. Amplitude is** $\boldsymbol{N_{min}}$**.**
   Here, $\boldsymbol{N_{min}}$ **samples overlap, and each sample has a value of one.**
4. ***Partial overlap.*** $\boldsymbol{N_{max} \le n \le N_y - 1}$**. Amplitude is** $\boldsymbol{N_y - n}$**.**
   Amplitude reduces by one each time ***n*** **is incremented.**

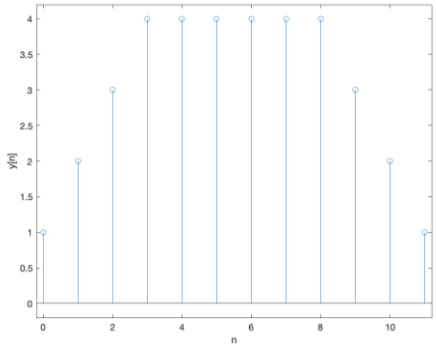
$$\boldsymbol{y[n] = \sum_{k=n-(N_x-1)}^{N_h-1} 1 = (N_h - 1) + (N_x - 1) + 1 - n = N_y - n}$$

5. **No overlap.** $\boldsymbol{n \ge N_y}$**. Amplitude is 0.**

iv. Validate the formula for $y[n]$ by using Matlab to compute the convolution for $N_x = 9$ samples and $N_h = 4$ samples. *3 points.*

Solution: **Using Matlab:**

```
h = ones(1, 4);
x = ones(1, 9);
y = conv(h, x);
n = 0 : 11;
stem(n, y);
xlim([-0.2 11.2]);
ylim([-0.2, 4.2]);
xlabel('n');
ylabel('y[n]');
```



v. Use the *z*-transform to find $y[n] = h[n] * x[n]$. Track region of convergence. *3 points.*

Solution: **Convolution in the discrete-time domain becomes a product in the z-transform domain:** $\boldsymbol{Y(z) = H(z)X(z)}$**. Here,**

$$\boldsymbol{H(z) = \sum_{n=0}^{N_h-1} h[n]\, z^{-n} = 1 + z^{-1} + z^{-2} + \cdots + z^{-(N_h-1)} \text{ if } z \neq 0}$$

$$\boldsymbol{X(z) = \sum_{n=0}^{N_x-1} x[n]\, z^{-n} = 1 + z^{-1} + z^{-2} + \cdots + z^{-(N_x-1)} \text{ if } z \neq 0}$$

Convolution in the time domain has become polynomial multiplication in the z-transform domain. The polynomial multiplication will produce a polynomial whose coefficients will fit a trapezoidal pattern because $\boldsymbol{N_x \neq N_h}$**.**

$$\boldsymbol{X(z) = 1 + 2\, z^{-1} + 3\, z^{-2} + \cdots + 2\, z^{-(N_y-2)} + z^{-(N_y-1)} \text{ if } z \neq 0}$$

(a) An LTI system outputs the weighted average of the previous output value $y[n-1]$ and current input value $x[n]$ using difference equation $y[n] = 0.9\,y[n-1] + 0.1\,x[n]$ for $n \geq 0$

i. What are the initial conditions and what value should they be set to? *3 points.*

***Solution*: Initial conditions can be discovered by computing the first values of $y[n]$ for $n \geq 0$: $y[0] = 0.9\,y[-1] + 0.1\,x[0]$. Initial condition $y[-1] = 0$ for LTI to hold.**

ii. Compute a formula for the impulse response $h[n]$ for the system. *3 points.*

***Solution:* To compute the impulse response $h[n]$, we let the input be an impulse $\delta[n]$: $h[n] = 0.9\,h[n-1] + 0.1\,\delta[n]$ for $n \geq 0$ with $h[-1] = 0$. We'll compute the output values and infer the impulse response as in LTI Example #2 on Lecture Slide 11-4:**

$h[0] = 0.9\,h[-1] + 0.1\,\delta[0] = 0.1$

$h[1] = 0.9\,h[0] + 0.1\,\delta[1] = 0.1\,(0.9)$

$h[2] = 0.9\,h[1] = 0.1\,(0.9)^2$ **…**

Inferring the pattern gives $h[n] = 0.1\,(0.9)^n\,u[n]$.

iii. Develop a formula for $y[n] = h[n] * x[n]$ using the convolution definition when the input signal is $x[n] = 0.8^n\,u[n]$. *6 points.*

