

System Identification – Practical Assignment 2

Transient Analysis of Step Responses

Logistics

This practical assignment should very preferably be carried out by each student separately. Otherwise, if there are more students than computers, students may team up in groups of 2.

The assignment solution consists of Matlab code. Develop this code in a single Matlab script.

The overall rules for labs are described on the website. For each particular lab, your attendance will only be registered if you have a working, original solution. The teacher will check that your code **works** during the lab class. Only after this has been done, for the **originality** check, upload your solution here:

<https://www.dropbox.com/request/kgRPgy3lzhkgjngCI6a>

Upload *only once*, a single *m-file*, named exactly according to the following pattern:

`2_EN_G.H.LastnameFirstname.m`

where G is your group, H your halfgroup, and your last and first names follow. E.g., `2_EN_3.1_PopAlex.m`.

Any files that are duplicate, nonstandard, inappropriately named, or corresponding to unchecked solutions will not be considered. Files will be automatically tested for plagiarism, and any solution that fails this test will be marked copied; only solutions that pass both the working and the originality test are definitively validated. Therefore, while you are encouraged to discuss ideas and algorithms amongst colleagues, sharing and borrowing pieces of code is forbidden.

Assignment description

In this assignment we will perform transient analysis of step response models – see the course material, Part II: *Step and Impulse Response Graphical Models*. We will do this for both first-order and second-order systems.

Each student is assigned an index number by the teacher. Then, the student downloads the Matlab data files that form the basis of the assignment from the course webpage. There are two files: the first contains several step inputs signals and the response of a first-order system, and the second contains similar data for a second-order system. The data is provided as an object called `data` of type `iddata` from the system identification toolbox, see `help iddata`. For convenience, a separate variable `t` holds the time vector of the experiment. Every dataset contains *five* consecutive step signals, each corresponding to 100 time steps; see the figure. The initial conditions are zero. The *first* step should be used for identification; the second step simply brings the system back to zero initial conditions, so that you do not have trouble with initial output mismatches due to nonzero initial conditions; and *the last three steps* should be used for validation.

Work on the following requirements first for the first-order system, and then for the second-order one:

- Develop a transfer function model of the system with the method described in Lecture 2, using the first step signal and response from the data. Include instructions that output to the console the transfer function, as well as relevant intermediate values when your script is run (e.g. gain K , time constant T for first-order systems, overshoot M and oscillation period T_0 for second order systems).

- Validate your model using steps three to five (this is the validation data). The validation should consist of: (a) a plot where the system output is compared with the model output on the same graph; (b) and the computation of the MSE. Both of these results should be automatically produced by the Matlab code you provide. Use Matlab function `lsim` to simulate the system response to the validation input.

Some relevant Matlab functions: `load`, `tf`, `lsim`, `plot`.

