7 Assignment

7.1 Control of a third-order process

7.1.1 Introduction

In Canvas, you will find the Simulink file `reg_sys.slx` which will be used to simulate the feedback control system for the third-order process:

$$H(s) = \frac{0.1}{(15s+1)(10s+1)(5s+1)} = \frac{0.1}{750s^3 + 275s^2 + 30s + 1} \tag{7.1}$$

We will assume that the measuring instrument (situated on the feedback loop) has a time constant of 2 seconds and a static gain of 1, so that it is described by a transfer function:

$$M(s) = \frac{1}{1+2s} \tag{7.2}$$

A PID with filtered derivative effect has been implemented (red block). By double clicking on it, you can modify its parameters. You will see that the option of specifying the integrator limits is available, which you can do for example by replacing the values under `Max␣paadrag/Min␣paadrag` from `Inf/-Inf` to `100/-100`. This is done to avoid that the integrator keeps integrating the error signal when the actuators are already working at their maximum value, which is technically known as *anti-windup*. The reference signal goes from 0 to 1 at time t = 10 seconds, and then back to 0 at time t = 210 seconds.

7.1.2 Routh-Hurwitz criterion

Let us consider the following proportional controller transfer function, where K_p is the proportional gain[1]:

$$C(s) = K_p \tag{7.3}$$

[1] Here are a few cool commands you might want to try in MATLAB, although you will most likely not use them in this exercise: lorenz, image, penny, why, xpbombs.

Question 7.1

Use the Routh-Hurwitz criterion to determine the inequalities that must be satisfied by K_p in order to get BIBO stability of the closed-loop transfer function:

$$G(s) = \frac{H(s)C(s)}{1 + H(s)C(s)M(s)}$$

Then, simulate the system with a value of K_p slightly lower and slightly bigger than the upper bound that you computed. Do the simulations confirm the theoretical analysis?

Hint: If you compute $H_{cl}(s)$ correctly, then its denominator will have the form $a_0s^4 + a_1s^3 + a_2s^2 + a_3s + a_4$. It might be convenient to divide every coefficient by a_0 to get a polynomial in the form $s^4 + b_1s^3 + b_2s^2 + b_3s + b_4$ before building the Routh-Hurwitz table. Moreover, something to remember to ease the calculations is that you can always multiply all the numbers in the same row of the table by the same positive number, without changing the final inference obtained from the table.

Solution: The closed-loop transfer function is:

$$H_{cl}(s) = \frac{\frac{0.1}{750s^3+275s^2+30s+1}K_p}{1 + \frac{0.1}{750s^3+275s^2+30s+1}K_p\frac{1}{1+2s}} = \frac{0.1K_p\,(1+2s)}{(750s^3 + 275s^2 + 30s + 1)\,(1+2s) + 0.1K_p}$$

The denominator of $H_{cl}(s)$ is:

$$1500s^4 + 1300s^3 + 335s^2 + 32s + 1 + 0.1K_p$$

By dividing everything by 1500, we obtain the following polynomial:

$$s^4 + \frac{13}{15}s^3 + \frac{67}{300}s^2 + \frac{8}{375}s + \frac{(1+0.1K_p)}{1500}$$

We can build the first two rows of the Routh-Hurwitz table from the coefficients of the above polynomial, and then multiply the second row by 15 to obtain:

1	$\frac{67}{300}$	$\frac{(1+0.1K_p)}{1500}$
$\frac{13}{15}$	$\frac{8}{375}$	0

$\Rightarrow$

1	$\frac{67}{300}$	$\frac{(1+0.1K_p)}{1500}$
13	$\frac{8}{25}$	0

Table 7.1: Routh-Hurwitz table (Rows 1-2).

So far, it is good, as both 1 and 13 are positive. The elements in the third row are calculated as:

$$\frac{67}{300} - \frac{1 \cdot \frac{8}{25}}{13} = \frac{31}{156}$$

$$\frac{(1+0.1K_p)}{1500} - \frac{1 \cdot 0}{13} = \frac{(1+0.1K_p)}{1500}$$

1	$\frac{67}{300}$	$\frac{(1+0.1K_p)}{1500}$		1	$\frac{67}{300}$	$\frac{(1+0.1K_p)}{1500}$
13	$\frac{8}{25}$	0	$\Rightarrow$	13	$\frac{8}{25}$	0
$\frac{31}{156}$	$\frac{(1+0.1K_p)}{1500}$	0		31	$\frac{13(1+0.1K_p)}{125}$	0

