MATLAB Exercise 1 – LTI System Analysis and Discrete–Time Fourier Transform

Digital Signal Processing
Sommersemester 2022
S. Doclo, D. Fejgin

22.06.2022

The aim of this session is to use MATLAB for exploring LTI–system characteristics (e.g., impulse response, transfer function, stability) in the time, frequency, and z-transform domains. Most examples for this session have already been solved by hand in the previous exercise sessions. The following table shows the required MATLAB functions for this session. To obtain more detailed information about these functions and its usage, either type `help <function-name>` in the command window, e.g. `help linearSystemAnalyzer` or consider online resources as Stack Overflow or Mathwork's forum "MATLAB Answers".

command	description
`linearSystemAnalyzer`	interactive user interface for analyzing impulse response, step response, pole–zero map, frequency response, etc. (please don't use the old function `ltiviewer`)
`tf`	creates a transfer function model
`impulse`	calculates and plots impulse response
`pzmap`	calculates and plots a pole–zero map
`zplane`	plots a pole–zero map
`freqz`	calculates and plots frequency response
`conv`	calculates linear convolution
`cconv`	calculates N–point circular convolution
`fft`	calculates Discrete Fourier Transform using a Fast Fourier Transform algorithm
`fftshift`	shifts zero–frequency component to center of spectrum.
`stem`	plots discrete sequence data

Given the following difference equation

$$b_0 y[n] + \ldots + b_K y[n-K] = a_0 x[n] + \ldots + a_N x[n-N],$$

with $x[n]$ the input signal and $y[n]$ the output signal. The transfer function of the corresponding LTI system can be expressed as

$$H(z) = \frac{Y(z)}{X(z)} = \frac{a_0 + a_1 z^{-1} + \ldots + a_N z^{-N}}{b_0 + b_1 z^{-1} + \ldots + b_K z^{-K}}.$$

The function `linearSystemAnalyzer('plottype',sys)` opens the LTI viewer for the LTI system `sys` enabling to plot impulse response, step response, pole-zero map, frequency response, etc. Before you can perform the analysis, the LTI system needs to be defined in the MATLAB workspace, e.g. using `sys = tf(a, b, T, 'variable', 'z^-1')`, where `a` and `b` are vectors containing the coefficients of the numerator and the denominator of the transfer function, respectively, and `T` is the sampling period (in seconds).

System analysis

In this section system analysis is considered. Consider the following tasks to verify the pen&paper results from the lecture, save computation time and to deepen your understanding.

Question no. 1

Consider the difference equation (from course slide 97:)

$$2y[n] - y[n-1] = x[n] - 3x[n-1] + 2x[n-2]$$

- Create the corresponding LTI system using `tf`
- Plot the pole–zero map (using `linearSystemAnalyzer` or `pzmap`). Is the system stable? Is this an FIR or an IIR filter?
- Plot the magnitude and the phase responses (using `linearSystemAnalyzer` or `freqz`).
- Plot the impulse response h (using `linearSystemAnalyzer` or `impulse`)
- Plot the response $y[n]$ of this system to the input signal $x[n] = \cos(2n)$ using `conv(x,h)` with different input signal lengths, i.e. the sample index is in the range $n = [0, 10]$, $n = [0, 20]$, $n = [0, 50]$, and $n = [0, 100]$. Compare the responses with the solution derived during the lecture. Explain why different input signal lengths lead to different responses. Hint: calculate the impulse response required in `conv(x,h)` using `h=impulse(sys)`.

Question no. 2

Consider the difference equation (from exercise session no. 3, question 1)

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

- Create the corresponding LTI system using `tf`
- Plot the pole–zero map (using `linearSystemAnalyzer` or `pzmap`). Is the system stable? Is this an FIR or an IIR filter?
- Plot the magnitude and the phase responses (using `linearSystemAnalyzer` or `freqz`).
- Plot the impulse response h (using `linearSystemAnalyzer` or `impulse`)

Question no. 3

Consider the difference equation $y[n] = x[n] + \alpha x[n-1]$ with $-1 \leq \alpha \leq 1$

- Set $\alpha = 0.9$ and create the corresponding `sys` in the MATLAB workspace. Is this an FIR or an IIR filter?
- Plot the pole–zero map (using `linearSystemAnalyzer` or `pzmap`). Is the system stable?
- Plot the magnitude and the phase responses (using `linearSystemAnalyzer` or `freqz`). What kind of filter (lowpass/highpass/bandpass) is this?
- Determine in which α range the filter is stable (using `linearSystemAnalyzer` or `pzmap`).
- Explore and explain the behavior of the filter (in terms of lowpass, highpass or bandpass filtering) across different values of α within the range $-1 \leq \alpha \leq 1$ (using `linearSystemAnalyzer` or `freqz`).