***Solution:* With $h[n] = 0.1\,(0.9)^n\,u[n]$ and $x[n] = 0.8^n\,u[n]$.**

$$y[n] = \sum_{k=-\infty}^{\infty} h[k]\,x[n-k] = \sum_{k=-\infty}^{\infty} \left(0.1\,(0.9)^k\,u[k]\right)\left((0.8)^{n-k}\,u[n-k]\right)$$

$u[k]$ is 1 for $k \geq 0$ and 0 otherwise. $u[n-k]$ is 1 when $n-k \geq 0$ or $k \leq n$, and 0 otherwise. Limits of summation become $k = 0$ and $k = n$ and $n \geq 0$ because $k \geq 0$:

$$y[n] = u[n] \sum_{k=0}^{n} 0.1\,(0.9)^k\,(0.8)^{n-k} = 0.1\,(0.8)^n\,u[n] \sum_{k=0}^{n} (0.9)^k\,(0.8)^{-k}$$

$$y[n] = 0.1\,(0.8)^n\,u[n] \sum_{k=0}^{n} \left(\frac{0.9}{0.8}\right)^k = 0.1\,(0.8)^n \frac{1-\left(\frac{0.9}{0.8}\right)^{n+1}}{1-\left(\frac{0.9}{0.8}\right)} u[n]$$

$$y[n] = 0.1\,\frac{(0.8)^{n+1} - (0.9)^{n+1}}{0.8 - 0.9}\,u[n] = -(0.8)^{n+1}\,u[n] + (0.9)^{n+1}\,u[n]$$

For more info, see Handout F Convolution of Two Causal Exponential Sequences.

iv. Use the z-transform to find $y[n] = h[n] * x[n]$. Track region of convergence. *4 points.*

***Solution:* Convolution in the discrete-time domain becomes a product in the z-transform domain: $Y(z) = H(z)X(z)$. Here,**

$H(z) = \frac{0.1}{1-0.9\,z^{-1}}$ **for** $|z| > 0.9$ **and** $X(z) = \frac{1}{1-0.8\,z^{-1}}$ **for** $|z| > 0.8$

$Y(z) = \left(\frac{0.1}{1-0.9\,z^{-1}}\right)\left(\frac{1}{1-0.8\,z^{-1}}\right)$ **for** $|z| > \max(0.8, 0.9)$

We'll use partial fractions decomposition to express the transfer function as a sum of two first-order terms and apply the inverse z-transform. See Ex. 8-10 on p. 219.

$Y(z) = 0.1\left(\frac{A}{1-0.9\,z^{-1}} + \frac{B}{1-0.8\,z^{-1}}\right)$ **which gives** $y[n] = A\,(0.9)^n\,u[n] + B(0.8)^n\,u[n]$

Using the method from Ex. 8-10 on p. 219, $A = \frac{1}{1-\frac{0.8}{0.9}} = 0.9$ **and** $B = \frac{1}{1-\frac{0.9}{0.8}} = -0.8$ **.**

PROBLEM 3: TRANSFER FUNCTION & FREQUENCY RESPONSE CONNECTIONS. ***16 points.***

Signal Processing First, problem P-8.16, page 242.

In addition, for each of the four filters,

i. give a formula for the transfer function in the z-domain including the region of convergence,

ii. give a formula for the frequency response from the transfer function in the z-domain in part i. why can we convert from the transfer function in the z-domain to a frequency response directly?

iii. plot the magnitude response in the frequency domain

iv. indicate the frequency selectivity as lowpass, highpass, bandpass, or bandstop**.**

Connecting filter poles/zeros to its frequency selectivity: Please see lecture slides 11-6 through 11-11 and watch the recording from our lecture on Oct. 31, 2023, from 1:42 to 39:03, which is available on Canvas. I have another recording of the same demos on YouTube video in spring 2014 for the Real-Time Digital Signal Processing Lab course from the 1:29 to 22:25 and from 43:01 to the end (50:51). Takeaways from either videorecording:

- When poles and zeros are separated in angle, the angles of the poles near the unit circle indicate the frequencies in the passband(s) and the angles of the zeros near or on the unit circle indicate the frequencies in the stopband(s). Please see lecture slide 11-7.
- Poles must be inside the unit circle for bounded-input bounded-output (BIBO) stability. Please see lecture slides 11-12 and 11-13.