Table 7.2: Routh-Hurwitz table (Rows 1-3).

so we obtain (the table on the right is obtained by multiplying the third row by 156):

The first column of the table still contains all positive elements, so we can continue further by computing the only non-zero element of the fourth row of the table:

$$\frac{8}{25} - \frac{13 \cdot 13\left(1 + 0.1K_p\right)}{125 \cdot 31} = \frac{1071 - 16.9K_p}{3875}$$

so we obtain (the table on the right is obtained by multiplying the fourth row by 3875): Finally,

1	$\frac{67}{300}$	$\frac{(1+0.1K_p)}{1500}$		1	$\frac{67}{300}$	$\frac{(1+0.1K_p)}{1500}$
13	$\frac{8}{25}$	0		13	$\frac{8}{25}$	0
31	$\frac{13(1+0.1K_p)}{125}$	0	$\Rightarrow$	31	$\frac{13(1+0.1K_p)}{125}$	0
$\frac{1071-16.9K_p}{3875}$	0	0		$1071 - 16.9K_p$	0	0

Table 7.3: Routh-Hurwitz table (Rows 1-4).

we can complete the table by computing the last element (which is reported hereafter multiplied by 125 and divided by 13): which means that the following two inequalities must be satisfied:

1	$\frac{67}{300}$	$\frac{(1+0.1K_p)}{1500}$
13	$\frac{8}{25}$	0
31	$\frac{13(1+0.1K_p)}{125}$	0
$1071 - 16.9K_p$	0	0
$1 + 0.1K_p$	0	0

Table 7.4: Routh-Hurwitz table (Rows 1-5).

$$\begin{cases} 1071 - 16.9K_p > 0 \\ 1 + 0.1K_p > 0 \end{cases} \qquad \begin{cases} K_p < 1071/16.9 \cong 63.3 \\ K_p > -1/0.1 = -10 \end{cases} \tag{7.4}$$

7.1.3 Manual tuning of the PID controller

You will now tune manually the controller (Table 7.5 resumes the effect of increasing each individual parameter).

For a PI controller, you should first increase the proportional gain K_p until satisfactory stability is seen in the response of $y(t)$ to the step change in $r(t)$. Then increase the integral gain K_i

↑ parameter	Rise time	Overshoot	Settling time	Steady-state error	Stability
K_p	Decrease	Increase	Small change	Decrease	Degrade
K_i	Decrease	Increase	Increase	Eliminate	Degrade
K_d	Minor change	Decrease	Decrease	No effect	Improve (if K_d small)

Table 7.5: Effect of increasing PID parameters.

(which means decreasing the integral time T_i) until the closed-loop system is almost unstable. Finally, tune again K_p until the stability is deemed satisfactory.

Question 7.2

Let `Max_paadrag/Min_paadrag` be `Inf/-Inf` and find values for K_p and T_i for a PI controller. Remember that there is not a single correct solution, but you need to find some values that provide a response that you consider to be satisfactory. Try to get a fast response without too much overshoot. Include the parameter values and the obtained plots.

Solution: A satisfactory response is obtained with $K_p = 30$, as shown in Fig. 7.1. By decreasing the integral time until $T_i = 15$ s, the system is driven to the edge of instability, as shown in Fig. 7.1. Then, further refining of the gain K_p leads to the response in 7.3 which, in my opinion, is satisfactory and, hopefully, also in yours.

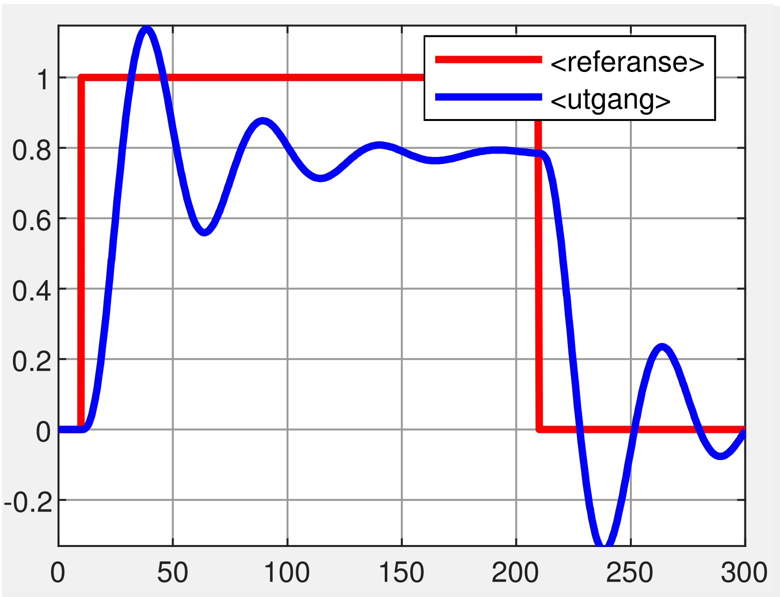

Figure 7.1: Reference signal and output signal, P-controller: $K_p = 30$, $T_i = \infty$.

