Course: ELEC-E8103 Modelling, Estimation, and Dynamic Systems

Final exercise 27.11.2019

The final exercise will start from 27.11.2019 09:00 to 08.12.2019 23:59. Please submit your solutions to the final exercise section of the course page in mycourses.aalto.fi before the deadline. The final exercise should be done individually, and NO discussion is allowed. Accordingly, exercise sessions will not be held for this exercise.

Submission of the final exercise:

You should submit your solutions to the "Final Exercise" section of the MyCourses page:

https://mycourses.aalto.fi/mod/assign/view.php?id=463813

Your submission should include a single zip file named as "surname_studentNumber_Final.zip", consisting of a pdf file named "surname_studentNumber_Final.pdf", and the following MATLAB files: "problem1.m", "rhs.m", "oscillator.mdl", "constants.m", "estimate_price.m", "sys1.sid", "sys2.sid", and "sys3.sid".

The hard deadline for submission of the solutions is 08.12.2019 at 23:59.

1. **(20 points) Figure 1 shows an oscillator circuit. The circuit includes three resistors denoted by R_1, R_2, and R_3, an inductor and a capacitor represented with L and C, respectively. The circuit is powered with a power source denoted by $E(t)$. We are interested in the current of the left loop (I_1) and the charge in the capacitor. These variables can be changed by changing the power signal.**

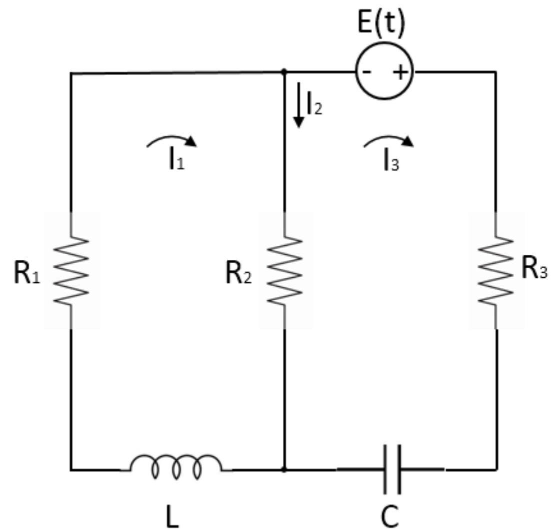

Figure 1 – RLC circuit.

a) Phase 1: Based on the description of the system, structure the problem. You are supposed to answer these questions: What signal(s) are the output(s) of the system? What signal(s) are the input(s) of the system? What are the constants of the system? What are the internal time-varying variables of the system?
(1 point)

b) Phase 2: Set up the basic equations of the system. You are supposed to derive the underlying differential equations of the system using first principles.
(5 points)

c) In this part, your task is to solve the differential equations using MATLAB ODE solvers (for instance ode45). First, create a MATLAB script named “problem1.m”. Constant initialization and plotting should be done in this script. In addition, “problem1.m” should call a function named “rhs.m”, in which the right-hand side of the equations are defined. Plot the response of the system, i.e., the evolution of the states, to the following input voltage function for 20 seconds and **include the plots in your report**:

$$E(t) = \sin(t)$$

The system parameters are

$$R_1 = 4\,\Omega$$
$$R_2 = 2\,\Omega$$
$$R_3 = 3\,\Omega$$
$$L = 1.6\text{ H}$$
$$C = 0.25\text{ F}$$

Current in the left loop at $t = 0$:

$$I_1(0) = 15\text{ A}$$

Charge in the capacitor at $t = 0$:

$$q(0) = 2\text{ A}\cdot\text{s}$$

Simulation time: $[0,20]$ seconds

(Note: The goal in this part is to solve the ODE directly without using state-space or transfer function formulation).
(7 points)

d) Simulate the system in Simulink. Save the Simulink model as "oscillator.mdl". Plot the response of the system, i.e., the evolution of the states, for the voltage input given in (c) for 20 seconds, and **include the plots in your report**. You should include all the constants used in the Simulink model in a separate file named "constants.m". After executing this file, all the necessary constants should be stored in MATLAB workspace. Save "constants.m" for submission.
(7 points)

