Problem set 3: Toolboxes

Handed out: Friday, October 28, 2016.

Due: 11:55pm, November 10, 2016.

You must hand in two files:

- One file, named your_name_E3.pdf, with the solutions to the exercises in this set, following the template available in the virtual campus. For every exercise, you must explain how you solved it, the necessary MATLAB code, the results obtained (run transcripts or plots), and some final comments (if any).
- The second file, named your_name_E3.zip, must contain all the MATLAB code you used to solve the exercises, organized in one or more text (.m) files, to allow the teacher to check your solution.

Please, make sure that the code in the pdf file is **exactly the same** as in the zip file. Remember that all the MATLAB code is supposed to be **entirely yours**. Otherwise, you must clearly specify which parts are not, and properly attribute them to the source (names of collaborators, links to webpages, book references,...). Read the Course Information document for more details on this subject.

Exercise 1. Curves intersections

The GPS system and the location systems based on radio base stations determine the position by the well-known technique called trilateration. It consists in calculating, by radio signals, the distances from the receiver to three satellites or base stations. Consider the 2D problem. In this case, the position is the intersection of three circumferences centred in the transmitters with radios equal to the measured distances, respectively. The intersection of the three circumferences is unique. Be the positions of the transmitters given by the matrix position = [2 1;2.5 3;1 2] and the distances by the vector distance = [1.5 1 2]. The rows of the matrix are the x-y coordinates of each transmitter. All the units are in kilometres.

Plot the three circumferences in the same graphic with **`ezplot`**. Use the convenient aspect ratio to get true circumferences.

Use the plot results to compute the exact position of the receiver and the intersections between all the different pairs of circumferences (use **`fsolve`**) choosing adequately the starting points. Plot these points in the above graphic. Put a mark at each transmitter and at the receiver position.

Exercise 2. Complex variable functions

1) Given the complex variable function $G(s)=\dfrac{3}{s(s+2)(s+3)}$, find the <u>modulus</u> and <u>argument</u> when the complex variable "s" takes the following values s=0, j, $2j$, $5j$, and ∞j (functions **`polyval`** or **`freqs`**, **`abs`**, **`angle`**).

2) Plot the Bode diagram for the frequency response $G(s)\big|_{s=j\omega}$ (function **`bode`**, [**`logspace`**]).

3) Obtain the polar plot (representation in the Nyquist plane) for the frequency response $G(s)\big|_{s=j\omega}$ (functions `logspace` and `nyquist`). Be careful with the specification of the frequencies vector (we want to see in detail the frequency response near the origin of the Nyquist plane).

Exercise 3. Time response

1) Plot the impulse response for the system $H(s) = \dfrac{2}{s^2 + 0.25s + 1}$ (function **`impulse`**).
2) Plot the step response for the system $H(s) = \dfrac{2}{s^2 + 0.25s + 1}$ (function **`step`**). What is the peak value? And the settling time value? (Use the right mouse button over the figure to obtain these two values.)

Exercise 4. Decoding Morse signal

This problem deals with the receiver for the Morse encoder analyzed in exercise 6 of problem set 2b.

1) Read the file **morse.wav** (with **audioread** function) which contains a Morse signal with the following parameters: dot_duration = 0.065sec, tone frequency = 700Hz, sampling frequency = 8000 Hz.
2) Listen the Morse signal with **sound** function and plot 4000 points of it. Make a zoom of the graphic to get inside of the signal.
3) Determine the spectral density of the signal using the FFT and confirm that the tone frequency obtained is approximately 700 Hz. (`fft, abs`, `max`)
4) Build a sinusoid (sinus) with frequency 700Hz and the same duration of Morse signal. Multiply the Morse signal by the sinusoid and then filter out the result with a filter FIR of 64 coefficients and a digital bandwidth of 5/T/fs, being T the tone_duration and fs the sampling frequency. (functions `fir1` and `filter`). Now you have the base band signal. Plot the resulting signal with the function stairs and see the ones and zeros.
5) Get the approximate derivative of the above function and plot it (MATLAB function diff). You can check that every dot and dash are always between a maximum and the next minimum with their corresponding duration. The silences are in between a minimum and the next maximum and their duration depends on if they are the end of a character, letters of words as described in problem set 2b.
6) Write a MATLAB program to decode the base band signal from its derivate using the function `local_max` to determine the maxima and minima of the derivate and the Morse code given in the supplied function.

Exercise 5. Linear Regression, Polynomial Fitting.

The file **`data5.mat`** contains two data vectors, *U* and *V*, with 1000 random samples each.

1) Compute the regression line for *U* and *V* (**`polyfit`**). Is the fit good enough? Compute a measure of the error as the norm of the difference between the real vector *V* and the vector obtained from the regression line. (**`polyval, plot, norm`**).
2) Try now a degree two polynomial. Does it significantly improve the approximation? Is it worth considering a polynomial of degree three? Compute the error as before.