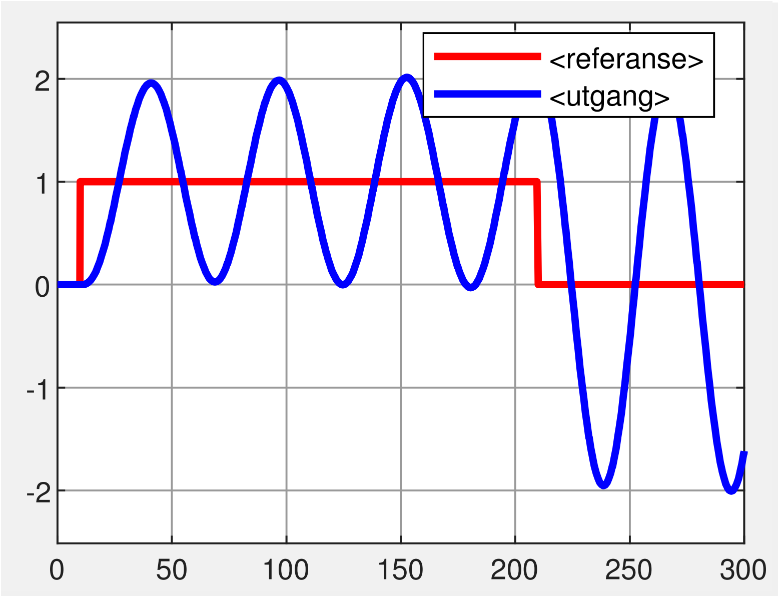

Figure 7.2: Reference signal and output signal, PI-controller: $K_p = 30$, $T_i = 15$.

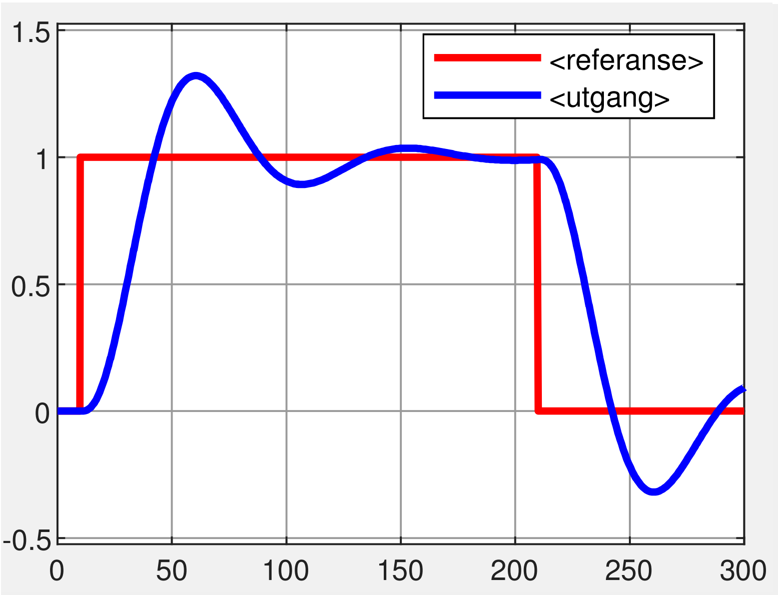

Figure 7.3: Reference signal and output signal, PI-controller: $K_p = 10$, $T_i = 15$.

For a PID controller, increase the gain K_p until satisfactory stability is seen in the response of $y(t)$ to the step change in $r(t)$. Then, activate the integral and derivative terms and tune them taking into account Table 7.5.

Question 7.3

Now, find a good value for the parameter T_d to be used in the PID controller. You will need to choose also the parameter T_f: a rule of thumbs is to choose is approximately as $T_f \approx 0.2T_d$. Compare the response obtained using the PID controller with the response obtained with the PI controller.