Question no. 4

Consider the difference equation $x[n] = y[n] + \alpha y[n-1]$ with $-1 \leq \alpha \leq 1$

- Set $\alpha = 0.9$ and create the corresponding `sys` in the MATLAB workspace. Is this an FIR or an IIR filter?
- Determine in which α range the filter is stable (using `linearSystemAnalyzer` or `pzmap`).
- Plot the pole–zero map (using `linearSystemAnalyzer` or `pzmap`). Is the system stable?
- Plot the magnitude and the phase responses (using `linearSystemAnalyzer` or `freqz`). What kind of filter (lowpass/highpass/bandpass) is this?
- Explore and explain the behavior of the filter (in terms of lowpass, highpass or bandpass filtering) across different values of α within the range $-1 \leq \alpha \leq 1$ (using `linearSystemAnalyzer` or `freqz`).

Z–transform, DTFT, DFT

The object of this section is to get familiar with the concepts of Z–transform, discrete-time Fourier transform (DTFT), discrete Fourier transform (DFT) and linear and circular convolution in terms of MATLAB. Consider the following tasks to verify the pen&paper results from the lecture, save computation time and to deepen your understanding.

Question no. 5

Consider the filters

$$H_1(z) = 2 + 3z^{-1} + 4z^{-2}; \quad H_2(z) = 3 + 4z^{-1} + 5z^{-2} + 6z^{-3};$$

Determine the cascaded filter $H(z) = H_1(z)H_2(z)$ using `conv`. Check by explicitly multiplying both filters with each other.

Question no. 6

Consider the discrete–time signal (from course slide 127)

$$x[n] = \mathrm{sinc}^2(2n)$$

- Plot the DTFT $X\left(e^{j\Omega}\right)$. Hint: since the DTFT computation requires to consider the discrete-time signal $x[n]$ over the range $n = [-\infty, \infty]$ and it is not implementable using `fft` function, consider a range where $x[n]$ converges nearly to zero, e.g., $n = [-50, 50]$
- Plot the DTFT of the discrete-time signal $y[n]$ obtained by inserting one 0 in between each sample of $x[n]$ (i.e. time–expansion)
- Plot the DTFT (amplitude only) of the discrete-time signal $z[n] = e^{jn}x[n]$

Question no. 7

Given the continuous time (CT) signal

$$x(t) = \cos(2\pi f_0 t) + 5\cos(2\pi f_1 t); \text{ where } f_0 = 2000\,\text{Hz and} f_1 = 2f_0$$

- What is the CTFT? Consider also its symmetry.

- Discretize the CT–signal (i.e. convert $x(t)$ to $x[n]$) and plot the mixture. Use an appropriate sampling rate (What is the minimum required sampling rate to avoid aliasing?)
- Compute the DFT $X[k]$ using `fft(x,nfft)`, where `nfft` is the number of frequency bins. Plot the magnitude of this spectrum (what is the maximum detectable frequency? What is the spacing of the frequency axis?). Investigate the effect of `nfft` and the sampling rate f_s.

Question no. 8

Consider the following time–domain input signal and filter

$$\mathbf{x} = [1, 1, 1, 1]^T ; \quad \mathbf{w} = [1, -1, 1]^T$$

- Plot the linear convolution output using `conv(x,w)`. Verify that convolution is commutative.
- Plot the 4–point circular convolution output using `cconv(x,w,n)` where n is the size of the circular convolution. Explain how to obtain a linear convolution output using `cconv`. Implement your solution.
- Consider the same input signal $\mathbf{x}$ and the new filter $w = [1, 0, 0, 0]$. Plot and compare the 4–point circular convolution and linear convolution outputs. Explain why there is/is not a difference between these convolutions.