

# FRT 041 System Identification

## Laboratory Exercise 2

Lennart Andersson  
Ulf Jönsson  
Revised: Johan Bengtsson  
Brad Schofield

Department of Automatic Control  
Lund Institute of Technology  
Lund University  
October 1995

Revised: January 2006

# 1. Introduction

In this laboratory exercise you will learn how to use Matlab's System Identification Toolbox. The toolbox provides a way to obtain discrete time models based on discrete time data.

## 1.1 Preparation

Read "Manual for Practical System Identification" in the appendix. This manual can also be useful during the projects. Section 6–10 are most important for the exercises in this lab. We assume that you have some experience from using Matlab. Work through the exercise below.

**Exercise** Write a sequence of Matlab commands, which solves the following tasks. Assume that you have the data sequence  $z_i = \text{iddata}(y_i, u_i, h)$  for identification and another data sequence  $z_v = \text{iddata}(y_v, u_v, h)$  for validation.

1. Estimate the impulse response by use of correlation analysis and plot the step response.
2. Estimate the transfer function by use of spectrum analysis. Plot the Bode diagram.
3. Estimate the parameters of a Box Jenkins model with model order  $n_B = 2, n_C = 2, n_D = 2, n_F = 3$  and  $n_k = 1$ .
4. Compute and plot the residuals for the Box-Jenkins model obtained in step 3.
5. Plot the Bode diagram and the noise spectrum for the Box Jenkins model obtained in step 3.
6. Do a crossvalidation for the Box-Jenkins model obtained in step 3.

## 2. Process specification

The process in this laboratory exercise is a simulation model of a real process with artificial measurement noise and a load disturbance. For the real process the *input signal and output signal must be between  $\pm 10V$* . To get realistic results you should make sure that this is satisfied during your simulations !

## 3. Identification

The purpose of this section is that you should go through the steps of system identification for an unknown process. The result will be a mathematical model of the process.

Login at the computer and start Matlab `matlab-R12.1 -nodesktop`.

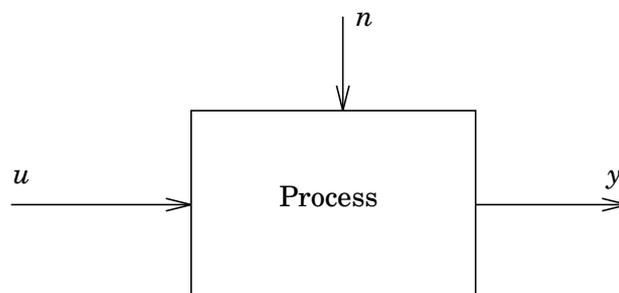
Download the file `simu.m` from the course webpage, and put it in your working directory. This file will be used to simulate the unknown process.

Use the Matlab command `help` frequently during the whole exercise !

Most of the time should be spent on obtaining parametric models and validation. Suggested timeplan:

- Experiments 60min
- Spectrum analysis 45min
- Parametric models and Validation 105min

### 3.1 Exercise 1



**Figure 1** Black box description of the process

Figure 1 shows the available process as a black box. The influence from disturbances is illustrated by the signal  $n$ . The process can be simulated using a special matlab command `simu`.

```
[y,u,t]=simu(u,h);
```

```
u process input  
h sampling period  
y process output  
t time vector
```

Simulate the process with zero input. A zero input vector can be created using

```
u=zeros(N,1);
```

N number of data points

Use the matlab command plot to display the simulation result. Explain the result.

### 3.2 Exercise 2

Do some simple experiments such as step responses to gain some basic information about the process.

A step input vector can be created using

```
u=[zeros(N1,1);A*ones(N2,1)];
```

N1 number of samples with zero input

N2 number of samples with input A

A amplitude

Check that the input signal and output signal are between  $\pm 10$  ! Determine dominant time constants and the stationary gain. Does the process have continuous time minimum phase character?

Is the process linear and time invariant?

### 3.3 Exercise 3

The choice of input signal is crucial in system identification. Here we will use a Pseudo-Random Binary Sequence (PRBS) input signal. A PRBS signal can be obtained as

```
prbs=A*kron(rand(N,1)>0.5,ones(n,1));
```

A Amplitude

N sequence length

n samples during each state

prbs column vector with  $n*N$  elements

where you have to choose  $A, N$  and  $n$ .

Generate identification data by simulating the system with a PRBS input.

Examine the data and prepare them for identification. How should the data be divided? It may be useful to use `iddata` objects to store your identification data.

### **3.4 Exercise 4**

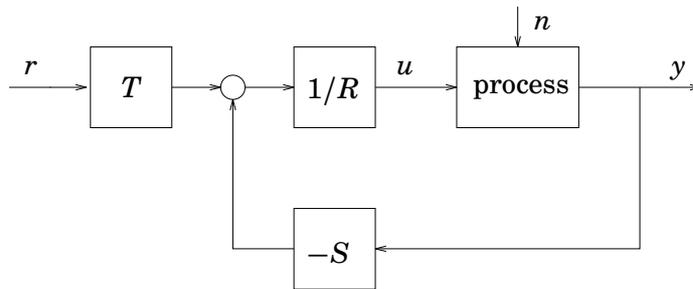
Use spectral analysis—e.g., `spa` or `spectrum` on the gathered data. What is the effect of experiment duration on the spectral estimates? Investigate the use of ‘windowing’ to improve the results. If desired, the Identification toolbox GUI may be used in this exercise. It is started by typing `ident` in Matlab.

### **3.5 Exercise 5**

Estimate some parametric models. This step contains both model structure selection, model order selection and validation. This exercise is the most difficult and time consuming! Therefore, spend most of the time here. It may be useful to use the Identification toolbox GUI in this exercise as well.

## **4. Controller design**

Based on the parametric model obtained by identification you should now design a controller.



**Figure 2** Closed-loop system

#### 4.1 Exercise 6

Design a controller and simulate the closed-loop system. Write help ppbox and in particular help rstd to see how the design can be performed (you may need to type oldboxes to access these functions). The design should satisfy the signal specification stated earlier, i.e., the input and output should be between  $\pm 10V$ . The total structure can be seen in Figure 2. The closed loop can be simulated using the command

```
[y,uc,t]=simu(yr,h,R,S,T);
```

```
yr    reference value
h     sampling period
R,S,T controller polynomials
y     process output
uc    process input
t     time vector
```