Solution: I was quite satisfied with the response obtained with $T_d = 10$ s and $T_f = 2$ s, as shown in Fig. fig:scope4. In fact, the derivative action damped quite a lot the overshoot :D

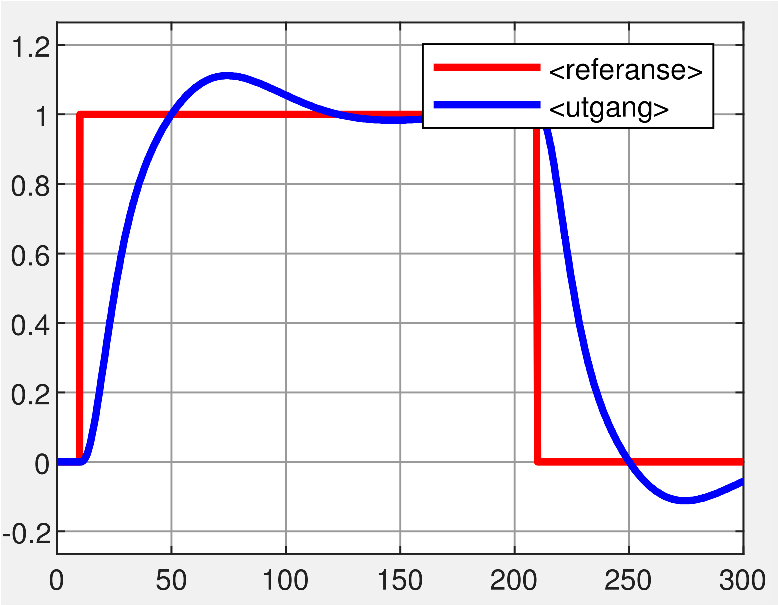

Figure 7.4: Reference signal and output signal, PID-controller: $K_p = 10$, $T_i = 15$, $T_d = 10$, $T_f = 2$.

7.1.4 Control without anti-windup

In order for you to be able to see the effect of the integrator limitation, you must increase the reference change so that the actuator goes into saturation.

You must still let `Max_paadrag/Min_paadrag` to be `Inf/-Inf`. To illustrate the phenomenon, you should set $T_d = 0$, but keep the other parameters that you found. The reason for this is that the derivative effect makes the input very hectic, so that it becomes hard to interpret from the plots what is going on.

Question 7.4

Change `Ny_Verdi` in the reference block to 12, so that the reference goes from 0 to 12 at t

= 0 and back to 0 at t = 210 s. Then simulate the model, and observe the input and output signals. Explain in words what is happening.

Solution: Fig. 7.5 shows the obtained plots. You can see from subfigure 6 that the output does not reach the reference, and that the error is bigger than 1 (subfigure 1). The reason why this happens is that the control input (blue line in subfigure 4) is limited to 100 by the saturation, although the computed control action keeps increasing (red line in subfigure 4) due to the effect of the integrator (subfigure 5). When the reference finally goes down again (at time t = 210 seconds), the value of the integrator must first be reduced before the computed control input reduces as well. The problem is that the integrator has accumulated a value of almost 500 (subfigure 5), and it takes some time before the integrator discharges (this is the phenomenon known as *windup*, obviously we refer to any technique able to avoid this phenomenon as *anti-windup*). For this reason, the output is kept high for a long time after the reference has gone to zero, which is undesirable.

