University of Siegen
Department of Electrical Engineering and Computer Science
Institute of Automatic Control Engineering

Prof. R. Mayr

March, 7 2006

Advanced Control

Experiment 1: Block Diagram Simulation

1 Introduction

MATLAB is a high-performance standard tool for design, simulation and analysis of dynamical systems in control engineering.
SIMULINK, an extension toolbox for MATLAB, provides a graphical user interface (GUI) for building linear and nonlinear system models as block diagrams.
The SIMULINK libraries provide single block components, for example signal sources, transfer functions and scopes, which can be used to create your own system, called "model".
The single block components are copied into your model by drag and drop operations.
After connecting the blocks appropriately and setting up the simulation and block parameters, the simulation can be started.

2 Simulation with MATLAB and SIMULINK

2.1 Starting MATLAB and SIMULINK, creating a new model

Start MATLAB by double-clicking the MATLAB-icon on the Windows desktop or by using the Windows “Start”-menu (Start → Programme → MATLAB *x.y* → MATLAB *x.y*). A Window with the MATLAB-Desktop opens up. A possible appearance of the MATLAB-desktop is shown in figure 1.

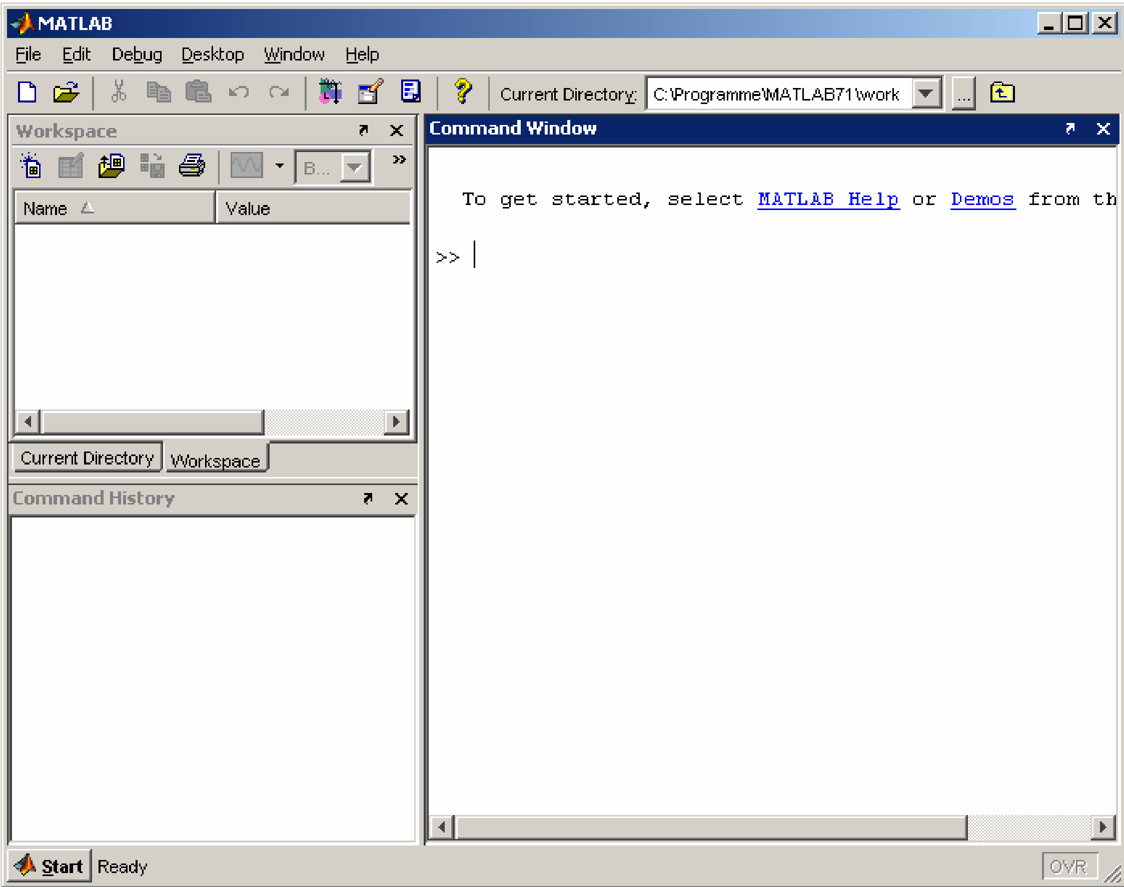


Figure 1: The MATLAB-desktop

Now SIMULINK can be started by entering the command “simulink” into the “Command Window” or clicking the SIMULINK-icon in the toolbar below the menu.

SIMULINK starts by opening up the “Simulink Library Browser”-window shown in figure 2 (see next page).

The upper part of the window contains a description of the selected group or block. The lower left part of the window contains the groups and subgroups of blocks organized in a tree view. The single block components can be found in the lower left part of the window.

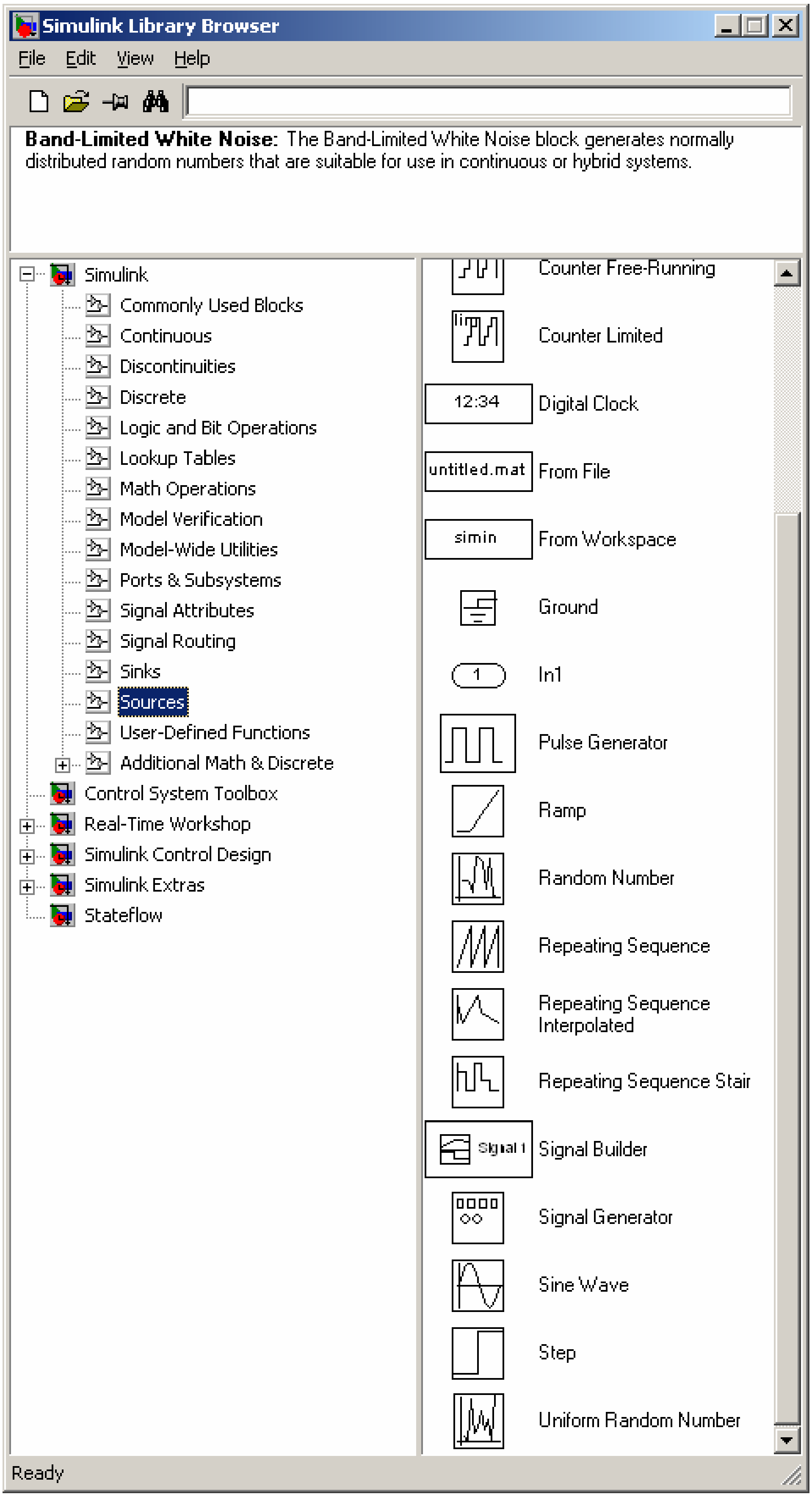


Figure 2: The SIMULINK library browser

Now start creating a new "model". Click the "Create a new model"-icon  in the toolbar below the menu or use the menu itself (File → New → Model).

The "Model"-window opens up. Figure 3 shows the Model-window that already includes the "Step"-block from the "Simulink / Sources"-library.

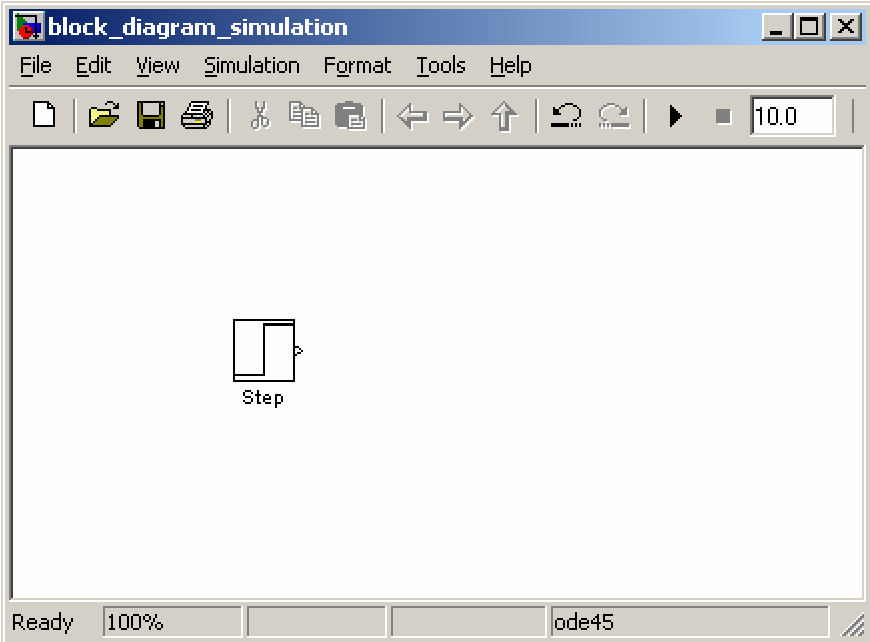


Figure 3: "Model"-window with "Step"-block

The single block components from the "Simulink Library Browser"-window can be copied to the "Model"-window using drag & drop.

2.2 Connecting blocks

Two blocks can be connected with each other by clicking the input or the output of one of the blocks, holding down the mouse button, moving the cursor to the corresponding output or input of the second block and releasing the mouse button.
You can draw a line with a vertex by releasing the mouse button when the connection is incomplete. From the point where you released the button it is possible to continue the line.
If you want to draw a branching point, position the cursor on the line, where you would like to have the branching point. Press and hold down the right mouse button. Move the cursor to the input of the block and release the mouse button.

