Lab #2: Implementation and Analysis of Discrete-Time Systems

Objective:

To gain experience in implementing and analyzing digital systems within MATLAB. In particular, you will look at time-domain convolution, system impulse responses, and the discrete-time Fourier transform (DTFT).

Assessment:

– Your grade for this lab will be based on your ability to create and work with digital systems within MATLAB, and on your reporting of the results. The report should contain any mathematical calculations or derivations carried out, MATLAB plots of results with brief descriptions, the MATLAB code with which you obtained your results, and answers to specific questions below.

– Clearly label all plots and their axes (points for style will be deducted otherwise)

– DON'T COPY OTHERS BLINDLY!!

– Please attend the lab section to which you have been assigned.

– You should complete this lab with one lab partner. If there are an odd number of students, then one group of three will be created by the TA.

– Each pair of students should complete one lab report together, which is to be submitted <u>one week</u> from the date of the lab. Those students in the "At Home" section must submit their reports one week from the day on which they demonstrated their lab to Jeff Bondy.

Pre-lab:

– Carefully read through this lab description, so that you know what is required.

– Read through the lecture notes (and bring them with you) so that you know how to answer the questions.

– Familiarize yourself with the MATLAB commands that may be required for this lab – see the list at the end of this lab description for some hints.

1. Introduction to convolution:

a. Create the discrete-time sequence:

$$x[n] = u[n] - u[n-10],$$

where $u[n]$ is the Heaviside step function. You do not have to zero-pad $x[n]$ (i.e., the vector you have should contain no zero elements).

b. Now, convolve $x[n]$ over and over:

$$a[n] = x[n] * x[n],$$
$$b[n] = a[n] * x[n],$$
$$c[n] = b[n] * x[n],$$
$$d[n] = c[n] * x[n].$$

c. Plot $a[n]$, $b[n]$, $c[n]$ and $d[n]$ using the `stem()` function.

d. Fit a polynomial to the first 10 samples of $a[n]$, $b[n]$, $c[n]$ and $d[n]$. What order of polynomial fits the first 10 values to greater than 10^{-10} precision for each of the signals? Explain why using the analytic form of convolution.

2. Convolution of signals and system impulse responses:

a. Load the supplied acoustic impulse response of a room into MATLAB using the command:

```
[impr,fs,nbits] = wavread('impr.wav');
```

This impulse response was obtained by creating an impulsive noise at one position in the room and recording (and digitizing) the sounds arriving at another position in the room.

b. Plot the impulse-response waveform `impr` using the `plot()` command and listen to it using the `soundsc()` command. What can you see and hear in the impulse response?

c. Load the supplied speech signal into MATLAB using the command:

```
[y,fs,nbits] = wavread('oilyrag.wav');
```

d. Convolve the speech signal with the impulse response, and plot and listen to the resulting signal. Describe what you see and hear, comparing it to the original speech signal `y`. Explain what the convolved signal is physically equivalent to, according to the impulse-response theory presented in Lecture #5.

3. Introduction to filtering:

a. Consider the two impulse responses, $h_1[n]$ and $h_2[n]$:

$$h_1[n] = \tfrac{1}{4}\delta[n] + \tfrac{1}{2}\delta[n-1] + \tfrac{1}{4}\delta[n-2],$$
$$h_2[n] = -\tfrac{1}{4}\delta[n] + \tfrac{1}{2}\delta[n-1] - \tfrac{1}{4}\delta[n-2],$$

and the six discrete-time sinusoidal signals $x_{a\text{-}f}[n] = 5\cos(\omega_{a\text{-}f}n)$ for $n = 0\ldots50$, with $\omega_{a\text{-}f} = 0$, $\pi/5$, $2\pi/5$, $3\pi/5$, $4\pi/5$, and π radians, respectively.

b. Calculate the energy in each of the signals a through f by taking the root mean square (RMS) of each of them:

$$\text{RMS}_{a-f} = \sqrt{\frac{1}{N}\sum_{n=1}^{N}\left(x_{a-f}[n]\right)^2}\;.$$

c. Now, convolve $h_1[n]$ with each of the signals $x_{a\text{-}f}[n]$, and calculate the resulting RMS values of the six different *output* signals. Calculate the difference between output and input RMS values for each of the signals in terms of the system *gain* measured in deciBels (dB):

$$\text{gain} = 20\log_{10}\left(\text{RMS}_{\text{out}}\middle/\text{RMS}_{\text{in}}\right)\;\text{dB}\,.$$

Plot the gain for each signal $x_{a\text{-}f}[n]$ as a function of the signal frequency $\omega_{a\text{-}f}$.

d. Repeat part c but with the impulse response $h_2[n]$.

e. What are the differences between the two plots? What are the differences between $h_1[n]$ and $h_2[n]$ in terms of the frequencies of the input signals they pass?

4. The Discrete-Time Fourier Transform (DTFT)

a. Create a MATLAB function to calculate the DTFT for a given input signal vector x and a given discrete-time frequency vector w. For the sample index n, create the vector:

```
n = -ceil((length(x)-1)/2):floor((length(x)-1)/2);
```

b. Using your function from part a, calculate the DTFTs of the impulse responses $h_1[n]$ and $h_2[n]$ from section 3 above, for frequencies w covering the range $[-3\pi, 3\pi]$. Are the resulting DTFTs real or complex valued? What should they be?

c. Plot the magnitude of each of the DTFTs from part b versus frequency w. Do you see periodicity in these spectra? If so, what is the period and why?

d. How do the two plots from part c explain the gain versus frequency plots from section 3 above?

5. Filtering white Gaussian noise

a. Create a signal 1000 samples long of white Gaussian noise. Convolve this signal with $h_1[n]$ and $h_2[n]$ from section 3 above, then compute the DTFT for each of the convolved signals using the function you created in section 4 above.

b. Plot the magnitude of each of the DTFTs from part a versus frequency w. Do the results agree with those of sections 3 and 4 above? Explain.

Potentially useful MATLAB commands

Note that this is not an exhaustive list! You are not required to incorporate all of these in your scripts.

help <topic>	helpwin	figure	plot	stem
hist	subplot	hold on	xlabel	ylabel
legend	title	function	clear	close
clc	zeros	ones	cos	exp
abs	conv	sum	repmat	sqrt
mean	log10	randn	polyfit	polyval
if	for	end	real	imag
angle	unwrap	phase	max	min
soundsc	wavread	wavwrite		