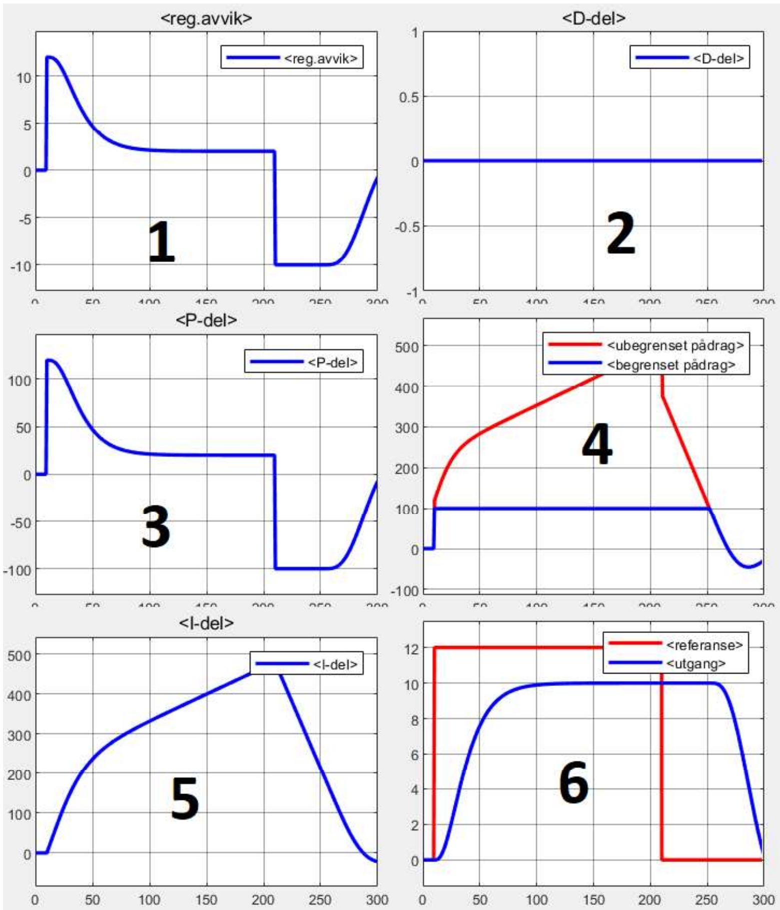

Figure 7.5: PI control without anti-windup under big reference changes.

7.1.5 Control with anti-windup

Specify now `Max paadrag/Min paadrag` to 100/-100.

Question 7.5

Use the PID parameters from Question 7.3. Simulate again and observe the computed control input and the obtained output response. Describe what is happening.

Solution: Fig. 7.6 shows the results obtained using the PID control with the implemented anti-windup. From subfigure 6, you see that the output still does not reach the desired reference $r(t) = 12$: there is nothing you can do about this, as this reference is beyond what you can reach with the available input (compare this with what would happen asking a standard car to reach the desired speed of 500 km/h). However, the value of the integral action is now limited to 100 (subfigure 5). The computed control action is still a little beyond the maximum available input due to the effect of the proportional action (see subfigures 3 and 4). As soon as the reference signal goes back to zero (at time t equal to 210 seconds), the proportional action is reduced to a negative number (subfigure 3) due to the change of sign in the tracking error (subfigure 1), and the integral action starts decreasing (subfigure 5). In this way, the control action reduces immediately (subfigure 4), which leads to an immediate reduction of the output (subfigure 6).

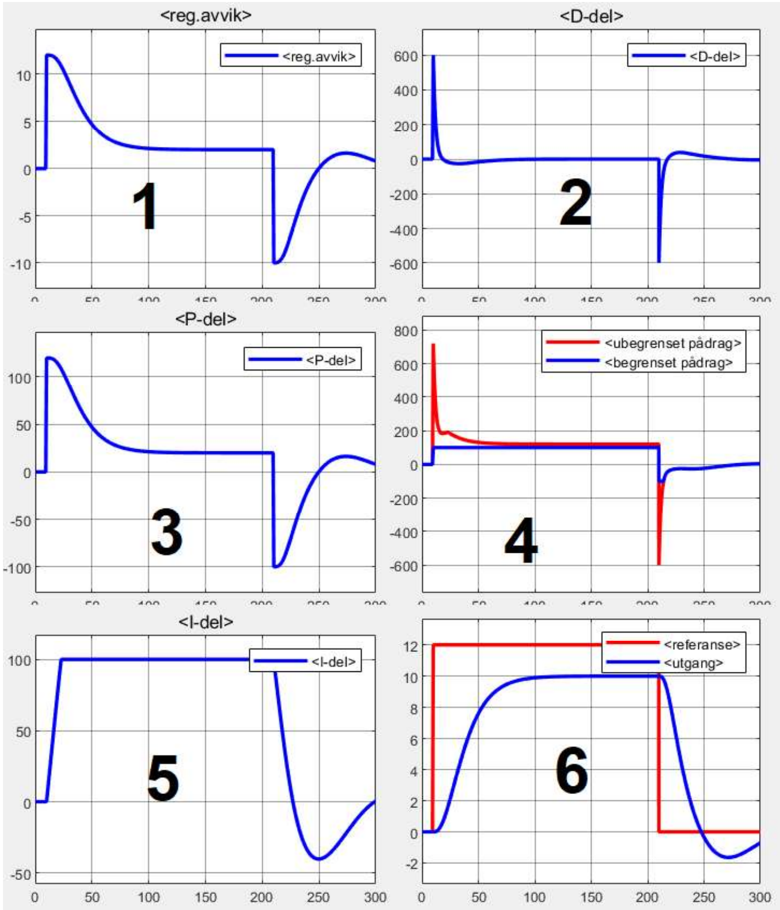

Figure 7.6: PID control with anti-windup under big reference changes.