2.3 Specifying block parameters

After a double-click on a block, a parameter or display window for the corresponding block opens up. The following text lists important blocks and their (important, not all) parameters.

2.3.1 "Step"-block (Library: Simulink / Sources)

Parameters: Step time (s)
Initial value
Final value

2.3.2 Transfer Function: "Transfer Fcn"- block (Library: Simulink / Continues)

A transfer-function of the form $G(s) = \frac{Y(s)}{U(s)} = \frac{b_m s^m + \ldots + b_1 s + b_0}{a_n s^n + \ldots + a_1 s + a_0}$ can be realized with this block.

Parameters: Numerator coefficients as a vector: $[b_m \quad b_{m-1} \quad \ldots \quad b_1 \quad b_0]$
Denominator coefficients as a vector: $[a_n \quad a_{n-1} \quad \ldots \quad a_1 \quad a_0]$

2.3.3 "Scope"-block (Library: Simulink / Sinks)

Figure 4 shows an opened scope-block with a toolbar in the upper part of the window.

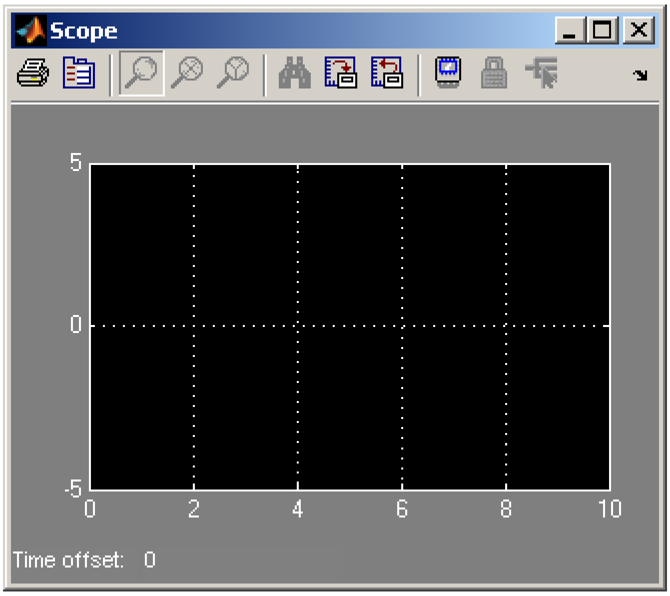


Figure 4: Opened scope-block

Functions in the toolbar: (from the left to the right side)

- Print
- Parameters
- Zoom
- Zoom X-axis
- Zoom Y-axis
- Autoscale
- Save current axes settings
- Restore current axes settings
- Floating scope
- Unlock axes selection
- Signal selection

If you choose the “Parameters”-icon, a window opens up with the cards “General” (shown in figure 5) and “Data history” (shown in figure 6).

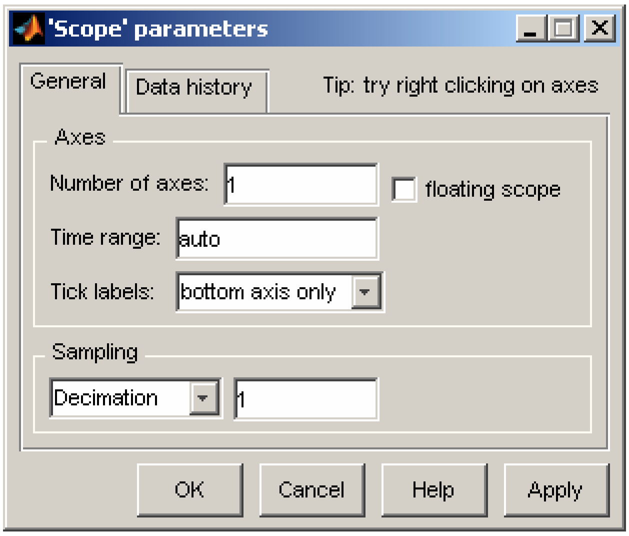


Figure 5: “General”-card of “Scope” parameters

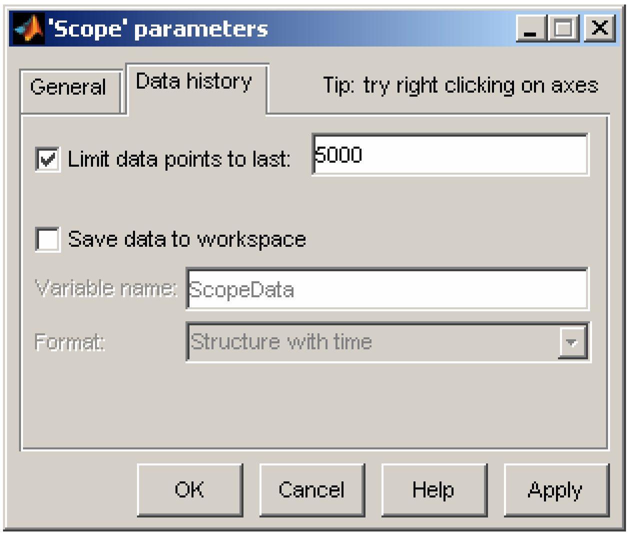


Figure 6: “Data history”-card of “Scope” parameters

Note: If you are missing calculated data at the beginning of your diagram, remove the hook in the check-box in front of “Limit data points to last:”.

The scaling and the title of the diagram can be changed by choosing the point “Axes properties” from the pop-up-window (shown in figure 7, next page) that opens up when you click on the diagram with the right mouse button. The “Axes properties”-window itself is shown in Figure 8 (also on the next page).

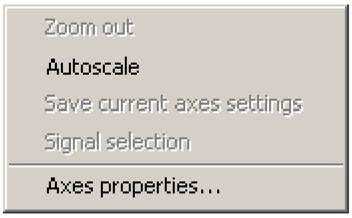


Figure 7: Pop-up-window for diagram options

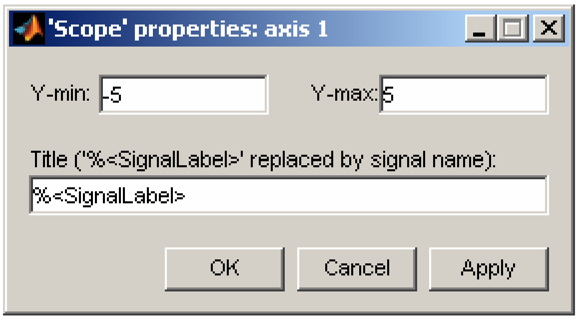


Figure 8: "Axes properties window" for axis 1

2.3.4 Multiplexer: "Mux"-block (Library: Simulink / Signal Routing)

Parameter: Number of inputs

2.3.5 "Add", "Subtract" or "Sum"-block (Library: Simulink / Math Operations)

Parameters: A sequence of plus or minus signs. The number of inputs is equal to the number of signs. The position of the inputs is influenced by the vertical bar "|"-sign.
Icon shape (round or rectangular)

2.3.6 "Gain"-block (Library: Simulink / Math Operations)

Parameter: Gain

2.3.7 "Saturation"-block (Library: Simulink / Discontinuities)

Parameters: Upper limit
Lower limit

2.3.8 "Integrator"-block (Library: Simulink / Continuous)

Parameters: External Reset (none, rising, falling, either, level)
Initial condition
Limit output (upper and lower saturation limit)

2.3.9 “Derivative”-block (Library: Simulink / Continuous)

2.4 Running the simulation and printing the results.

Parameters for the simulation can be set by choosing the menu point “Configuration Parameters…” from the “Simulation” menu in the model window. The window “Configuration parameters” shown in figure 9 will open up.

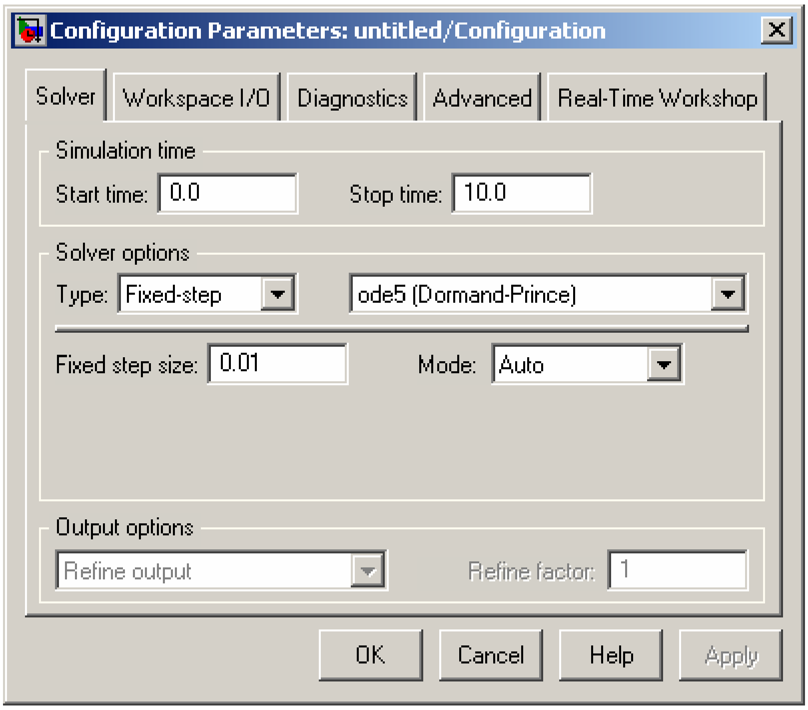


Figure 9: Configuration parameters for the simulation

Change the parameters according to the list below and accept the settings.

Parameter:	“Solver options”, “Type”:	Fixed-step
	“Solver”:	ode5 (Dormand-Prince)
	“Fixed step size”:	0.01 sec

After adjusting the settings, the simulation can be started using the menu (Simulation → Start) or clicking on the “start simulation”-button ▶ in the toolbar of the model window.

When the simulation has run, the results can be printed out by clicking onto the “Print”-icon in the scope window. Select the “Eigenschaften” (= properties) button and change the paper format from “Hochformat” (= portrait format) to “Querformat” (= landscape format). Accept all setting and start the print process by clicking the “OK”-button.

3 Experiment

3.1 Two second-order systems have to be modelled with “Transfer Function”-blocks. The step responses shall be recorded simultaneously.

Given parameters:	Transfer function:	$G(s) = \frac{Y(s)}{U(s)} = \frac{K_s}{{T_b}^2 s^2 + T_a s + 1}$
	Relation for the damping:	$D = \frac{T_a}{2T_b}$
	Given Values:	$K_{s1} = K_{s2} = 0.18$
		${T_{b1}}^2 = {T_{b2}}^2 = 0.36\,\sec^2$
		$D_1 = 1;\ D_2 = 0.6$
	SIMULINK model:	see figure 10

Calculate the values for T_{a1} and T_{a2}, set the parameters of the transfer function blocks according to your results and start the simulation. Print out the result.

