



EE – Automatic Control

Control Theory and Practice
Advanced Course

Computer Exercise:
MULTIVARIABLE SYSTEMS

Anders Hansson, Per Bodin och Elling Jacobsen, December 1998

1 Introduction

In this computer exercise we will investigate properties of multivariable systems. The application is a model of a four-tank-process. In particular we will consider pairings between different input and output signals and non-minimum phase dynamics. The pairings are analyzed using RGA and we will investigate their influence in decentralized PI control.

Preparations: Chapters 3.3, 3.5, 6.5, 7.3-7.5, 7.7 and 8.3 in Glad, T. and Ljung, L.: “Control theory—multivariable and nonlinear methods”.

Presentation: All problems in this exercise should be solved, but only the tasks on the report form should be handed in. The report form and the date when it should be handed in can be found on the course website. The exercise should be performed in pairs of students.

2 Theoretical overview

This section contains a short overview of basic theory for multivariable systems. The content is based on the course book.

2.1 Poles and zeros

As for SISO systems we can define poles and zeros for a linear MIMO system with transfer matrix $G(s)$.

The poles of $G(s)$ are defined as the eigenvalues of the system matrix A in a minimal state space realization of the system and they are calculated as the roots of the *pole polynomial*, $\det(sI - A)$. The pole polynomial can also be obtained by calculating the least common denominator of all sub-determinants of $G(s)$.

It is more difficult to extend the definition of zeros from the SISO case to the MIMO case. The most common definition of a zero of the system is a value of s where the transfer matrix $G(s)$ loses rank. For the special case of square transfer matrices, the zeros are given by the roots of $\det G(s) = 0$.

2.2 Singular values, directions and H_∞ norms

The singular values σ_i of a matrix A are defined as $\sigma_i = \sqrt{\lambda_i}$, where λ_i are the eigenvalues of the matrix A^*A . The largest singular value of A is denoted $\bar{\sigma}(A)$, and the smallest as $\underline{\sigma}(A)$. If $y = Ax$, it follows from the singular value definition that

$$\underline{\sigma}(A) \leq \frac{|y|}{|x|} \leq \bar{\sigma}(A)$$

where the relation between the norm of y and the norm of x , $|y|/|x|$, can be interpreted as the gain of the matrix A . If x is parallel with the eigenvector corresponding to

the largest eigenvalue of $A^T A$ then we have $|y| = \bar{\sigma}(A)|x|$, and if x is parallel to the eigenvector corresponding to the smallest then we obtain $|y| = \underline{\sigma}(A)|x|$. This way we can define directions corresponding to the largest and smallest singular value for A respectively.

For a linear stable multivariable system with transfer matrix $G(s)$ we have

$$Y(i\omega) = G(i\omega)U(i\omega)$$

where $Y(s)$ and $U(s)$ are the Laplace transforms of the system's output and input signals. According to the definition of singular values we therefore have

$$\underline{\sigma}(G(i\omega)) \leq \frac{|Y(i\omega)|}{|U(i\omega)|} \leq \bar{\sigma}(G(i\omega))$$

Introducing

$$|G(i\omega)| := \bar{\sigma}(G(i\omega))$$

it holds

$$|Y(i\omega)| \leq |G(i\omega)||U(i\omega)|.$$

This notation is analogous to the SISO case where the norm is interpreted as the absolute value of $G(i\omega)$, and the inequality above turns to an equality.

To understand the inequalities in the MIMO case we can choose the input parallel to the direction corresponding to the largest or smallest singular values of $G(i\omega)$. The directions decide the “mix” of the input signal components that results in the largest and smallest gain of the system respectively.

