



## Signals II

### Laboratory 4

#### The Z-transform and the analysis of LTI systems

##### 3.1 Rational Z-transforms

As noted in your textbook, the common Z-transform pairs can be written as a ratio of two polynomials in  $z^{-1}$  or  $z$ . That is

$$H(z) = \frac{N(z)}{D(z)} = \frac{b_0 + b_1 z^{-1} + \dots + b_M z^{-M}}{a_0 + a_1 z^{-1} + \dots + a_N z^{-N}}$$

If we know that a system can be described by  $H(z)$ , its output to a known input  $X(z)$  can be computed by multiplying  $H(z) X(z)$ . On the other hand, if we know the input  $X(z)$  and the output  $Y(z)$ , the system can be identified. The time domain solution can be easily obtained by using partial-fraction expansions.

##### Problem 1.

Consider the system:

$$H(z) = \frac{z^2 - z}{z^2 + 3z + 2}$$

How many poles and zeros do you have?

Poles \_\_\_\_\_

Zeros \_\_\_\_\_

Use the command *roots* in Matlab to determine the numerical values of these poles and zeros. Use *roots([1 -1 0])* for the zeros and *roots([1 3 2])* for the poles.

Poles =

Zeros =

Describing the same system in terms of increasing order of  $z^{-1}$ , we have

$$H(z) = \frac{1 - z^{-1}}{1 + 3z^{-1} + 2z^{-2}}$$

Use *roots([1 -1])* for the zeros and *roots([1 3 2])* for the poles. Do you have the same answer?

Poles =

Zeros =

Compute  $h(n)$  using the poles and zeros found with Matlab.

$h(n) =$

## Problem 2

Use the *filter* command in Matlab to generate and plot the impulse response  $h[n]$  of the following second order difference equation. Plot  $h[n]$  in the range of  $-10 \leq n \leq 100$ .

$$y[n] - 1.8\cos(\pi/16)y[n-1] + 0.81y[n-2] = x[n] + 0.5x[n-1]$$

Determine the impulse response using the inverse Z-transform and confirm your results.

### 3.2 Frequency response of difference equations

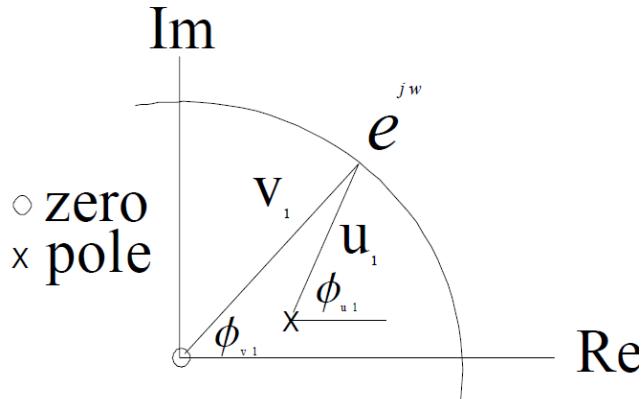
The frequency response of a system  $h(n)$  can be obtained by evaluating the z-transform of the system on the unit circle. That is, if a system is described by

$$H(z) = A \frac{\prod_{k=1}^K (z - z_k)}{\prod_{m=1}^M (z - p_m)},$$

the contribution of each pole and each zero to the magnitude in the frequency domain depends on the length of the vector from the pole or zero to the point  $e^{jw}$ . Thus, taking the magnitude and evaluating at  $e^{jw}$  yields:

$$|H(e^{jw})| = |A| \frac{\prod_{k=1}^K |e^{jw} - z_k|}{\prod_{m=1}^M |e^{jw} - p_m|} = |A| \frac{\prod_{k=1}^K v_k}{\prod_{m=1}^M u_m}$$

as shown in the next figure.

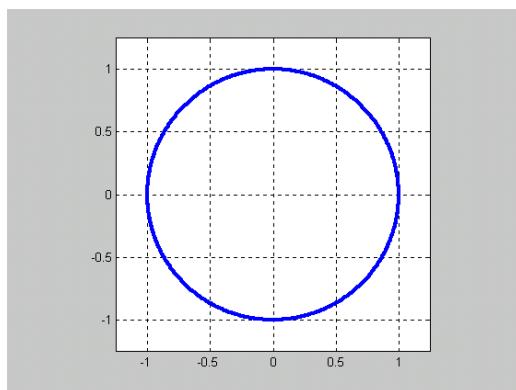


The phase is given by

$$\angle H(e^{jw}) = \angle A + \sum_{k=1}^K \angle(e^{jw} - z_k) + \sum_{m=1}^M \angle(e^{jw} - p_k)$$

### Problem 3.

Plot the poles and zeros of the system in Problem 2.



Using the command *freqz*, plot the magnitude and phase of the frequency response of this system. If the system is to be defined as a filter, what kind of filter it is?