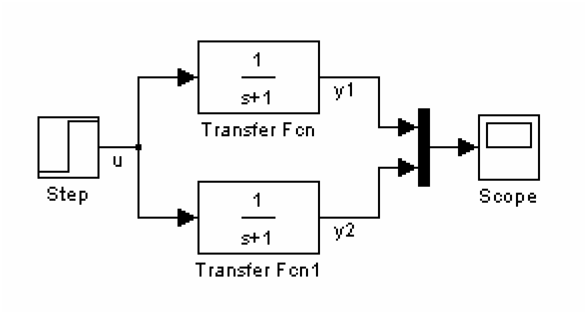


Figure 10: SIMULINK model for exercise 3.1, 3.2 and 3.3

3.2 Recording of step responses like in exercise 3.1, but with different values for damping

Given parameters:	Given Values:	$D_1 = 0;\ D_2 = -0.05$
	All other parameters:	see exercise 3.1

3.3 Recording of step responses like in exercise 3.1, but with different transfer functions

Given parameters:	Transfer functions:	$G_1(s) = \frac{Y_1(s)}{U(s)} = \frac{K_I}{s}$
		$G_2(s) = \frac{Y_2(s)}{U(s)} = \frac{K_I}{s(T_1 s + 1)}$
	Given Values:	$K_I = 0.1\,\frac{1}{\sec};\ T_1 = 1.5\ \sec$
	SIMULINK model:	see figure 10

3.4 Test the transient response and the disturbance rejection of a closed loop control with a P-controller.

Given parameters:	Controlled system:	$G_s(s) = \frac{K_{sI}}{s(T_{s1}s+1)}$	
	Controller (P):	$G_c(s) = K_{cP}$	
		$K_{sI} = 1.25 \frac{1}{\text{sec}}$; $T_{s1} = 0.1$ sec	
	Reference variable:	$w = 0.8$	(Step time: 0 sec)
	Disturbance variable:	$-d = 0.4$	(Step time: 3.5 sec)
	Saturation:	± 1	
	SIMULINK model:	see figure 11	

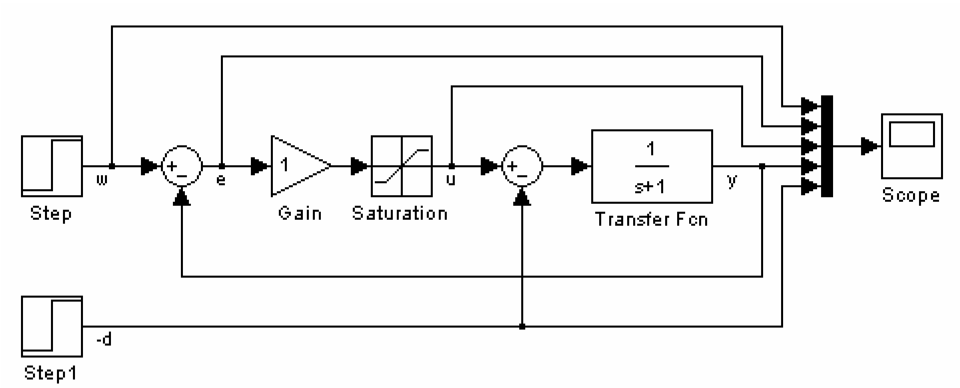


Figure 11: SIMULINK model for exercise 3.4

Transfer function of the closed system in figure 11:

$$G_w(s) = \frac{Y(s)}{W(s)} = \frac{G_c(s)\, G_s(s)}{1+G_c(s)\, G_s(s)} = \frac{K_{cP} \frac{K_{sI}}{s(T_{s1}s+1)}}{1+K_{cP}\frac{K_{sI}}{s(T_{s1}s+1)}} = \frac{K_{cP}\, K_{sI}}{s(T_{s1}s+1)+K_{cP}\, K_{sI}}$$

$$G_w(s) = \frac{K_{cP}\, K_{sI}}{T_{s1}\left(s^2 + \frac{1}{T_{s1}}s + \frac{K_{cP}\, K_{sI}}{T_{s1}}\right)}$$

With $W(s) = \frac{w_0}{s}$ follows: $Y(s) = \frac{w_0\, K_{cP}\, K_{sI}}{T_{sI}} \frac{1}{s\left(s^2 + \frac{1}{T_{s1}}s + \frac{K_{cP}\, K_{sI}}{T_{s1}}\right)}$

Abbreviations: $D = \frac{\alpha}{\beta}$ $\qquad 2\alpha = \frac{1}{T_{s1}}$ $\qquad \beta^2 = \frac{K_{cP}\, K_{sI}}{T_{s1}}$

Record and print out the dynamic transient response disturbance rejection for the system with a P-controller for the following parameters:

3.4.1 $K_{cP} = 2;$ Calculate the value for the damping D.

3.4.2 $D = 0.8;$ Calculate the proportional coefficient K_{cP}.

3.4.3 $D = 1.25;$ Calculate the value for K_{cP}.

3.5 Test the transient response and the disturbance rejection of a closed loop control with a PD-controller.

Extend the controller of the model given in figure 11 with a D-part to get a system with a PD-controller. Figure 12 shows the complete model.

Given parameters:	Controlled system:	$G_s(s) = \dfrac{K_{sI}}{s(T_{s1}s+1)}$	
	Controller (PD):	$G_c(s) = K_{cP} + s\,K_{cD}$	
		$K_{sI} = 1.25\,\dfrac{1}{\text{sec}};\ T_{s1} = 0.1$ sec	
	Reference variable:	$w = 0.8$	(Step time: 0 sec)
	Disturbance variable:	$-d = 0.4$	(Step time: 3.5 sec)
	Saturation:	± 1	
	SIMULINK model:	see figure 12	

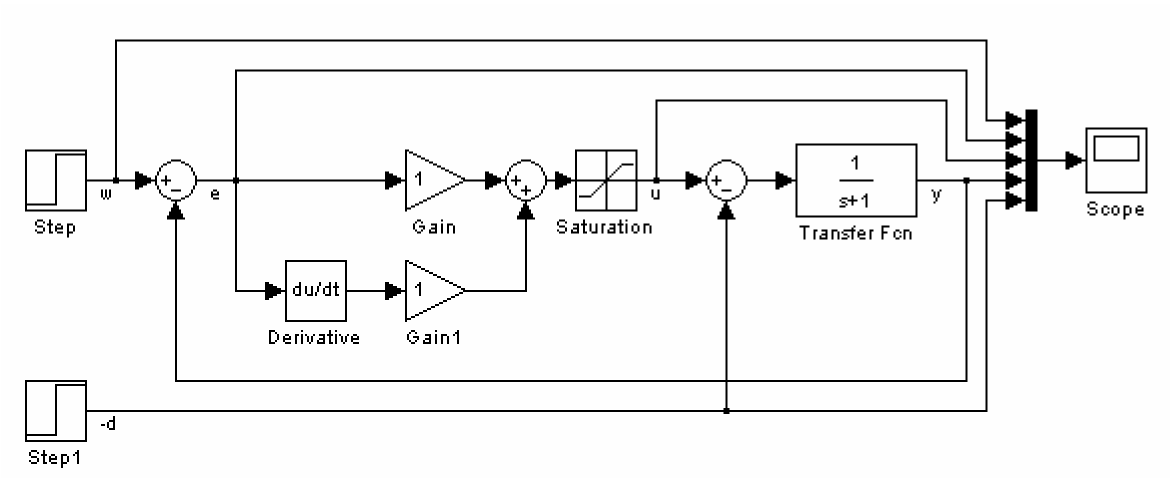


Figure 12: SIMULINK model for exercise 3.5

Transfer function of the closed system in figure 12:

$$G_w(s) = \frac{Y(s)}{W(s)} = \frac{G_c(s)\,G_s(s)}{1+G_c(s)\,G_s(s)} = \frac{\left(K_{cP} + s\,K_{cD}\right)\dfrac{K_{sI}}{s(T_{s1}s+1)}}{1+\left(K_{cP} + s\,K_{cD}\right)\dfrac{K_{sI}}{s(T_{s1}s+1)}}$$

$$G_w(s) = \frac{\left(K_{cP} + s\,K_{cD}\right)K_{sI}}{s(T_{s1}s+1) + \left(K_{cP} + s\,K_{cD}\right)K_{sI}}$$

With the abbreviation $T_{cV} = \dfrac{K_{cD}}{K_{cP}}$ follows: $G_w(s) = \dfrac{K_{cP}\left(1+s\,T_{cV}\right)K_{sI}}{s(T_{s1}s+1) + K_{cP}\left(1+s\,T_{cV}\right)K_{sI}}$

Record and print out the dynamic transient response and disturbance rejection for the system with a PD-controller for the following parameters:

3.5.1 $K_{cP} = 10;\ T_{cV} = T_{s1};$ Calculate the derivative action coefficient K_{cD}.

3.5.2 $K_{cP} = 10;\ T_{cV} = 0.03$ sec; Calculate the value for K_{cD}.

3.6 Test the transient response and the disturbance rejection of a closed loop control with a PID-controller.

Extend the controller of the model given in figure 12 with a I-part to get a system with a PID-controller. Figure 13 shows the complete model.

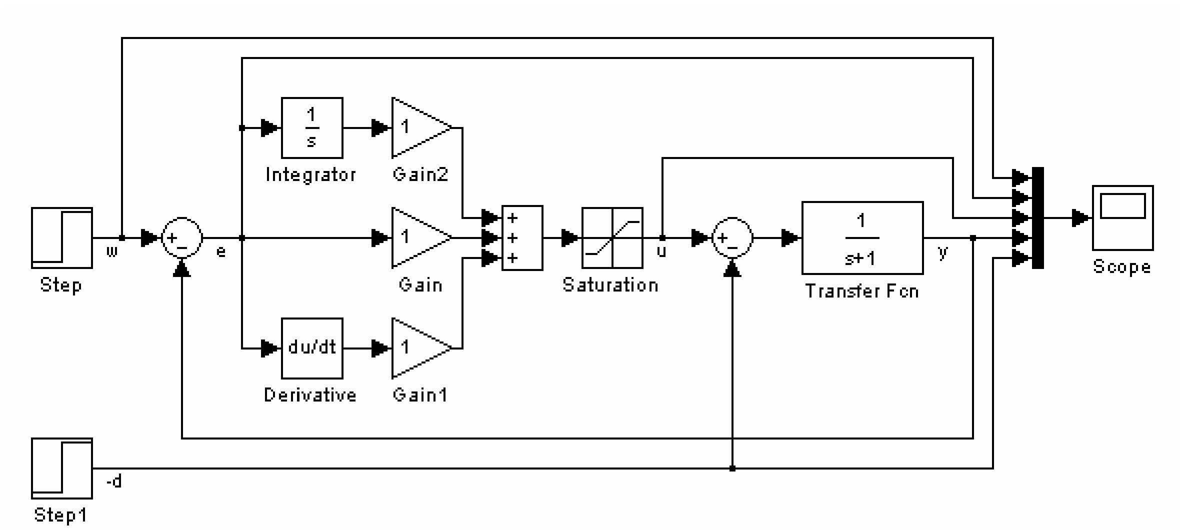


Figure 13: SIMULINK model for exercise 3.6

Given parameters: Controlled system: $G_s(s) = \frac{K_{sI}}{s(T_{s1}s+1)}$

Controller (PID): $G_c(s) = K_{cP} + \frac{K_{cI}}{s} + s\,K_{cD}$

$K_{sI} = 1.25\,\frac{1}{\sec};\ T_{s1} = 0.1\ \sec$

Reference variable:	$w = 0.8$	(Step time: 0 sec)
Disturbance variable:	$-d = 0.4$	(Step time: 3.5 sec)
Saturation:	± 1	
SIMULINK model:	see figure 13 (previous page)	

For the controller function $G_c(s)$ we use the abbreviations $T_{cV} = \frac{K_{cD}}{K_{cP}}$ and $T_{cN} = \frac{K_{cP}}{K_{cI}}$:

$$G_c(s) = K_{cP} + \frac{K_{cI}}{s} + s\,K_{cD} = K_{cP}\left(1 + \frac{K_{cI}}{s\,K_{cP}} + \frac{s\,K_{cD}}{K_{cP}}\right) = K_{cP}\left(1 + \frac{1}{s\,T_{cN}} + s\,T_{cV}\right)$$

Record and print out the dynamic transient response and disturbance rejection for the system with a PID-controller for the following parameters:

3.6.1 $K_{cP} = 10;\ T_{cN} = 0.5\ \sec;\ T_{cV} = 0.1\sec;$ Calculate the derivative action coefficient K_{cD} and the integral action coefficient K_{cI}.