7.2 Control of first-order processes

You will now design a controller for the following first-order system:

$$H(s) = \frac{0.5}{200s + 1} \tag{7.5}$$

7.2.1 P-controller

Let us start first with the proportional controller (K_p is the proportional gain):

$$C(s) = K_p \tag{7.6}$$

Question 7.6

Find out the closed-loop transfer function $H_{cl}(s) = Y(s)/R(s)$, where $Y(s)$ is the output of $H(s)$ and $R(s)$ is the reference signal, under the assumption that the input to the controller is $E(s) = R(s) - Y(s)$. Put $H_{cl}(s)$ in the standard first-order form and find the expression of how the pole of $H_{cl}(s)$ moves when K_p increases.

Solution: The closed-loop transfer function is:

$$H_{cl}(s) = \frac{H(s)C(s)}{1 + H(s)C(s)} = \frac{\frac{0.5}{2s+1}K_p}{1 + \frac{0.5}{2s+1}K_p} = \frac{0.5K_p}{200s + 1 + 0.5K_p}$$

which can be put in the standard first-order form as follows

$$H_{cl}(s) = \frac{\frac{0.5K_p}{1+0.5K_p}}{1 + \frac{200}{1+0.5K_p}s}$$

The single pole is given by:

$$1 + \frac{200}{1 + 0.5K_p}s = 0 \quad \Rightarrow \quad s = -\frac{1 + 0.5K_p}{200}$$

It can be seen that the pole moves towards the left of the complex plane when K_p increases, which means that at least theoretically it is possible to have an arbitrarily fast response by increasing the gain K_p.

Question 7.7

From $H_{cl}(s)$, find an expression of how the closed-loop time constant τ_{cl} and the closed-loop static gain $H_{cl}(0)$ vary as a function of K_p.

Solution: Given the above standard first-order form, by comparison with:

$$H_{cl}(s) = \frac{H_{cl}(0)}{1 + s\tau_{cl}}$$

we find out that:

$$H_{cl}(0) = \frac{0.5K_p}{1 + 0.5K_p}$$

$$\tau_{cl} = \frac{200}{1 + 0.5K_p}$$

Question 7.8

What value of K_p should you use to obtain $\tau_{cl} = 100$ s?

Solution: In order to obtain $\tau_{cl} = 100$ s, one must choose:

$$\frac{200}{1 + 0.5K_p} = 100 \quad \Rightarrow \quad K_p = 2$$

Question 7.9

Implement the feedback control system comprising process + controller (with the K_p computed as answer to Question 7.8). Use a unit step signal and simulate with a total length of 500 s. Use a fixed-step Euler integration method with fixed-step size 1 s (you can change this parameter by right-clicking on a blank point of the Simulink scheme, then select `Model configuration parameters`). Provide two plots: i) comparison between the reference $r(t)$ and the output $y(t)$; ii) usage of the control input $u(t)$. What are the closed-loop time constant and closed-loop static gain as read from the plots? Do they correspond to the theoretical ones?

Solution: The feedback control system comprising process + controller is shown in Fig. 7.7. The plots obtained from the simulation are the ones shown in Figs 7.8-7.9. The theoretical closed-loop static gain is:

$$H_{cl}(0) = \frac{0.5K_p}{1 + 0.5K_p} = \frac{0.5 \cdot 2}{1 + 0.5 \cdot 2} = \frac{1}{2} = 0.5$$

which is the final value read from the plot of $y(t)$. On the other hand, 63 % of the final value corresponds to 0.315, which is reached at approx. 100 s, thus confirming that the closed-loop time constant is $\tau_{cl} = 100$ s.

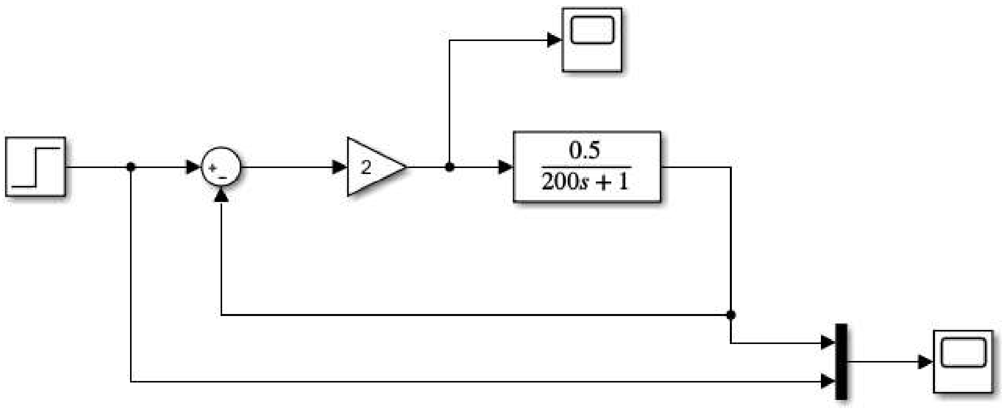

Figure 7.7: Simulink implementation of the control scheme.