Solution: **Pole-Zero Plot #1**

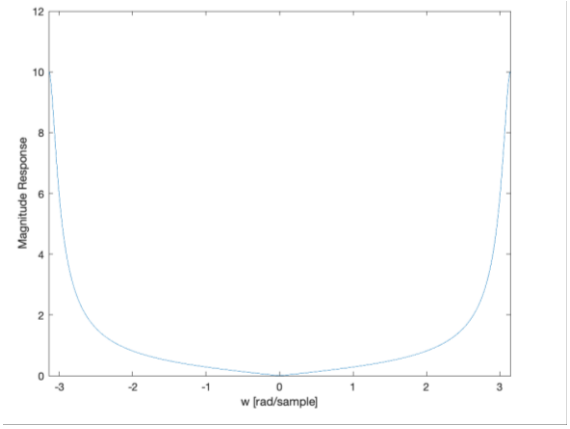
$\boldsymbol{H(z) = C\,\frac{z-z_0}{z-p_0}}$ **for** $\boldsymbol{|z| > |p_0|}$ **where** $\boldsymbol{z_0 = 1}$ **and** $\boldsymbol{p_0 = -0.9}$ **and** $\boldsymbol{C}$ **is given in Matlab code.**

Because the region of convergence $\boldsymbol{|z| > 0.9}$ **includes the unit circle** $\boldsymbol{e^{j\omega}}$**,**

$\boldsymbol{H_{freq}(\omega) = H(z)]_{z=e^{j\omega}} = C\,\frac{e^{j\omega}-z_0}{e^{j\omega}-p_0}}$ **for** $\boldsymbol{|z| > |p_0|}$

```
%%% Specify the filter
z0 = 1;
p0 = -0.9;
C = 0.5;   %% to match (D)

%%% Plot the magnitude response
w = -pi : (2*pi/10000) : pi;
Hnumer = (1 - z0*exp(-j*w));
Hdenom = (1 - p0*exp(-j*w));
H = C * Hnumer ./ Hdenom;
plot(w, abs(H));
xlim( [-pi pi] );
xlabel('w [rad/sample]');
ylabel('Magnitude Response');
```



Filter has a highpass selectivity and matches (D).

Solution: **Pole-Zero Plot #2**

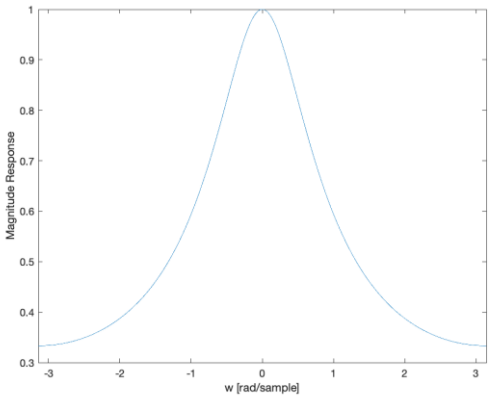
$H(z) = C\,\frac{z-z_0}{z-p_0}$ **for** $|z| > |p_0|$ **where** $z_0 = 0$ **and** $p_0 = 0.5$ **and** C **is given in Matlab code.**

Because the region of convergence $|z| > 0.5$ **includes the unit circle** $e^{j\omega}$**,**

$$H_{freq}(\omega) = H(z)]_{z=e^{j\omega}} = C\,\frac{e^{j\omega}-z_0}{e^{j\omega}-p_0} \text{ for } |z| > |p_0|$$

```
%%% Specify the filter
z0 = 0;
p0 = 0.5;
C = 0.5;    %% to match (B)

%%% Plot the magnitude response
w = -pi : (2*pi/10000) : pi;
Hnumer = (1 - z0*exp(-j*w));
Hdenom = (1 - p0*exp(-j*w));
H = C * Hnumer ./ Hdenom;
plot(w, abs(H));
xlim( [-pi pi] );
xlabel('w [rad/sample]');
ylabel('Magnitude Response');
```



Filter has a lowpass selectivity and matches (B).