3.6.2 The same parameters as in exercise 3.6.1, but without saturation of the control variable u(t).

4 Analysis

4.1 Label the curves recorded in exercise 3.1, 3.2 and 3.3 with the name of the recorded function: $y_1(t),\ y_2(t)$.

4.2 Enter characteristic variables in the diagrams recorded for exercise 3.1 and 3.3.

Diagram for exercise 3.1: equivalent dead time T_u
(only for $y_1(t)$) build-up time T_g
steady-state value $y(t \to \infty)$

Diagram for exercise 3.3: integral action coefficient K_I
time constant T_1

4.3 Calculate the step responses $Y_i(s)$ for the curves recorded in exercise 3.1, 3.2 and 3.3. Use inverse LAPLACE transform to get the corresponding time functions $y_i(t)$.

Note: Appendix B contains a table with some useful LAPLACE transforms.

4.4 For exercise 3.4 till 3.6:

Label the recorded curves $w(t),\ y(t),\ e(t),\ u(t),\ -d(t)$.
Enter rise time t_{rise}, setting time t_{set} and overshoot $y_{\max}$ into the diagrams for a tolerance zone of $\pm 5\%$.

4.5 For exercise 3.6:

Calculate the reference transfer function $G_W(s)$ and the disturbance transfer function $G_D(s)$ for the closed loop system (without saturation of the control variable).

Calculate $Y(s)$ considering the reference transfer function $G_W(s)$ and the disturbance transfer function $G_D(s)$. Figure 14 shows the superposition of the two components $Y_W(s) = G_W(s)\ W(s)$ and $Y_D(s) = G_D(s)\ D(s)$.

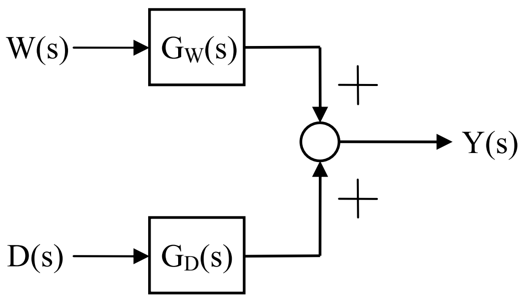


Figure 14: The superposition of $Y_W(s) = G_W(s)\ W(s)$ and $Y_D(s) = G_D(s)\ D(s)$

Use the superposition and the LAPLACE limit theorem to calculate $\lim_{t\to\infty} y(t)$ and $\lim_{t\to\infty} e(t)$ with and without disturbance.

Note: Appendix A contains the LAPLACE limit theorems.

5 Comparison of different controller types

Figure 15 shows the block diagram of the given closed loop control.

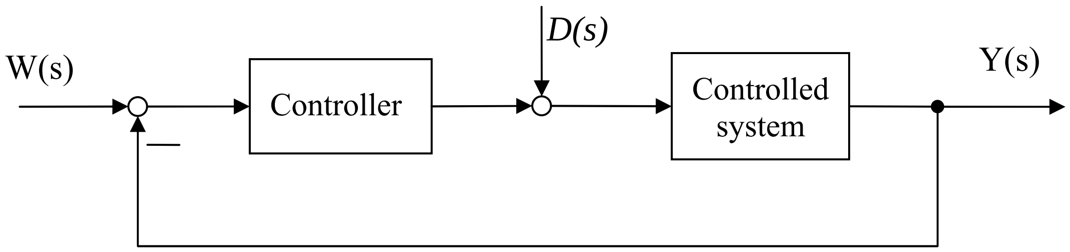


Figure 15: Closed loop control

Given parameters: Controlled system: $G_s(s) = \dfrac{K_s}{T_b^2 s^2 + T_a s + 1}$

P - Controller: $G_{cP}(s) = K_{cP}$

PD - Controller: $G_{cPD}(s) = K_{cP} + s\,K_{cD}$

PI - Controller: $G_{cPI}(s) = K_{cP} + \dfrac{K_{cI}}{s}$

PID - Controller: $G_{cPID}(s) = K_{cP} + \dfrac{K_{cI}}{s} + s\,K_{cD}$

Reference variable: $W(s) = \dfrac{W_0}{s}$

Disturbance variable: $D(s) = \dfrac{D_0}{s}$

5.1 Calculate the reference transfer function $G_W(s)$ and the disturbance transfer function $G_D(s)$ for all types of controllers (P, PD, PI or PID).

5.2 Use the superposition shown in figure 14 and the LAPLACE limit theorem (Appendix A) to calculate $\lim_{t\to\infty} y(t)$ and the steady-sate error $\lim_{t\to\infty} e(t)$ for each type of controller.

5.3 Which of the given controllers (P, PD, PI or PID) are NOT able to avoid a steady-state error with the given reference- and disturbance variables?

5.4 What is the important part of the controller that avoids the steady-state error (P, D or I)?

6 Appendix

6.1 Appendix A: LAPLACE limit theorems

Initial value theorem: $\lim\limits_{t \to 0^+} f(t) = \lim\limits_{s \to \infty} s\,F(s)$

Final value theorem: $\lim\limits_{t \to \infty} f(t) = \lim\limits_{s \to 0} s\,F(s)$

6.2 Appendix B: Table of LAPLACE transforms

Fundamental functions:

No.	$F(s)$	$f(t),\ t > 0$	Function name
1	1	$\delta(t)$	unit impulse
2	e^{-Ts}	$\delta(t-T)$	shifted unit impulse
3	$\frac{1}{s}$	$\sigma(t)$	unit step
4	$\frac{1}{s}e^{-Ts}$	$\sigma(t-T)$	shifted unit step
5	$\frac{1}{s^2}$	$t\,\sigma(t)$	ramp
6	$\frac{1}{s^2}e^{-Ts}$	$(t-T)\ \sigma(t-T)$	shifted ramp

Further functions:

No.	$F(s)$	$f(t),\ t > 0$
7	$\frac{1}{1+T_1 s}$	$\frac{1}{T_1} e^{-\frac{t}{T_1}}$
8	$\frac{1}{(1+T_1 s)\,s}$	$1 - e^{-\frac{t}{T_1}}$
9	$\frac{1}{(1+T_1 s)\,s^2}$	$t - T_1 + T_1\, e^{-\frac{t}{T_1}}$
10	$\frac{1}{(s+a)^2}$	$t\ e^{-at}$
11	$\frac{1}{(s+a)^2\, s}$	$\frac{1}{a^2}\left[1 - e^{-at} - a\ t\ e^{-at}\right]$
12	$\frac{\omega_0^{\ 2}}{s^2 + 2D\omega_0 s + \omega_0^{\ 2}}$	$\frac{\omega_0}{\sqrt{1-D^2}} e^{-D\omega_0 t} \sin(\omega_0 \sqrt{1-D^2}\ \ t);\ -1 < D < 1$ $\frac{\omega_0^{\ 2}}{\omega_e} e^{-D\omega_0 t} \sin(\omega_e\, t)\ \ with\ \ \omega_e = \omega_0 \sqrt{1-D^2}$

13	$\dfrac{{\omega_0}^2}{(s^2 + 2D\omega_0 s + {\omega_0}^2)\,s}$	$1 - \dfrac{1}{\sqrt{1-D^2}} e^{-D\omega_0 t} \sin(\omega_e t + \varphi);\ -1 < D < 1$ *with* $\omega_e = \omega_0 \sqrt{1-D^2};\ \varphi = \arccos(D)$
14	$\dfrac{\omega}{s^2 + \omega^2}$	$\sin(\omega\, t)$
15	$\dfrac{s}{s^2 + \omega^2}$	$\cos(\omega\, t)$
16	$\dfrac{1}{(s^2 + \omega^2)\,s}$	$\dfrac{1}{\omega^2}\left[1 - \cos(\omega\, t)\right]$

University of Siegen
Department of Electrical Engineering and Computer Science
Institute of Automatic Control Engineering

Prof. R. Mayr February, 20 2006

Advanced Control

Experiment 2: Simulation of position control

1 Introduction

The object of the experiment is to analyse the behaviour of different control strategies on one controlled system.
The controlled system, shown in figure 1, consists of two first-order systems and an integrator.

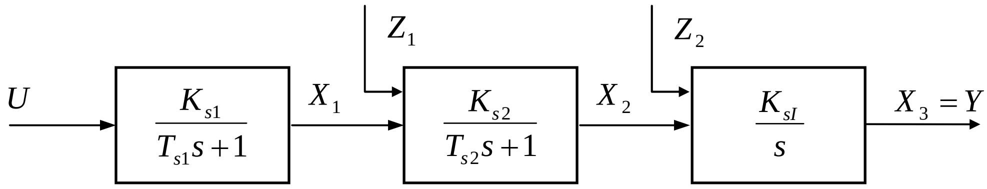


Figure 1: Block diagram of the controlled system

System parameters: $K_{s1} = 0.5;\ T_{s1} = 0.25\ \text{sec};\ K_{s2} = 1;\ T_{s2} = 0.5\ \text{sec};\ K_{sI} = \frac{1}{T_{sI}} = 0.8\frac{1}{\text{sec}}$

For this controlled system the variables $x_1(t)$ and $x_2(t)$ are as important as the controlled value $y(t) = x_3(t)$ itself. Besides conventional control and cascade control state control is tested with the system.

2 Simulation with MATLAB and SIMULINK

2.1 Used blocks and their important parameters

This chapter lists only the blocks not mentioned in chapter 2.3 of the instructions for experiment 1.

2.1.1 "PID Controller"-block (Library: Simulink Extras / Additional Linear)

Parameters:
- Proportional: K_{cP}
- Integral: $K_{cI} = \frac{K_{cP}}{T_{cN}}$
- Derivative: $K_{cD} = K_{cP} T_{cV}$

2.1.2 Absolute value: "Abs"-block (Library: Simulink / Math Operations)

2.1.3 "Product"-block (Library: Simulink / Math Operations)

2.1.4 Digital "Display"-block (Library: Simulink / Sinks)

2.2 Parameters for the simulation

Select the menu point "Configuration Parameters…" from the "Simulation" menu in the model window. The window "Configuration parameters" will open up.

Change the parameters according to the list below:

Parameter:		
	"Stop time":	30.0 sec
	"Solver options", "Type":	Fixed-step
	"Solver":	ode5 (Dormand-Prince)
	"Fixed step size":	0.006

Figure 2 on the next page shows the configuration parameters window with the correct settings.

