Signals and Systems MATLAB HW4

Deadline: 2022/5/31 23:59

Z-Transform

The objective of this homework is to learn MATLAB commands related to the z-transform. You will learn how to use the following MATLAB commands:

- **residuez**
- **zplane**
- **roots**
- **zp2sos**

You may use the commands **freqz** and **filter** learned from the last homework to finish the problem set below.

1. Background

The z-transform of a signal $x[n]$ can be expressed as

$$X(z) = \frac{b_1 + b_2 z^{-1} + \cdots + b_N z^{-N+1}}{a_1 + a_2 z^{-1} + \cdots + a_M z^{-M+1}}\,, \tag{1}$$

where $\{a_k\}_{k=1}^{M}$ and $\{b_k\}_{k=1}^{N}$ are the coefficients of the denominator and nominator, respectively.

- To obtain the inverse z-transform of (1), we may apply the command **residuez**. This command computes the partial-fraction expansions for z-transform expressed as a ratio of two polynomials in z^{-1}. Let vector **r** denote the partial-fraction expansion coefficients corresponding to the poles in **p** and vector **k** contain the direct terms, the syntax is

  [r, p, k] = residuez(b, a) (2)

  where

  $$\mathbf{a} = [a_1, a_2, \ldots, a_M]$$

  and

  $$\mathbf{b} = [b_1, b_2, \ldots, b_N].$$

  With these computed vectors (i.e., **r**, **p** and **k**), we may obtain $x[n]$ by observing the coefficients of the partial-fraction expansion.

- To find the poles and zeros of $x[n]$, we may use the command **roots**. For a polynomial with finite orders which is written as

$$C(z) = c_1 + c_2 z^{-1} + \cdots + c_K z^{-K+1},$$

we may use the command **roots**($[c_1, c_2, \ldots, c_K]$) to find its roots. Furthermore, the poles and zeros in (1) can be displayed by using the command **zplane(b, a)**.

- A useful method to implement a discrete-time LTI system is to divide the system into two LTI systems. This method reduces the probability of overflow of registers memorizing the coefficients. Now we redefine (1) as

$$X(z) = \frac{(1 - d_1 z^{-1})(1 - d_2 z^{-1}) \cdots (1 - d_L z^{-1})}{(1 - m_1 z^{-1})(1 - m_2 z^{-1}) \cdots (1 - m_P z^{-1})},$$

and define

$$\mathbf{z} = [d_1, d_2, \ldots, d_L]$$
$$\mathbf{p} = [m_1, m_2, \ldots, m_P]$$
$$\mathbf{k} = 1.$$

By giving the command **sos = zp2sos(z, p, k)**, we may obtain two systems with transfer function $X_1(z)$ and $X_2(z)$, respectively, such that $X(z) = X_1(z)X_2(z)$.

2. Questions

A causal discrete-time LTI system has the transfer function

$$H(z) = \frac{0.09(z-1)^2(z+1)^2}{(z - 0.3 - 0.4i)(z - 0.3 + 0.4i)(z - 0.1 - 0.1i)(z - 0.1 + 0.1i)}.$$

Please write a MATLAB script (saved as **myztran.m**) to implement the following problems.

(a) (10%) Use the MATLAB function **zplane** to plot the locations of poles and zeros of $H(z)$. Please also state the ROC in your report.

(b) (20%) Use the output of the MATLAB function **residuez** to construct the real $h[n]$, where $h[n]$ is the inverse z-transform of $H(z)$. Then, use the MATLAB function **stem** to plot $h[n]$ vs n for $n = 0 \sim 20$.
Hint: What is the meaning of **r**, **p** and **k** in eq. (2)?

(c) (15%) Use the MATLAB function **plot** to plot the magnitude and phase response of $H(z)$ vs ω for $z = e^{j\omega}$.
Hint: You may consider the MATLAB function **freqz**.

(d) (15%) Write down a representation of $H(z)$ as a cascade of two second-order systems with real coefficients in your report, that is, $H(z) = H_1(z)H_2(z)$.
Hint: You may consider the MATLAB function **zp2sos**.

(e) (20%) Use the MATLAB function **plot** to plot the magnitude response of each system in (d), i.e., $H_1(z)$ vs ω and $H_2(z)$ vs ω, for $z = e^{j\omega}$. Furthermore, directly plot the multiplication result of the magnitude response $|H_1(z)|$ and $|H_2(z)|$. Compare the result with (c) in your report.

(f) (20%) Use the MATLAB function **filter** to find the real $y[n]$ when an input $x[n] = \delta[n]$ is passed through the system $H(z)$. Then, use the MATLAB function **stem** to plot the impulse response $y[n]$ vs n for $n = 0 \sim 20$, and compare it with the result in (b).

Note: It would be better to show the 8 figures from (a)(b)(c)(e)(f) in 8 sub-figures, which are integrated into one figure. For example, you can use the MATLAB function **subplot** in your **myztran.m** file.

3. NTU COOL Submission

- Please upload a compressed file (.zip), which includes your **m-file** (saved as **myztran.m**) and a **word file** (saved as **report.doc**). Please show the figures mentioned above in the word file (report.doc) and answer the questions.
- The compressed file should be named as **ID_MATLAB4.zip**.
  (e.g., B09901xxx_MATLAB4.zip)