Solution: **Pole-Zero Plot #3**

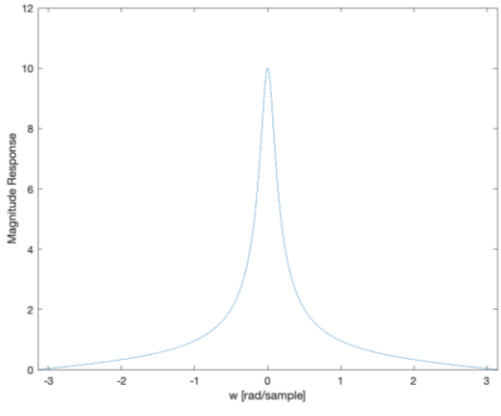
$H(z) = C\,\frac{z-z_0}{z-p_0}$ **for** $|z| > |p_0|$ **where** $z_0 = -1$ **and** $p_0 = 0.9$ **and** C **is given in Matlab code.**

Because the region of convergence $|z| > 0.9$ **includes the unit circle** $e^{j\omega}$**,**

$$H_{freq}(\omega) = H(z)]_{z=e^{j\omega}} = C\,\frac{e^{j\omega}-z_0}{e^{j\omega}-p_0} \text{ for } |z| > |p_0|$$

```
%%% Specify the filter
z0 = -1;
p0 = 0.9;
C = 0.5;    %% to match (A)

%%% Plot the magnitude response
w = -pi : (2*pi/10000) : pi;
Hnumer = (1 - z0*exp(-j*w));
Hdenom = (1 - p0*exp(-j*w));
H = C * Hnumer ./ Hdenom;
plot(w, abs(H));
xlim( [-pi pi] );
xlabel('w [rad/sample]');
ylabel('Magnitude Response');
```



Filter has a lowpass selectivity and matches (A).

Solution: **Pole-Zero Plot #4**

$H(z) = C\,\frac{(z-z_0)(z-z_1)}{(z-p_0)(z-p_1)}$ **for** $|z| > \max\{\,|p_0|, |p_1|\,\}$ **where** $z_0 = 0$, $z_1 = 0$, $p_0 = 0.9\,e^{j\frac{\pi}{6}}$, $p_1 = 0.9\,e^{-j\frac{\pi}{6}}$ **and** C **is given in the Matlab code below.**

Because the region of convergence $|z| > 0.9$ **includes the unit circle** $e^{j\omega}$**,**

$$H_{freq}(\omega) = H(z)]_{z=e^{j\omega}} = C\,\frac{(e^{j\omega}-z_0)(e^{j\omega}-z_1)}{(e^{j\omega}-p_0)(e^{j\omega}-p_1)} \text{ for } |z| > |p_0|$$

We can expand the numerator and denominator of the transfer function $H(z)$ **:**

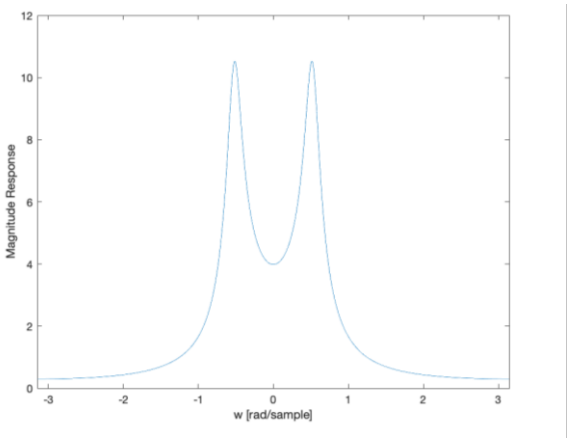
$$H(z) = C\,\frac{(z-z_0)(z-z_1)}{(z-p_0)(z-p_1)} = C\,\frac{z^2-(z_0+z_1)\,z+z_0z_1}{z^2-(p_0+p_1)\,z+p_0p_1} = C\,\frac{b_0z^2+b_1z+b_2}{a_0z^2+a_1z+a_2}$$

```
%%% Specify the filter
%%% Zeros and numerator coefficients
z0 = 0;
z1 = 0;
b0 = 1;
b1 = -(z0+z1);
b2 = z0*z1;
numerCoeffs = [b0 b1 b2];

%%% Poles and denominator coefficients
p0 = 0.9*exp(j*pi/6);
p1 = 0.9*exp(-j*pi/6);
a0 = 1;
a1 = -(p0+p1);
a2 = p0*p1;
denomCoeffs = [a0 a1 a2];

%%% Gain for the filter
C = 1;

%%% Plot the magnitude response
w = -pi : (2*pi/10000) : pi;
Hnumer = (b0 + b1*exp(-j*w) + b2*exp(-j*2*w));
Hdenom = (a0 + a1*exp(-j*w) + a2*exp(-j*2*w));
H = C * Hnumer ./ Hdenom;
plot(w, abs(H));
xlim( [-pi pi] );
xlabel('w [rad/sample]');
ylabel('Magnitude Response');
```



Filter has a bandpass selectivity and matches (E).