If a smaller value than 0.006 sec is chosen as the "sample time", there are more than 5000 data values for the scope to display. In this case review the "Data history" card in the scope parameters window and remove the hook in the checkbox in front of "Limit data points to last:". The correct settings are shown in figure 3 on the next page.

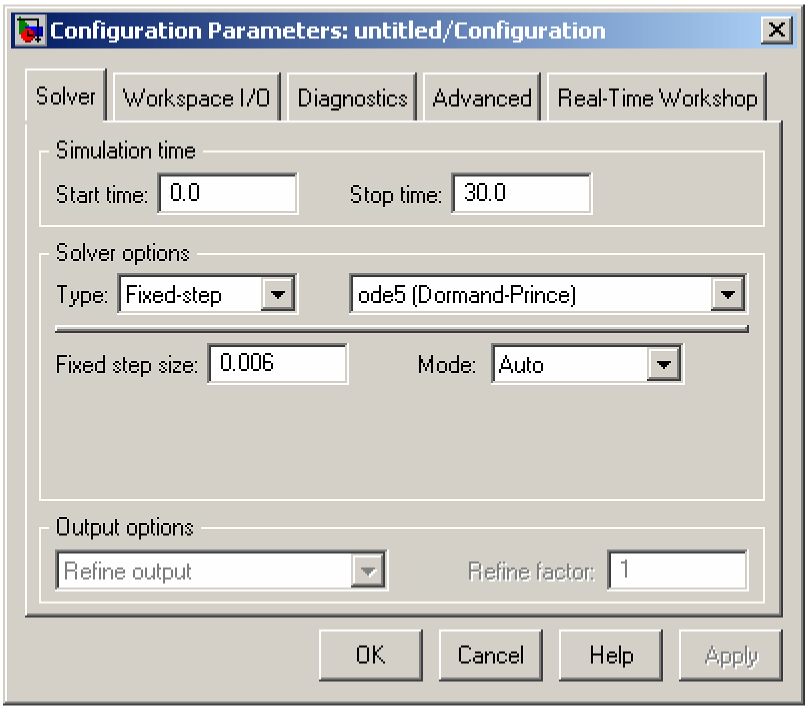


Figure 2: Configuration parameters for the simulation

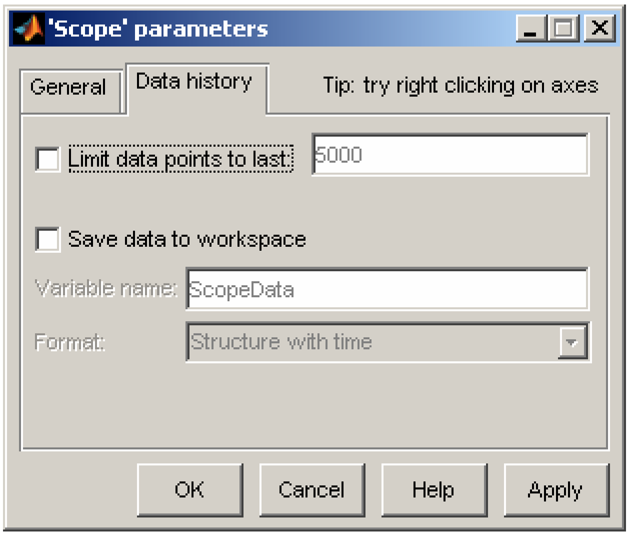


Figure 3: “Data history”-card of “Scope” parameters

3 Experiment

The controlled system given in figure 1 will now be equipped with a single PID-controller, a cascade control structure and a state controller.
In order to have a benchmark to compare the different control strategies the control error itself, the Integral of Absolute value of Error (IAE) and the Integral of Squared Error (ISE) is calculated and displayed.

3.1 Standard feedback control system with PID-controller

Examine the transmission behaviour of the closed-loop control consisting of the controlled system shown in figure 1 and a PID-controller. The complete MATLAB-model is shown in figure 4.

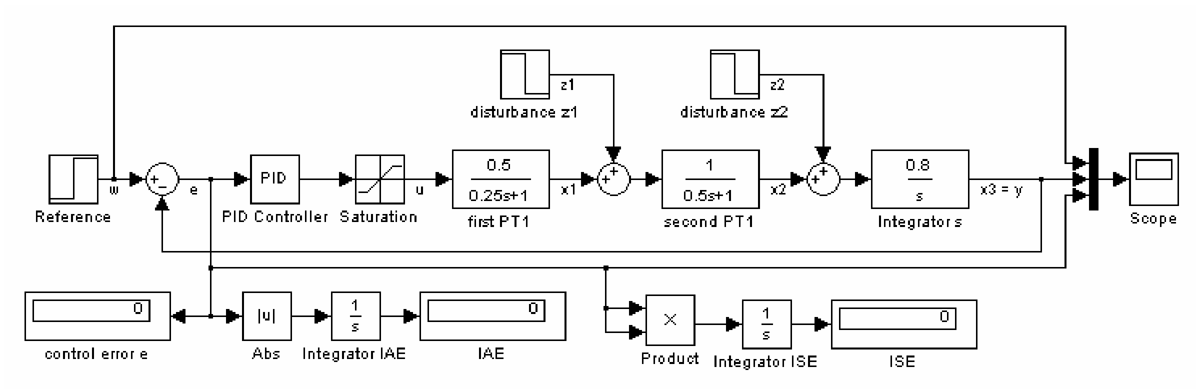


Figure 4: Closed-loop control with PID-controller

Record and print the curves for $w(t)$, $y(t)$, $e(t)$ and note the final values for IAE and ISE.
Use the following parameters for the simulation:

Given parameters:	Reference value:	$w = 1$	(Step time: 1 sec)
	Disturbance $z_1(t)$:	$z_1(t) = -1;$	(Step time: 10 sec)
	Disturbance $z_2(t)$:	$z_2(t) = -1;$	(Step time: 20 sec)
	Saturation:	$u_{\min} = -40;\ u_{\max} = 40$	
	Controller parameters:	$K_{cP} = 40;\ K_{cI} = 32\,\frac{1}{\text{sec}};\ K_{cD} = 11.3\text{ sec}$	

Label the printed curves with the name of the function ($w(t)$, $y(t)$, $e(t)$).

3.2 Cascade control

Examine the transmission behaviour of the closed-loop control consisting of the controlled system shown in figure 1 and a cascade control consisting of a PD-controller for the interior control loop and a PID-controller for the outer control loop. The complete MATLAB-model is shown in figure 5.

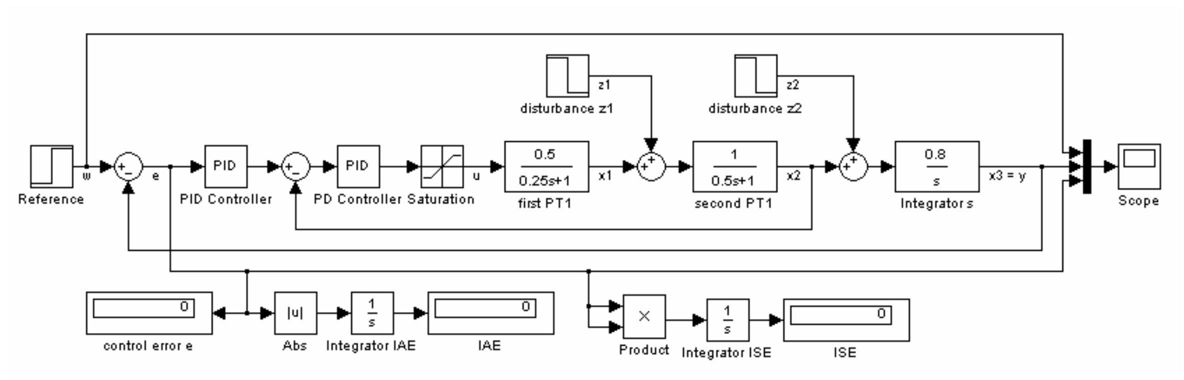


Figure 5: Closed-loop control with a cascade controller

Record and print the curves for $w(t)$, $y(t)$, $e(t)$ and note the final values for IAE and ISE. Use the following parameters for the simulation:

Given parameters:	Reference value:	$w = 1$	(Step time: 1 sec)
	Disturbance $z_1(t)$:	$z_1(t) = -1;$	(Step time: 10 sec)
	Disturbance $z_2(t)$:	$z_2(t) = -1;$	(Step time: 20 sec)
	Saturation:	$u_{\min} = -40;\ u_{\max} = 40$	
	Controller parameters:		
	outer control loop:	$K_{coP} = 40;\ K_{coI} = 32\,\frac{1}{\sec};\ K_{coD} = 11.3\ \sec$	
	interior control loop:	$K_{ciP} = 3;\quad K_{ciI} = 0\,\frac{1}{\sec};\quad K_{ciD} = 0.10\ \sec$	

Label the printed curves with the name of the function ($w(t)$, $y(t)$, $e(t)$).

3.3 State control

Examine the transmission behaviour of the closed-loop control consisting of the controlled system shown in figure 1 and a state controller. The complete MATLAB-model is shown in figure 6.

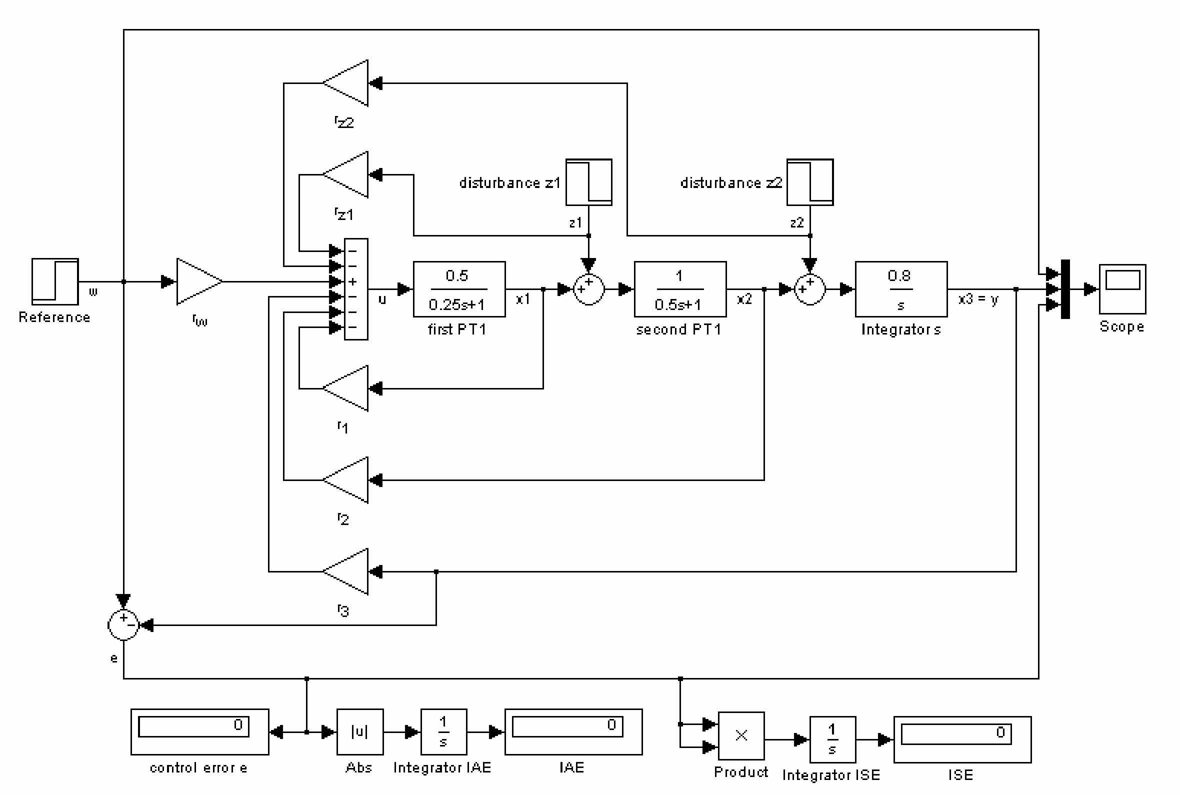


Figure 6: Closed-loop control with a state controller

Equations for the system shown in figure 6:

