EE402 Mini Project 2

M. Mert Ankarali[*]
Department of Electrical and Electronics Engineering
Middle East Technical University

Due: 5-November-2018, @16:40 (There will be a box to drop the Mini Projects in front of D-226. The box will be removed after 16:40.)

Important: In this mini project, there are four problems. In problems 2 and 3, you need to perform some computations in MATLAB and plot some results. In problem 4, you will need to build a Simulink model and perform some tests. You should provide all of your source codes, Simulink models, and results with your submission.

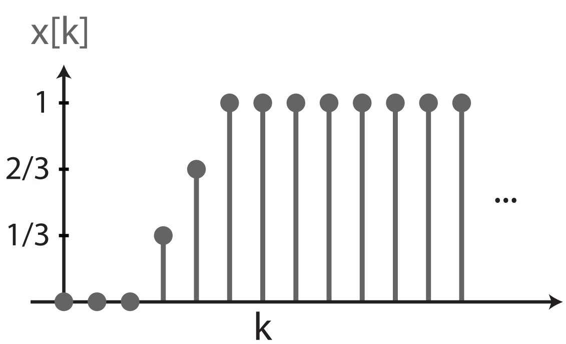
1. (25 Points) Find the Z-transform of the following discrete-time signals (given that $x[k] = 0$ for $k < 0$ for all parts.) or continuous time transfer functions ($\mathcal{Z}\{X(s)\}$).

   (a) $x[k] = 9k\,2^k - 4^k + 3$

   (b) $x[k] = \sum\limits_{h=0}^{k} a^h$, where a is a constant

   (c) $x[k] = k(k-1)\ldots(k-h+1)a^{k-h}$ (Answer is already given in the tables of the textbook. Your goal is to derive the answer in this question)

   (d) $x[k]$ given by the following graph



   (e) $X(s) = \frac{1-e^{Ts}}{s} \frac{1}{(s+a)^2}$

2. (30 Points) In this problem your goal is taking the inverse Z-transform of the following z-domain function using 4 different methods and comparing all solutions in MATLAB.

$$X(z) = \frac{z^{-1}}{(1-z^{-1})\left(1+1.3z^{-1}+0.4z^{-2}\right)^2}$$

   (a) Find a closed form expression of the inverse Z-transform of $X(z)$ using partial fraction expansion.

   (b) In this part, your goal is to find the inverse Z transform by long division method. However, unlike the example in the class (or examples in textbook) you will not perform the long-division by hand. Instead, you are supposed to write a MATLAB code that performs the long-division iteratively and finds the inverse Z-transform of $X(z)$ until $k = 20$.

(c) Inside the Lecture Notes 3, we covered a method of inverse Z-transform which treats the $X(z)$ as a "system", then transforms this synthetic system to the difference equation form, and finally the inverse Z transform is found by "simulating" this new difference equation form.

- Following the procedure detailed in the Lecture notes 3, find the necessary difference equation form.
- Write a custom code that "simulates" the difference equation (with proper input) until $k = 20$. The output of this simulation should give $x[k] = \mathcal{Z}^{-1}\{X(z)\}$

(d) MATLAB has built-in toolboxes (Control System Toolbox, Signal Processing Toolbox, Symbolic Math toolbox, etc.) that can compute the inverse Z-transform of a function using different approaches (One of them is explained in the textbook). Compute/find the inverse Z-transform of $X(z)$ using one of the built-in methods of MATLAB.

(e) (25 Points) In the parts a-d of this problem, you solved/computed the inverse Z-transform of the same $X(z)$. In MATLAB, plot each result on the same figure, by using *subplot* feature in order to compare the results. Use same axis limits for all sub-plots. Please appropriately label each plot and clearly indicate the type of the method. I recommend you to solve the MATLAB solutions of all parts on the same m-file.

3. In this problem, your goal is to solve the following difference equation using 3 different methods.

$$y[k] - 7y[k-1] + 12y[k-2] = x[k] \ , \ y[k] = 0 \ , \ \text{for } k < 0$$
$$x[k] = 5\,2^k + k \ , \ x[k] = 0 \ , \ \text{for } k < 0$$

(a) Solve the difference equation in time domain (express the solution as sum of two parts: homogeneous and particular) and find a closed form expression.

(b) Solve the difference equation using Z-transform and find a closed form expression.

(c) Simulate the difference equation in MATLAB until $k = 20$. You can either implement your own custom code, or use one of the built-in features of MATLAB.

(d) Plot the $y[k]$ in MATLAB for the solutions of all parts and compere the results.

4. (25 Points) Consider the following transfer function of a discrete time controller/filter.

$$G(z) = \frac{4 - 0.02z^{-1} + 0.3z^{-2} + 0.02z^{-3}}{2 - 6z^{-1} + 2z^{-2} - 8z^{-3} + 0.2z^{-4}}$$

- Find a minimal realization of the discrete time transfer function.
- In Simulink, build the block diagram realization (use *delay block* and other necessary elements) and connect a sinusoidal input source to the input terminal and connect a scope to the output terminal.
- On the same model page, implement the transfer function now using the *Discrete Filter* block and connect the input of this block to the same input source, and connect the output to the same scope.
- Simulate the Simulink model(s) and compare the outputs.
- Perform the simulation with different input frequencies and comment on the behavior of the frequency response of the model(s).