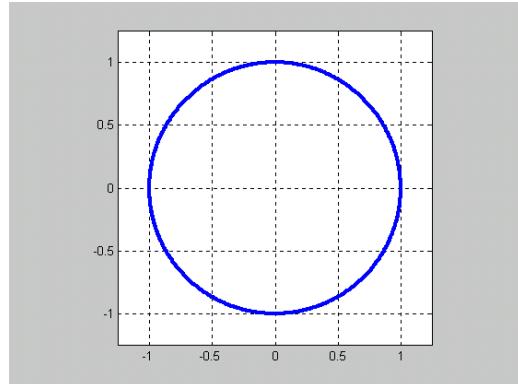
Filter is \_\_\_\_\_

Do the same for the following filters:

$$* \quad y[n] + 0.13y[n-1] + 0.52y[n-2] + 0.3y[n-3] = 0.16x[n] - 0.48x[n-1] + 0.48x[n-2] - 0.16x[n-3]$$

$$H(z) =$$

---

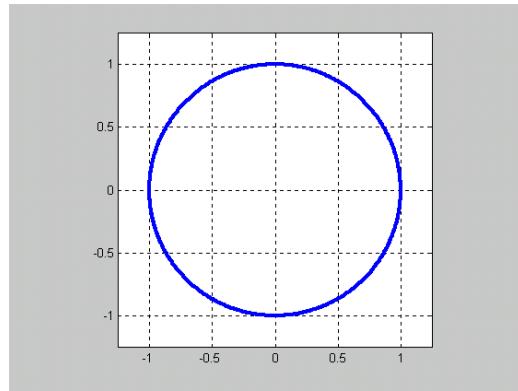


Filter is \_\_\_\_\_

$$* \quad y[n] - 0.268y[n-2] = 0.634x[n] - 0.634x[n-2]$$

$$H(z) =$$

---

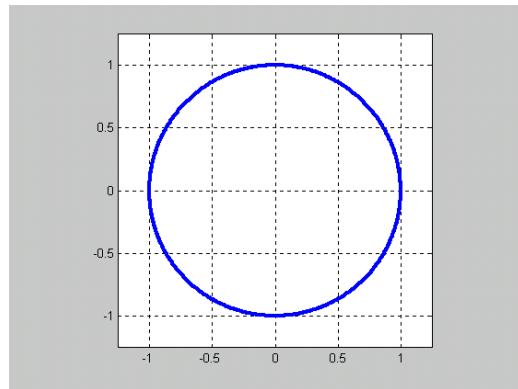


Filter is \_\_\_\_\_

$$* \quad y[n] + 0.268y[n-2] = 0.634x[n] + 0.634x[n-2]$$

$$H(z) =$$

---

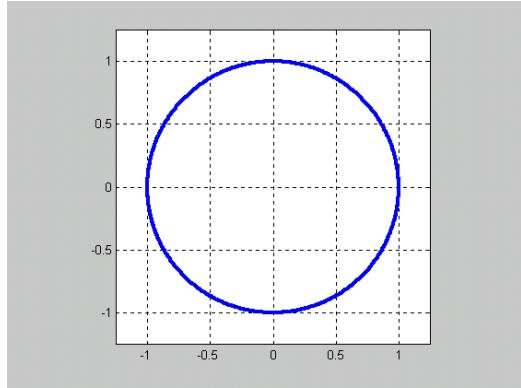


Filter is \_\_\_\_\_

$$* \quad 10y[n] - 5y[n-1] + 2y[n-2] = x[n] - 5x[n-1] + 10x[n-2]$$

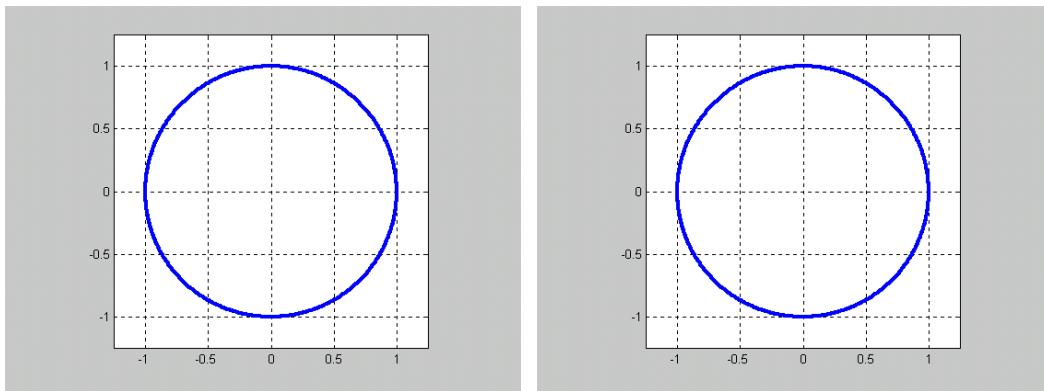
$$H(z) =$$

---



Filter is \_\_\_\_\_

Without taking into consideration the cut-off frequencies, draw the poles and zeros of the simplest low-pass and high-pass filters you can think of. That is, use a minimum number of poles and zeros. Explain your results.



You will be given an acoustic signal in the file BoatWhale.mat. Note that the signal is already in Matlab format. The samples correspond to a recording using a hydrophone of a whale sound but it also records the sound of the boat ( $F_s = 22050\text{Hz}$ ). Design a filter that can attenuate the sound of the boat. Please include the pole-zero diagram and your methodology.

Matlab instructions that can help you:

To listen to the audio use *sound*

To obtain a polynomial from roots use the command *poly*

The command *tf* can give you the pole-zero plot.

Hint, the sound of the boat has a good number of harmonics.