LAPLACE domain	time domain
$X_1(s) = \frac{K_{s1}}{T_{s1}s+1} U(s)$	$\dot{x}_1(t) = -\frac{1}{T_{s1}} x_1(t) + \frac{K_{s1}}{T_{s1}} u(t)$
$X_2(s) = \frac{K_{s2}}{T_{s2}s+1} X_1(s)$	$\dot{x}_2(t) = -\frac{1}{T_{s2}} x_2(t) + \frac{K_{s2}}{T_{s2}} x_1(t)$
$X_3(s) = \frac{1}{T_{sI}s} X_2(s)$	$\dot{x}_3(t) = \frac{1}{T_{sI}} x_2(t)$
$Y(s) = X_3(s)$	$y = x_3(t)$

The equations for the system written in matrix form (time domain):

$$\underline{\dot{x}}(t) = \underline{A}\,\underline{x}(t) + \underline{B}\,u(t) = \begin{bmatrix} \dot{x}_1(t) \\ \dot{x}_2(t) \\ \dot{x}_3(t) \end{bmatrix} = \begin{bmatrix} -\frac{1}{T_{s1}} & 0 & 0 \\ \frac{K_{s2}}{T_{s2}} & -\frac{1}{T_{s2}} & 0 \\ 0 & \frac{1}{T_{sI}} & 0 \end{bmatrix} \begin{bmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{bmatrix} + \begin{bmatrix} \frac{K_{s1}}{T_{s1}} \\ 0 \\ 0 \end{bmatrix} u(t)$$

The poles of the closed-loop control can be calculated with the equation

$$\det\left(s\,\underline{I} - \underline{A} + \underline{B}\,\underline{R}\right) = 0\,.$$

with $\underline{I}$: identity matrix
$\underline{A}$: system matrix
$\underline{B}$: input matrix
$\underline{R}$: feedback matrix

$$\det\left(s\,\underline{I} - \underline{A} + \underline{B}\,\underline{R}\right) = \det\left(\begin{bmatrix} s & 0 & 0 \\ 0 & s & 0 \\ 0 & 0 & s \end{bmatrix} - \begin{bmatrix} -\frac{1}{T_{s1}} & 0 & 0 \\ \frac{K_{s2}}{T_{s2}} & -\frac{1}{T_{s2}} & 0 \\ 0 & \frac{1}{T_{sI}} & 0 \end{bmatrix} + \begin{bmatrix} \frac{K_{s1}}{T_{s1}} \\ 0 \\ 0 \end{bmatrix} \begin{bmatrix} r_1 & r_2 & r_3 \end{bmatrix} \right) =$$

$$= s^3 + s^2\left[\frac{1}{T_{s1}}\left(1 + K_{s1}r_1\right) + \frac{1}{T_{s2}}\right] + s\left[\frac{1}{T_{s1}T_{s2}}\left(1 + K_{s1}r_1\right) + \frac{K_{s1}K_{s2}}{T_{s1}T_{s2}}r_2\right] + \frac{K_{s1}K_{s2}}{T_{s1}T_{s2}T_{sI}}r_3 = 0$$

The coefficients of the equation above are now compared with the coefficients of the following equation:

$$(s - \lambda_1)(s - \lambda_2)(s - \lambda_3) = s^3 - s^2(\lambda_1 + \lambda_2 + \lambda_3) + s(\lambda_1\lambda_2 + \lambda_1\lambda_3 + \lambda_2\lambda_3) - \lambda_1\lambda_2\lambda_3 = 0$$

From the comparison and with $\lambda_{2/3} = \sigma \pm j\omega$ we get:

$$p_0 = \frac{K_{s1}K_{s2}}{T_{s1}T_{s2}T_{sI}}r_3 = -\lambda_1\lambda_2\lambda_3 = -\lambda_1\left(\sigma^2 + \omega^2\right)$$

$$p_1 = \frac{1}{T_{s1}T_{s2}}\left(1 + K_{s1}r_1\right) + \frac{K_{s1}K_{s2}}{T_{s1}T_{s2}}r_2 = \lambda_1\lambda_2 + \lambda_1\lambda_3 + \lambda_2\lambda_3 = 2\lambda_1\sigma + \sigma^2 + \omega^2$$

$$p_2 = \frac{1}{T_{s1}}\left(1 + K_{s1}r_1\right) + \frac{1}{T_{s2}} = -(\lambda_1 + \lambda_2 + \lambda_3) = -\lambda_1 - 2\sigma$$

For r_1, r_2 and r_3 we get (symbolic and with the values of the system):

$$r_1 = \frac{(p_2 T_{s2} - 1)T_{s1} - T_{s2}}{K_{s1} T_{s2}} \qquad r_1 = \frac{1}{2} p_2 - 3$$

$$r_2 = \frac{p_1 T_{s1} T_{s2} - 1 - K_{s1} r_1}{K_{s1} K_{s2}} \qquad r_2 = \frac{1}{4} p_1 - \frac{1}{2} p_2 + 1$$

$$r_3 = \frac{p_0 T_{s1} T_{s2} T_{sI}}{K_{s1} K_{s2}} \qquad r_3 = \frac{5}{16} p_0$$

Calculate the coefficients (p_0, p_1, p_3) and the feedback factors (r_1, r_2, r_3) for the pole placements given in 3.3.1 to 3.3.6. Choose the gain for the reference variable to $r_w = r_3$.
To avoid a steady-state error on disturbance, the disturbance feedback factors (r_{z1}, r_{z2}) can be determined in the following way:

- Set $z_1 = -1$ (Step time: 10 sec), $z_2 = 0$, $r_{z1} = 0$ and $r_{z2} = 0$.
- Run the simulation and calculate the disturbance feedback factor r_{z1} with the help of the gain for the reference variable r_w and the steady-state error $e(t \to \infty) = \lim_{t \to \infty} e(t)$:
  $r_{z1} = r_w \, e(t \to \infty)$.
- The steady-state error should be zero, if you set the disturbance feedback factor r_{z1} and run the simulation again.
- Now set $z_2 = -1$ (Step time: 20 sec), r_{z2} remains zero.
- Run the simulation to determine the steady state error. Calculate the second disturbance feedback factor r_{z2}: $r_{z2} = r_w \, e(t \to \infty)$
- The calculated feedback factor is correct, if the steady-state error disappears, after setting the factor and rerunning the simulation.

In case all parameters are complete for exercise 3.3.1 to 3.3.6, record and print the curves for $w(t)$, $y(t)$, $e(t)$ and note the final values for IAE and ISE for each exercise.

Pole placements for the experiment:

3.3.1 $\lambda_1 = \lambda_2 = \lambda_3 = -4$

3.3.2 $\lambda_1 = \lambda_2 = \lambda_3 = -5$

3.3.3 $\lambda_1 = -4$; $\lambda_{2/3} = -4 \pm 2j$

3.3.4 $\lambda_1 = -4$; $\lambda_{2/3} = -4 \pm 4j$

3.3.5 $\lambda_1 = -6.4$; $\lambda_{2/3} = -4 \pm 2j$

3.3.6 $\lambda_1 = -2$; $\lambda_{2/3} = -4 \pm 4j$

4 Analysis

4.1 Performance indices IAE and ISE

Give a formula for the calculation of each of the indices in the time domain.
$IAE(t) = \ldots,\ ISE(t) = \ldots$

4.2 State control

4.2.1 What is the necessary condition to apply state control?

4.2.2 In exercise 3.3 the state variables are directly used to apply state control.
If the state variables are not directly available, is there a possibility to apply state control anyway?
If it is possible, give the name of the auxiliary tool that permits us to apply state control without direct access to the state variables and the necessary condition to apply the auxiliary tool itself.

University of Siegen
Department of Electrical Engineering and Computer Science
Institute of Automatic Control Engineering

Prof. R. Mayr May, 16 2006

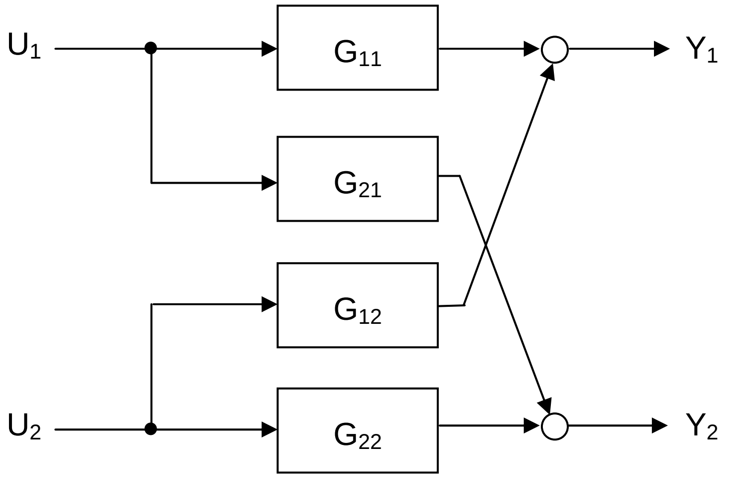
Advanced Control

Experiment 3: Decoupling of multivariable systems and nonlinear compensation control

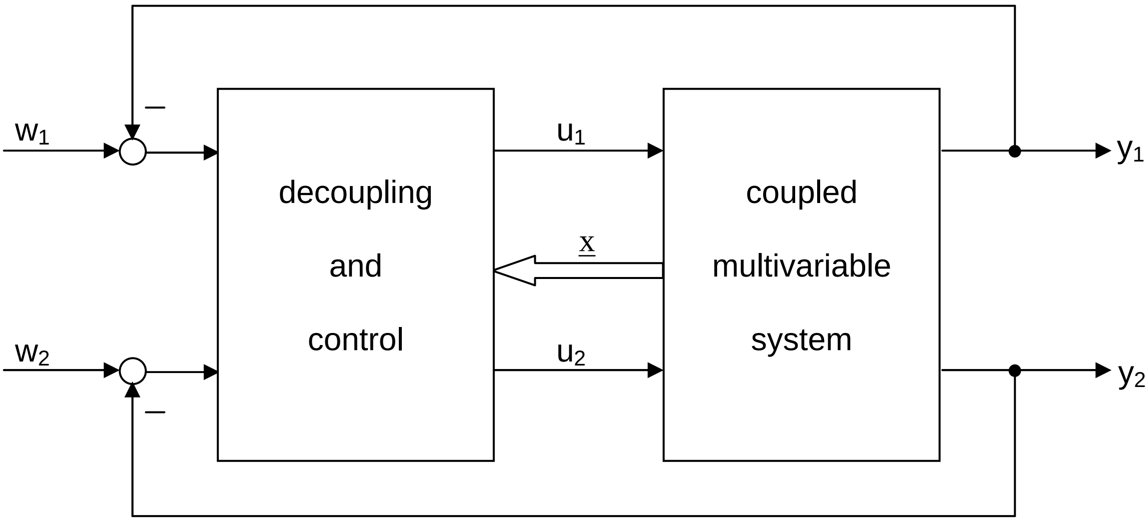
1 Theoretical Introduction

1.1 The Decoupling of Multivariable Systems

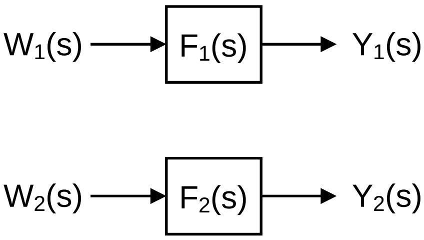
For the multivariable, coupled system



the control algorithms based on decoupling method must be found.