The largest gain of the multivariable system $G(i\omega)$ is denoted $\|G\|_\infty$, which is given by

$$\|G\|_\infty = \max_{\omega} |G(i\omega)|.$$

$\|G\|_\infty$ is called the H_∞ norm of G . For output signal $y(t)$ and input signal $u(t)$ it holds

$$\|y\| \leq \|G\|_\infty \|u\|.$$

Therefore the H_∞ norm can be interpreted as the time domain gain of the system. It holds

$$\sup_u \frac{\|y\|}{\|u\|} = \|G\|_\infty,$$

which can be seen as the definition of the H_∞ norm for G .

2.3 Decentralized control

A fundamental problem in multivariable control is the pairings between the inputs and outputs. This means that if one input changes there is generally a change in all outputs. A measure of the strength of the pairings in a multivariable system, $G(s)$, is given by the *Relative Gain Array*, RGA of the transfer matrix G . It is defined by

$$\text{RGA}(G(i\omega)) := G(i\omega) .* [G^{-1}(i\omega)]^T,$$

where “.*” denotes element wise multiplication. We can use RGA to determine which input and output that are suitable to pair in a decentralized controller. Two rules of thumb:

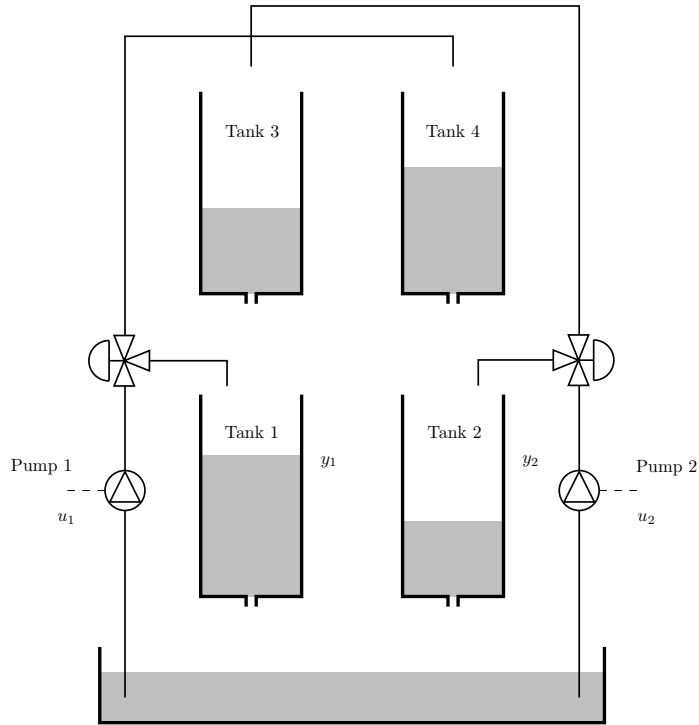


Figure 1: The four-tank process

1. Find a pairing such that diagonal elements of $\text{RGA}(G(i\omega_c))$ are as close to 1 as possible, where ω_c is the intended cross-over frequency.
2. Avoid pairings which correspond to negative elements in $\text{RGA}(G(0))$.

3 Exercises

In this computer exercise linear models of a four-tank process will be investigated. The system is shown schematically in Figure 1. The input signals are the voltages of the pumps, u_1 and u_2 . The output signals that we want to control are the levels in the lower tanks, y_1 and y_2 . Connected to each pump there is a valve that divides the water to the upper and lower tanks. A linear multivariable model with 2 inputs and 2 outputs is given by $Y(s) = G(s)U(s)$ where

$$Y(s) = \begin{bmatrix} Y_1(s) \\ Y_2(s) \end{bmatrix}, \quad U(s) = \begin{bmatrix} U_1(s) \\ U_2(s) \end{bmatrix} \quad \text{and} \quad G(s) = \begin{bmatrix} g_{11}(s) & g_{12}(s) \\ g_{21}(s) & g_{22}(s) \end{bmatrix}$$

Depending on the settings of the valves we obtain different $G(s)$. In this computer exercise two different valve settings will be investigated: in the first, most of the water will go directly to the lower tanks and $G(s)$ is minimum phase; in the second, most of the water go to the upper tanks and $G(s)$ will be non-minimum phase.

