**Solution Set for Homework #7**

*By Prof. Brian Evans*

November 12, 2024

**PROBLEM 1: TRANSFER FUNCTIONS IN THE 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:

1. First-order unnormalized averaging filter (lowpass filter): and the initial condition
2. First-order difference filter (highpass filter): and the initial condition
3. Second-order difference filter (highpass filter): and the initial condition

**Solution for part (a):**  for and as a necessary condition for the system to be at rest.The impulse response is:

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

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



the impulse response is

.

The transfer function is:

Therefore, system has one pole at *z* = 0,

and one zero at *z* = 1.

MATLAB code:

zplane([1 -1])

****

**Solution for part (c):** With



the impulse response is



and the transfer function is:

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 and using *z*-domain techniques:

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

For a discrete-time LTI system with input signal and impulse response , the output signal is the convolution of and :

(a) Compute the output when the input is a rectangular pulse of amplitude 1 for and amplitude 0 otherwise, and is filtered by an LTI unnormalized averaging filter whose impulse response is a rectangular pulse of amplitude 1 for and amplitude 0 otherwise. Assume .

1. Write the difference equation relating output and input . *3 points.*

***Solution*: The impulse response has extent :**

**Since for ,**

1. 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 for n ≥ 0: .**

**The initial conditions are . They must be set to zero as necessary conditions for LTI system properties to hold.**

1. Develop a formula for using the convolution definition in terms of . Show the intermediate steps in computing the convolution. *6 points.*

**A picture containing indoor, night sky

Description automatically generatedA picture containing indoor, night sky

Description automatically generated*Solution:* First, we’ll define and and the length of the convolution result . As we flip and slide across , where the shift *n* is with respect to the convolution variable *k*, the extent of is and the extent of is .**

**There are five cases to consider:**

1. ***No overlap*. . Amplitude is 0.**
2. ***Partial overlap*. . Amplitude is .**

**Initial overlap of one sample at with a product of one. Each shift by one in *n* adds one more overlapping sample with product of one.**

1. ***Complete overlap*. . Amplitude is .**

**Here, samples overlap, and each sample has a value of one.**

1. ***Partial overlap*. 1. Amplitude is .**

**Amplitude reduces by one each time *n* is incremented.**

1. **No overlap. . Amplitude is 0.**
2. ***Chart

   Description automatically generated***Validate the formula for by using Matlab to compute the convolution for *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]');

1. Use the *z*-transform to find . Track region of convergence. *3 points.*

***Solution:* Convolution in the discrete-time domain becomes a product in the z-transform domain: *.* Here,**

**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 .**

1. An LTI system outputs the weighted average of the previous output value and current input value using difference equation for

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 for *n* ≥ 0: . Initial condition for LTI to hold.**

ii. Compute a formula for the impulse response for the system. *3 points.*

***Solution:* To compute the impulse response , we let the input be an impulse for *n* ≥ 0 with . We’ll compute the output values and infer the impulse response as in LTI Example #2 on Lecture Slide 11-4:**

**…**

**Inferring the pattern gives .**

iii. Develop a formula for using the convolution definition when the input signal is *6 points.*

***Solution:* With and**

**is 1 for and 0 otherwise. is 1 when or , and 0 otherwise. Limits of summation become and and because :**

**For more info, see** [**Handout F Convolution of Two Causal Exponential Sequences**](http://users.ece.utexas.edu/~bevans/courses/signals/handouts/Appendix%20F%20Convolution%20Exp%20Sequences.pdf)**.**

iv. Use the *z*-transform to find . Track region of convergence. *4 points.*

***Solution:* Convolution in the discrete-time domain becomes a product in the z-transform domain: *.* Here,**

**and**

**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.**

**which gives**

**Using the method from Ex. 8-10 on p. 219, and .**

**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](https://www.youtube.com/watch?v=WWEKNvvJBvs&list=PLaJppqXMef2ZHIKM4vpwHIAWyRmw3TtSf) 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**

**where and and *C* is given in Matlab code.**

Shape

Description automatically generated**Because the region of convergence includes the unit circle ,**

%%% 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**

**where and and *C* is given in Matlab code.**

**Because the region of convergence includes the unit circle ,**

Chart, histogram

Description automatically generated%%% 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**

**where and and *C* is given in Matlab code.**

Chart

Description automatically generated with medium confidence**Because the region of convergence includes the unit circle ,**

%%% 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**

**where , , , and *C* is given in the Matlab code below.**

**Because the region of convergence includes the unit circle ,**

**We can expand the numerator and denominator of the transfer function :**

Chart, histogram

Description automatically generated%%% 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).**