The coupled multivariable system with the decoupling controller should behave like two independent, uncoupled systems:



A decoupling controller can be designed by executing the following 6 steps:

- **Step 1:**

  Determine the differential order for every subsystem. (Every subsystem complies precisely with one output variable each)

$$\delta_i = \begin{cases} 0, \text{ if } \underline{d}_i^T \neq \underline{0}^T \\ k, \text{ if k is the smallest number} \\ \qquad \text{in the range } 0 < k \leq n, \text{ for which} \\ \qquad \text{the following is valid} : \underline{c}_i^T \, \underline{A}^{k-1} \, \underline{B} \neq 0^T \end{cases}$$

- **Step 2:**

  Calculate $\underline{C}$* and $\underline{D}$*

$$\underline{C}^* = \begin{bmatrix} \underline{c}_1^{*T} \\ \vdots \\ \underline{c}_q^{*T} \end{bmatrix}, \quad \underline{c}_i^{*T} = \underline{c}_i^T \, \underline{A}^{\delta_i}$$

$$\underline{D}^* = \begin{bmatrix} \underline{d}_1^{*T} \\ \vdots \\ \underline{d}_q^{*T} \end{bmatrix}, \quad \underline{d}_i^{*T} = \begin{cases} \underline{d}_i^T & \text{for } \underline{d}_i^T \neq \underline{0}^T \\ \underline{c}_i^T \, \underline{A}^{\delta_i - 1} \, \underline{B} & \text{for } \underline{d}_i^T = \underline{0}^T \end{cases}$$

- **Step 3:**
  Calculate the matrix $\underline{D}^{*-1}$ by inverting the matrix $\underline{D}^*$.

The system is only decouplable if

$$\det(\underline{D}^*) \neq 0$$

- **Step 4:**

  Determination of the coefficients

$$\begin{matrix} \ell_1, \alpha_{10}, \cdots, \alpha_{1,\delta_1 - 1} & \quad \textit{with} \quad & \ell_1 = \alpha_{10} \\ \vdots \qquad \vdots & & \vdots \\ \ell_q, \alpha_{qo}, \cdots, \alpha_{q,\delta_q - 1} & \quad \textit{with} \quad & \ell_q = \alpha_{qo} \end{matrix}$$

  due to the desired dynamics for the controlled overall system.

- **Step 5:**
  Set up the matrices $\underline{L}$ and $\underline{M}^*$

$$\underline{L} = \begin{bmatrix} \ell_1 & 0 & 0 & & 0 \\ 0 & \ell_2 & 0 & \cdots & 0 \\ 0 & 0 & \ell_3 & & 0 \\ & \vdots & & \ddots & \vdots \\ 0 & 0 & 0 & \cdots & \ell_q \end{bmatrix}$$

$$\underline{M}^* = \begin{bmatrix} \underline{M}_1^{*T} \\ \vdots \\ \underline{M}_q^{*T} \end{bmatrix}, \quad \underline{M}_i^{*T} = \begin{cases} \underline{0}^T & \text{for } \delta_i = 0 \\ \sum\limits_{k=0}^{\delta_i - 1} \alpha_{ik} \, \underline{c}_i^T \, \underline{A}^k & \text{for } \delta_i > 0 \end{cases}$$

- **Step 6:**

  Insert the calculated matrices into the control law:

$$\underline{u}(t) = \underline{D}^{*-1} \left(- \underline{C}^{*} \underline{x}(t) + \underline{L}\, \underline{w}(t) - \underline{M}^{*} \underline{x}(t) \right)$$

1.2 Direct application of the principle of the feedback linearization

For a non-linear single input/output system

$$\dot{\underline{x}}(t) = \underline{A}(\underline{x}) + \underline{B}(\underline{x}) \cdot u(t)$$
$$y(t) = \underline{C}(\underline{x})$$

the direct application of feedback linearization includes the following 4 steps:

- **Step 1:**

  The output equation is to be differentiated

- **Step 2:**

  Replace the derivatives of the state variables on the right side of the differentiated output equation regarding the state differential equation.

  > Step 1 and step 2 are to be repeated alternatingly until the right side of the (differentiated) output equation shows a direct dependence on a control variable. (The number of differentiations, which were carried out, corresponds to the differential order)

- **Step 3:**

  Carry out pole placement.
  The number of poles should comply with the differential order.

- **Step 4:**

  Set up the control law by solving the equations regarding $u(t)$.

2 Preparation for Experiment

(At home before you start the experiment)

Attach the calculations for this chapter to the experiment results!

2.1. For the multivariable, coupled system in state-space description

$$\underline{\dot{x}} = \begin{bmatrix} -2 & 1 \\ 1 & -3 \end{bmatrix} \underline{x} + \begin{bmatrix} 1 & 2 \\ 3 & 0 \end{bmatrix} \underline{u}$$

$$\underline{y} = \begin{bmatrix} 1 & 2 \\ 3 & 4 \end{bmatrix} \underline{x}, \qquad \textbf{Eq. 1}$$

a decoupling control system must be designed. Use the step by step procedure of the theoretical introduction (6 steps) for your calculation.

Note: Use ℓ_1=1 and ℓ_2=3 in step 4.

2.2. Declare the gain parameters for the system simulation shown in Fig. 1 (3.1.1).

2.3. Evaluate a nonlinear controller by using feedback linearization described in theoretical introduction. The nonlinear equation system is

$$\dot{x} = x^2 + x \cdot u$$

$$y = x \qquad \textbf{Eq. 2}$$

The output equation in step 1 is

$$\dot{y} + y = w \qquad \textbf{Eq. 3}$$

3 Experiment Procedure

3.1 Decoupling of Multivariable Systems

3.1.1 The structure of Eq. 1 is represented by a block diagram shown in Fig. 1

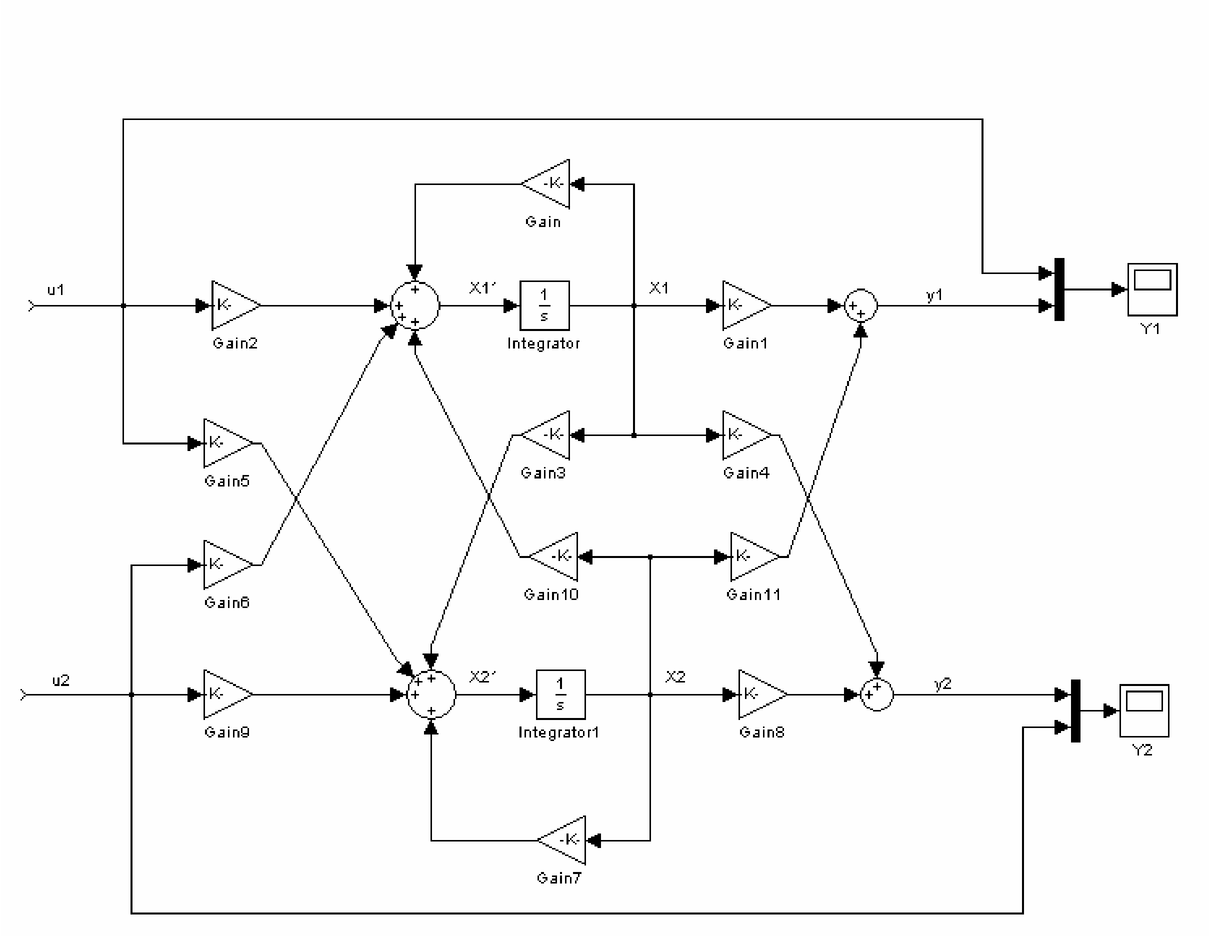


Fig. 1: Block Diagram of a Multivariable Coupled System

Enter the model shown in figure 1 in the SIMULINK editor and replace the gain parameters “-K-“ in the block diagram with the correct values in order to reproduce **Eq. 1**.

Print out the model with the correct gain values.

Print out the recorded curves for the following tasks (3.1.2 to 3.1.8).