3.1 Poles, zeros and RGA

A linear state space model for the four-tank process is generated by the MATLAB functions `minphase` and `nonminphase`. To put the minimum phase model in the variable `sysmp` we write

```
sysmp = minphase
```

The following MATLAB functions can be used in this exercise. The poles of the system `sys` are obtained with

```
pole(sys)
```

and its zeros with

```
tzero(sys)
```

The system should be given as a (minimal) state space description when these functions are used. (Otherwise numerical problems can appear in MATLAB.)

The singular values for a system are calculated with `sigma`. To extract system matrices and transfer functions the functions `ssdata` and `tfdata` are used, respectively. The singular value decomposition are calculated using `svd`. To calculate step responses for linear systems the function `step` can be used. The Bode diagram of a system is plotted using `bode`. Note that dB scale is used. Use the MATLAB Help to learn more about the functions.

The problems below should be solved both for the minimum phase and non-minimum phase case. It is suitable to start with the former.

Exercise 3.1.1. Calculate the transfer matrix $G(s)$. Investigate each element of the matrix (Hint: `G(1,1)` extracts element (1,1)). Calculate the poles and zeros of the elements.

.....
.....

Exercise 3.1.2. Calculate the poles and zeros of the multivariable system. Do these imply any constraint on the achievable control performance?

.....
.....

Exercise 3.1.3. Investigate the largest and smallest singular values for the system at different frequencies. Calculate the H_∞ norm of the system.

.....
.....

Exercise 3.1.4. Investigate the RGA of the system at frequency 0. What conclusions can we draw about the possibility of using decentralized control?

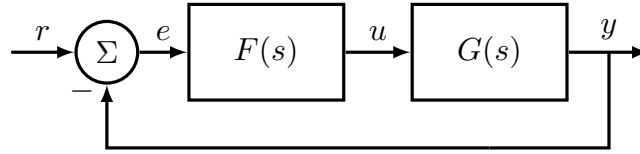


Figure 2: Control.

.....

Exercise 3.1.5. Plot the step response for one input at the time. Investigate the outputs: is the system coupled? Is this in line with the properties of RGA?

.....

Now solve the above problems above for the non-minimum phase case.

Exercise 3.1.6. Describe the most important differences between the two cases and discuss how it affects the control performance.

.....

3.2 Decentralized control

We will now investigate control of the four-tank process as illustrated in Figure 2, where both the process $G(s)$ and the controller $F(s)$ are 2×2 matrices of transfer functions. The simplest way to control a system is to use decentralized control. This means that one input is paired with one output, which is controlled with a one-dimensional controller. An example is depicted in Figure 3, where output y_1 is controlled with the input u_1 through the controller $f_1(s)$. Similarly, the output y_2 is controlled with the input u_2 through $f_2(s)$. Here y_1 is paired with u_1 and y_2 is paired with u_2 . This corresponds to Figure 2 with

$$F(s) = \begin{bmatrix} f_1(s) & 0 \\ 0 & f_2(s) \end{bmatrix}$$

The other way around, if y_1 is paired with u_2 and y_2 is paired with u_1 , then $F(s)$ is given by

$$F(s) = \begin{bmatrix} 0 & f_1(s) \\ f_2(s) & 0 \end{bmatrix}$$

In the first case, the controllers $f_1(s)$ and $f_2(s)$ are designed using single-variable control design with the transfer functions between u_1 and y_1 and between u_2 and y_2 . A