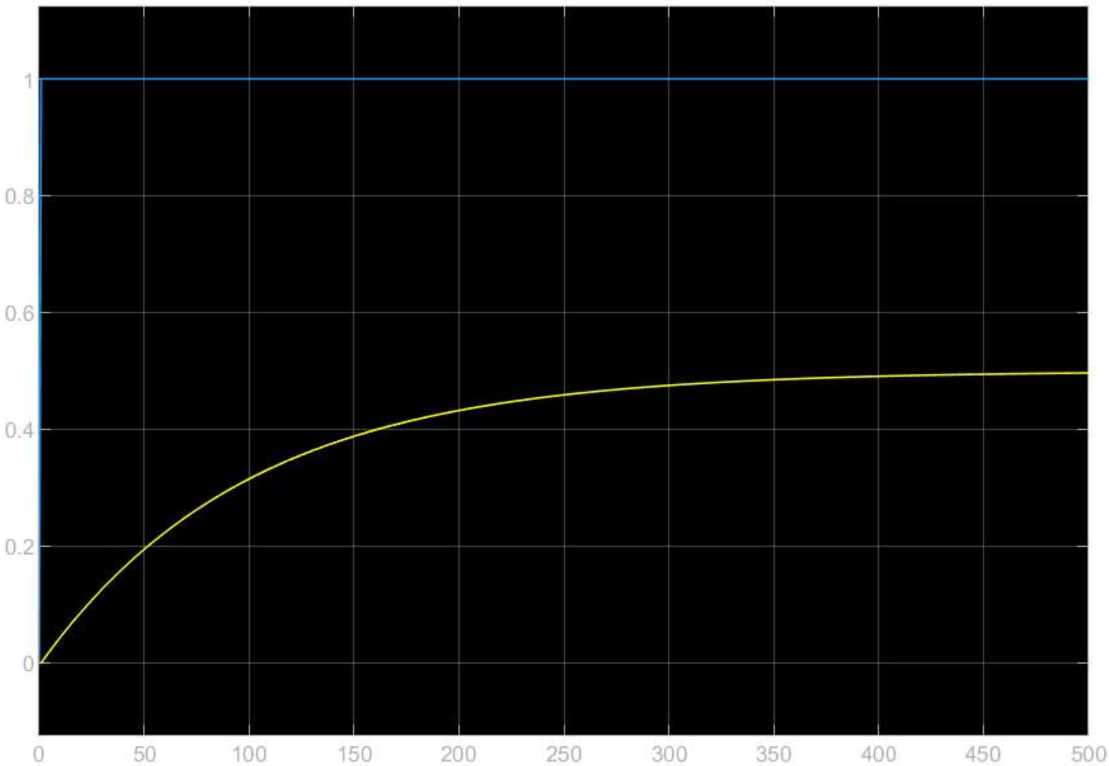

Figure 7.8: Reference $r(t)$ and output $y(t)$, $K_p = 2$.