3.1.2 Record $y_1(t)$ and $y_2(t)$ as well as $u_1(t)$ and $u_2(t)$ with $u_1(t) = \sigma(t)$ and $u_2(t) = 2\sigma(t)$.

3.1.3 Record $y_1(t)$ and $y_2(t)$ as well as $u_1(t)$ and $u_2(t)$, where $u_1(t) = \sigma(t)$ and $u_2(t)$ is a pulse code with the parameters shown in Fig. 2

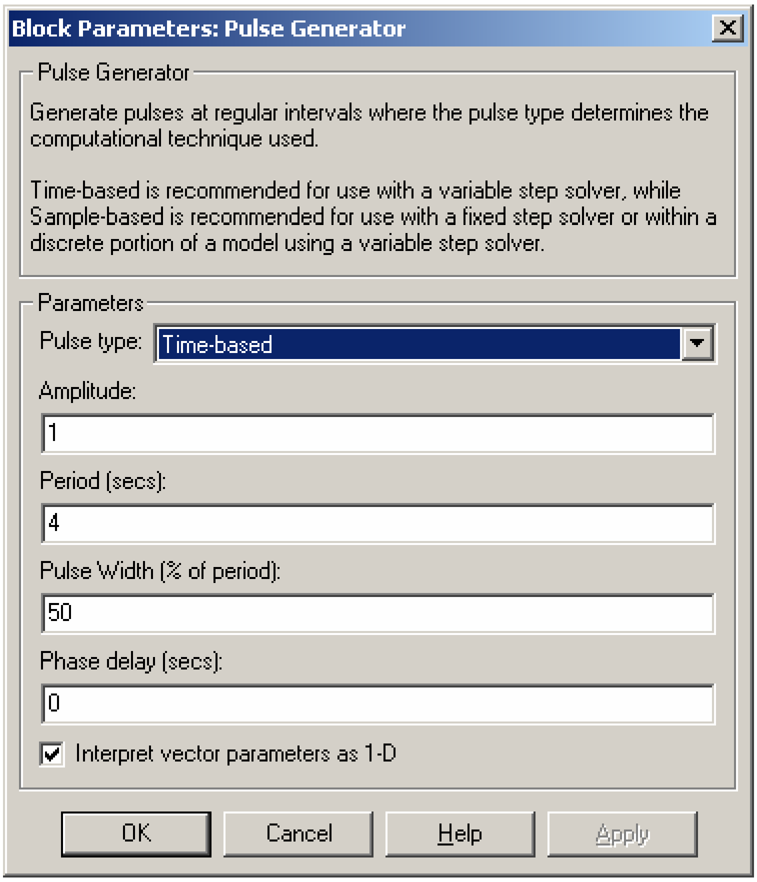


Fig. 2: Block Parameters: Pulse Generator

3.1.4 Extend the current system by a decoupling controller calculated in your preparation.

Note: Use the "Fcn"-block from "Function & Tables"-Library and the "Mux"-block from the "Signals & Systems"-Library to build the controller.

Print out the model extended with the decoupling controller.

3.1.5 Record $y_1(t)$ and $y_2(t)$ as well as $w_1(t)$ and $w_2(t)$ with $w_1(t)=\sigma(t)$ and $w_2(t)=2\sigma(t)$.

3.1.6 Record $y_1(t)$ and $y_2(t)$ as well as $w_1(t)$ and $w_2(t)$, where $w_1(t)=\sigma(t)$ and $w_2(t)$ is a pulse code in accordance to exercise 3.1.3 or **Fig. 2**.

3.1.7 Design a corresponding simplified system, record and compare the outputs with the results from task 3.1.5. The signals for the reference variables are $w_1(t)=\sigma(t)$ and $w_2(t)=2\sigma(t)$. Print out the model of the simplified system.

3.1.8 Record the output of the simplified system with the following signals for the reference variables: $w_1(t)=\sigma(t)$ and $w_2(t)$ is a pulse code in accordance with **Fig. 2**. Compare the recorded curves with the results from task 3.1.6.

3.2 Nonlinear compensation control

3.2.1 The block diagram in Fig. 3 is described by following nonlinear differential equation system

$$\dot{x} = x^2 + x \cdot u$$
$$y = x$$

Eq. 2

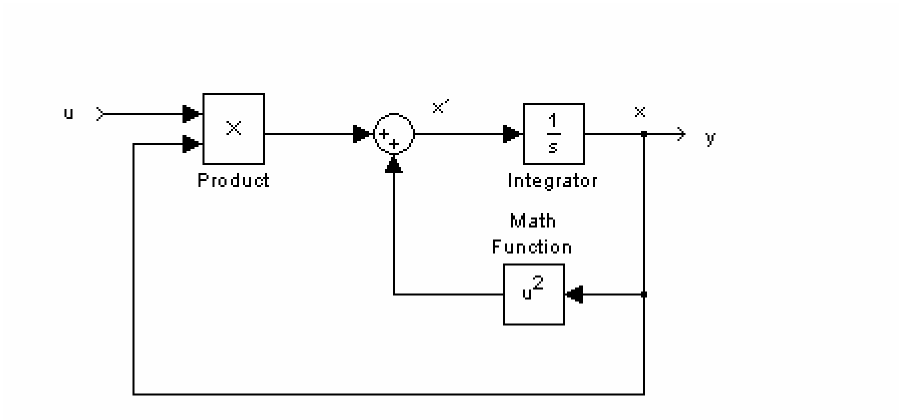


Fig. 3: Block Diagram of a Nonlinear System

The initial condition of the integrator is $x_0 = 0$.

Enter the model shown in figure 3 in the SIMULINK editor. Record $y(t)$ where $u(t)=\sigma(t)$. Give a reason, why $y(t)$ is zero during the simulation.

Print out the recorded curves for the following tasks (3.2.2 to 3.2.3).

3.2.2 Double click on the integrator and change the initial condition of x_0 from $x_0 = 0$ into $x_0 = 10^{-5}$. Record $y(t)$ where $u(t) = \sigma(t)$ as well as $u(t) = 8\sin\left(\frac{\pi}{2}t\right)$. Set the following parameters for the “Sine Wave”-block:

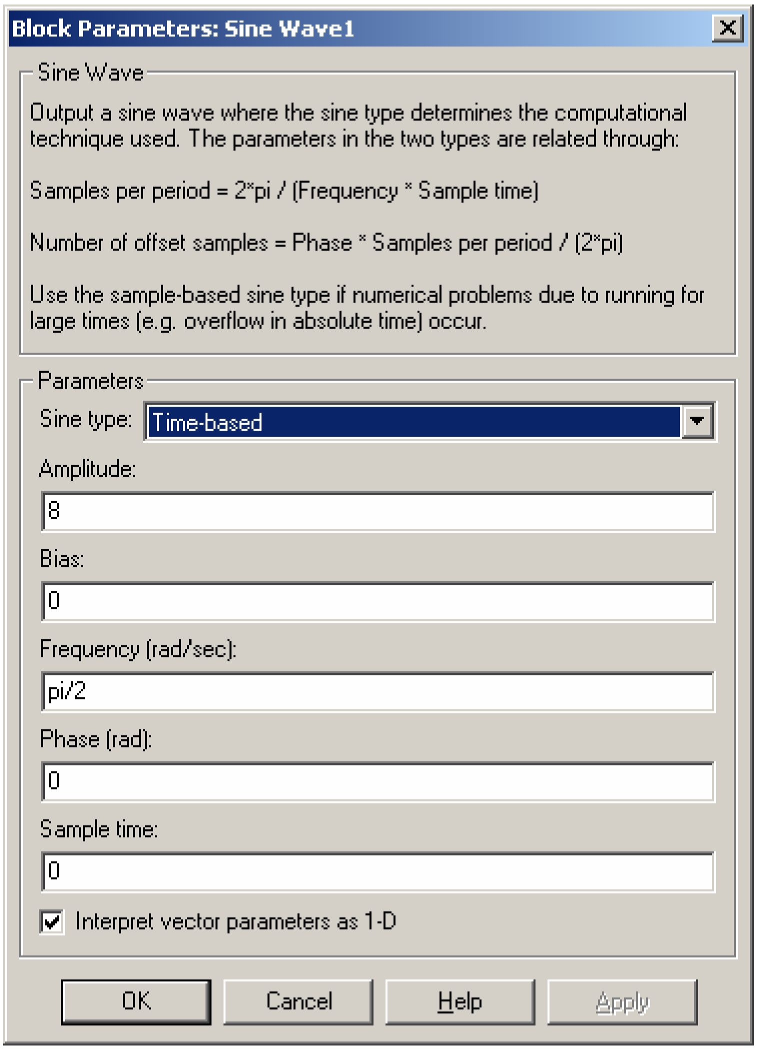


Fig. 4: Sine wave parameters

Is the system stable?

3.2.3 Extend the nonlinear system by calculated feedback controller in 2.3 and record $y(t)$ where $w(t) = \sigma(t)$ as well as $u(t) = \sin(t)$.

Is the system stable now? Why is the amplitude of output signal lower than the amplitude of input signal?

Note: Use the “Product”-block from “Math”-Library with parameters according to Fig. 5 to realize the division.

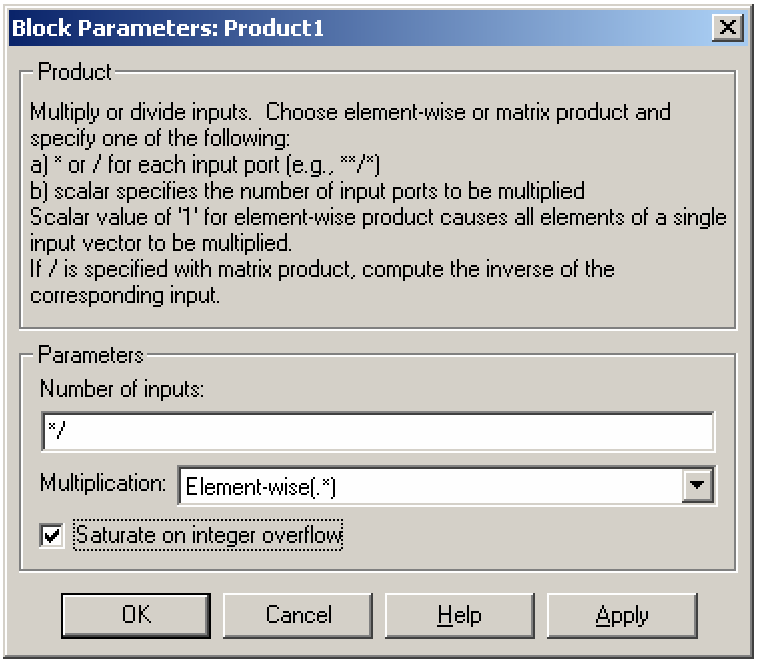


Fig. 5: Product Parameters

4 Decoupling and feedback linearization

4.1 Calculate the differential order δ_1 of the first subsystem y_1 of the following system given in state space form.

$$\underline{\dot{x}} = \begin{bmatrix} -3 & 2 \\ 1 & -5 \end{bmatrix} \underline{x} + \begin{bmatrix} 4 & 7 \\ 2 & 0 \end{bmatrix} \underline{u}$$
$$\underline{y} = \begin{bmatrix} 4 & -1 \\ 3 & 1 \end{bmatrix} \underline{x} + \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} \underline{u}$$

4.2 Use the principle of feedback linearization to evaluate a nonlinear controller for the following nonlinear system.

$$\dot{x} = \sin(x) + u$$
$$y = x^2$$

After applying the controller, the overall system should behave like a PT_1-Element.

$$\dot{y} + y = w$$

Reinsert the equation of your controller into the equation of the system to verify your calculation.