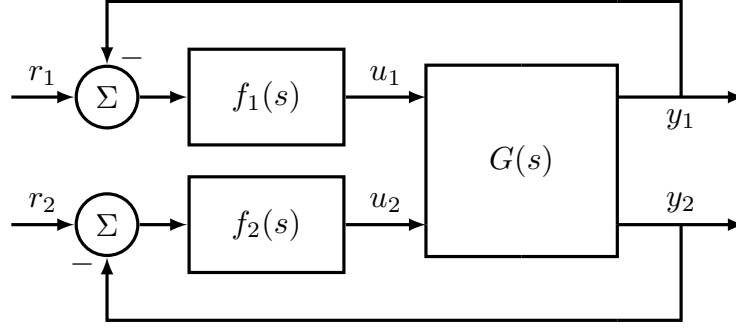


Figure 3: Decentralized control.

procedure for one-dimensional control design was investigated in the computer exercise on loop-shaping. Here we will design PI controllers

$$f_j(s) = K_j \left(1 + \frac{1}{sT_{i_j}} \right), \quad j = 1, 2$$

such that the intended phase margin φ_m and cross-over frequency ω_c are obtained. The loop gain is given by $L = GF$. Therefore we wish to shape l_{11} and l_{22} in such a way that given specifications are fulfilled.

Let us now investigate the case where y_1 is to be controlled with u_1 . Denote the intended phase margin and cross-over frequency by φ_m and ω_c respectively. From l_{11} we see that K_1 and T_{i_1} are given by the following equations:

$$|g_{11}(i\omega_c)f_1(i\omega_c)| = 1 \quad (1)$$

$$\arg g_{11}(i\omega_c) + \arg f_1(i\omega_c) - \varphi_m = -\pi \quad (2)$$

Note that Equation (2) is equivalent to

$$\arg g_{11}(i\omega_c) + \arctan(\omega_c T_{i_1}) - \pi/2 - \varphi_m = -\pi$$

where $\arg g_{11}(i\omega_c)$ can be obtained from the Bode diagram of $g_{11}(s)$. This gives T_{i_1} . Then we can draw the Bode diagram for the loop gain

$$l_{11}(s) = g_{11}(s) \left(1 + \frac{1}{sT_{i_1}} \right)$$

Equation (1) now gives

$$K_1 = \frac{1}{|l_{11}(i\omega_c)|}$$

where $|l_{11}(i\omega_c)|$ is obtained from the Bode diagram of $l_{11}(s)$. Control design for other input/output pairings is performed similarly. Apart from MATLAB functions already mentioned, the following ones can help you: **tf**, **zoom**, **append**, **inv** and **feedback**.

The problems below should be solved both for the minimum phase and non-minimum phase case. It is suitable to start with the former.

Exercise 3.2.1. Design a decentralized controller by pairing inputs and outputs according to the RGA analysis. The intended phase margin is $\varphi_m = \pi/3$ and the crossover frequency ω_c is 0.1 rad/s for the minimum phase case and 0.02 rad/s for the non-minimum phase case. (To make sure that the problem is correctly solved, investigate the Bode diagram of L .)

.....

.....

Exercise 3.2.2. Calculate the singular values of the sensitivity function

$$S(s) = (I + G(s)F(s))^{-1}$$

and the complementary sensitivity function

$$T(s) = (I + G(s)F(s))^{-1}G(s)F(s)$$

Is the design good with respect to sensitivity and robustness?

.....

.....

Exercise 3.2.3. Simulate the closed-loop system in Simulink by typing `closedloop`. A Simulink window will appear where the block diagram is shown. Make sure that the variables `F` and `G` in the MATLAB work-space contain the controller and the process respectively. Go to the Simulation menu and click Start. On the screen the unit step responses from the references to the outputs $y_1(t)$ (at $t = 100$) and $y_2(t)$ (at $t = 500$) are plotted together with the inputs. The time instant of the steps can be modified by clicking on the step blocks and changing the Step time. The total simulation time can be modified by changing the Stop time in the menu Simulation/Parameters. Simulation data is saved in the variable `simout` in the MATLAB workspace. Is the control good? Are the outputs coupled?

.....

.....

Now solve the above problems for the non-minimum phase case.

Exercise 3.2.4. Describe the most important differences between the two cases.

.....

.....