2. **(19 points) The goal of this task is to implement a computer program that can estimate the sale price of a residential property based on a database. Suppose you are the data analyst of a consulting company, and your client is a real-estate agent. The real-estate agent provides you with a database, which has been taken from 200 properties. The properties have been sold in the recent month. The data has been stored in the data file "data2.xlsx". Your task is to design a computer program in MATLAB, that can be used by the real-estate agent, for pricing new residential properties.**

The database includes relevant information about the residential properties such as the sold price (euros), the area of living space ($\boldsymbol{m^2}$), the construction year, the number of rooms, the floor number, and the location of the house defined by $\boldsymbol{x}$ and $\boldsymbol{y}$ coordinates ($\boldsymbol{km}$). Your program should take the parameters of a new residential property and suggests a price to the real-estate agent.

Your program should be in the format of a function named estimate_price.m as below,

function price_house = estimate_price(house_params)

where,

$price\_house$ (scalar) is the estimated price for a test house;

$house\_params$ (vector) includes the specifications of the test house that you want to estimate its price. For example, $house\_params$ should be in the following format: $house\_params = [45\ 1978\ 1\ 1\ 0.2\ 0.3]$ for the first residential property in Table 1.

Note that the downtown of the city is located at the following location,

$$\boldsymbol{P_{Downtown}} = \left| \begin{array}{l} \boldsymbol{x = 1.43\ (km)} \\ \boldsymbol{y = 0.63\ (km)} \end{array} \right.$$

The downtown is considered to be the expensive district of the city (Like the neighborhood of Kamppi in Helsinki). Being close to the downtown is generally considered as a positive point for a property.

Estimate the price for the following test cases (Table 1) by your program, and **include the estimated prices in your final report**.

Table 1. Residential properties for testing your algorithm

	Living area (m^2)	Construction year	Number of rooms	Floor number	x (km)	y (km)
House No. 1	45	1978	1	1	0.2	0.3
House No. 2	56	2000	2	2	0.6	1.6
House No. 3	72	2016	3	6	1.4	0.65

Your solution should have the following substances:

Accuracy: your program should estimate the price of the residential properties reasonably. We will test your program with our standard solution, and the provided test cases in this document.

Validity: you are supposed to validate your solution by means of data and appropriate methods, which were discussed in the lectures 5 and 6 of the course. **In the report, you should explain your validation procedure in sufficient detail, and by including appropriate plots and measures. The code, which produces the relevant plot and measures, should be implemented inside the aforementioned function.**

The points for this task will be given based on the **accuracy of your estimations** for the test cases, as well as your **justification of the solution**.

3. **(21 points) The goal of this question is to identify dynamic systems using input-output data. Copy the files "runExam.m", and "modeldata.mat" in your MATLAB current folder. Run the "runExam.m" script.**

   The following text should appear in your command window.

   >> runExam

   Please Enter the numeric part of your student number!:

   Now, you should type your student number and press Enter. If your student number ends with an alphabetical letter, you should just type the numerical part of your student number, e.g. if your student number is 12345W, you should type 12345.

   Then your data will be stored to MATLAB workspace. Data has been collected from three different dynamic systems. Input-output datasets are (u_1,y_1), (u_2,y_2), and (u_3,y_3). The sampling frequency for all of the datasets has been 1 Hz. Identify polynomial models of the mentioned systems using MATLAB System Identification Toolbox. You should explicitly select a model as your final answer for each input-output dataset. You are supposed to answer the following question for each system.

   a) What is your selected model structure?

   b) Present the resulted plots and information related to validation procedure **only for your selected model structure**. (**Hint**: For instance residual analysis plots, poles and zeros plot, variance analysis information, etc.)

   c) What are the alternative system(s) for the data if you think there are any?

   Save the final identification session for each dynamic system as "sys1.sid" for (u_1,y_1), "sys2.sid" for (u_2,y_2), and "sys3.sid" for (u_3,y_3). **Include a snapshot of your final sid files in your report.**

   The points will be given based on the resulted model structures and their orders, as well as the identification path.

   (7 points for the identification of each system, 21 points in total)