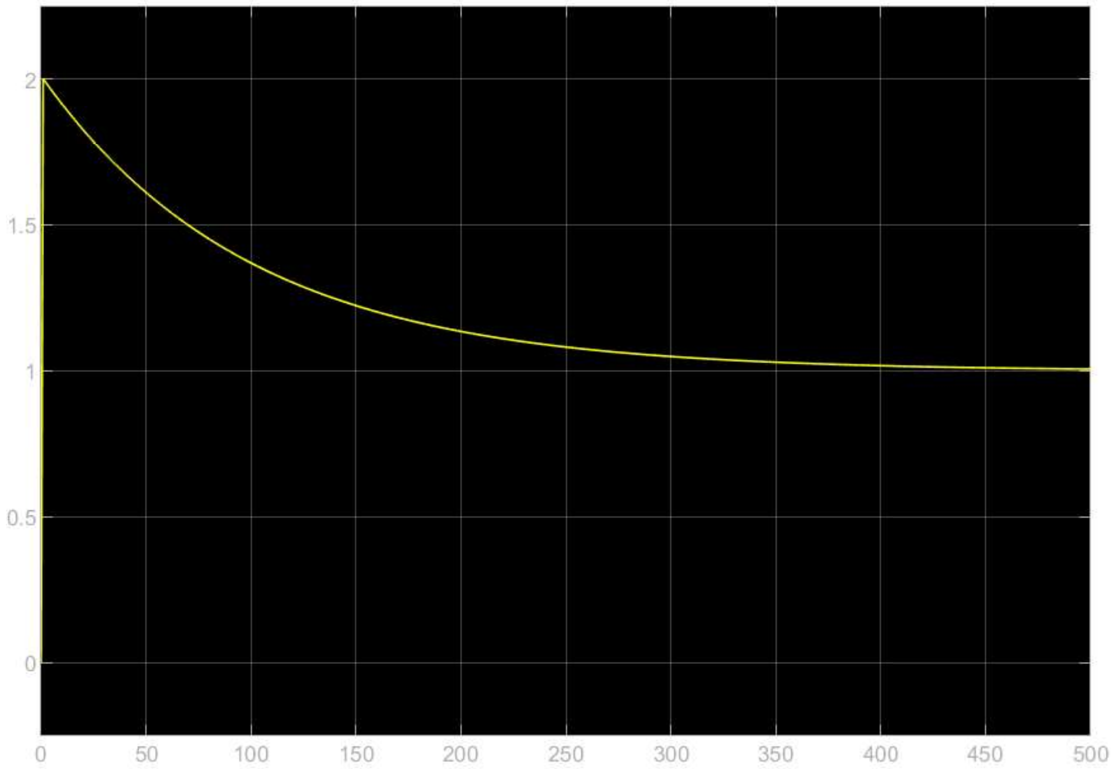

Figure 7.9: Control input $u(t)$, $K_p = 2$.

Question 7.10

What is the steady-state error $e_\infty = \lim_{t\to\infty} e(t)$? Verify this value by applying the final value theorem either directly to $E(s)$ or indirectly to $Y(s)$, which means that after computing $y_\infty = \lim_{t\to\infty} y(t)$, you calculate $e_\infty = r_\infty - y_\infty$.

Solution: The steady-state error, as read from Fig. 7.8, is $e_\infty = 0.5$. Then we have:

$$e_\infty = \lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} s\left(R(s) - Y(s)\right)$$
$$= \lim_{s\to 0} s\left(R(s) - H_{cl}(s)R(s)\right) = \lim_{s\to 0} s\left(1 - H_{cl}(s)\right)\frac{1}{s} = 1 - H_{cl}(0) = 1 - 0.5 = 0.5$$

7.3 PI-controller

To eliminate the steady-state error to a step in the reference signal $R(s)$, you will use a PI controller given by:

$$C(s) = K_p \frac{1 + sT_i}{sT_i} \tag{7.7}$$

where K_p is the proportional gain and T_i is the integral time.

Question 7.11

Find the closed-loop transfer function $Y(s)/R(s)$.

Solution: The closed-loop transfer function is given by:

$$H_{cl}(s) = \frac{H(s)C(s)}{1 + H(s)C(s)} = \frac{0.5K_p(1 + sT_i)}{sT_i(200s + 1) + 0.5K_p(1 + sT_i)}$$

Question 7.12

Prove analytically that the steady-state error to a step change in the reference signal $R(s)$ is $e_\infty = 0$ when you use a PI controller *(as long as the closed-loop system is BIBO stable).*

Solution: The application of the final value theorem leads to:

$$\begin{aligned} y_\infty &= \lim_{t \to \infty} y(t) = \lim_{s \to 0} sY(s) = \lim_{s \to 0} sH_{cl}(s)R(s) \\ &= \lim_{s \to 0} s \frac{0.5K_p(1 + sT_i)}{sT_i(200s + 1) + 0.5K_p(1 + sT_i)} \frac{1}{s} = \frac{0.5K_p}{0.5K_p} = 1 \end{aligned}$$

which means that $e_\infty = r_\infty - y_\infty = 1 - 1 = 0$.

Question 7.13

Confirm the theoretical result from the previous question by simulating the feedback control system with the PI controller in Simulink. Choose values of K_p and T_i so that the closed-loop system is BIBO stable. Let the length of the simulation be 500 seconds.

Solution: The implementation of the PI controller in Simulink is shown in Fig. 7.10, where $K_p = 30$ and $T_i = 300$ have been used. From Fig. 7.11, we see that the output reaches the reference with zero steady-state error.

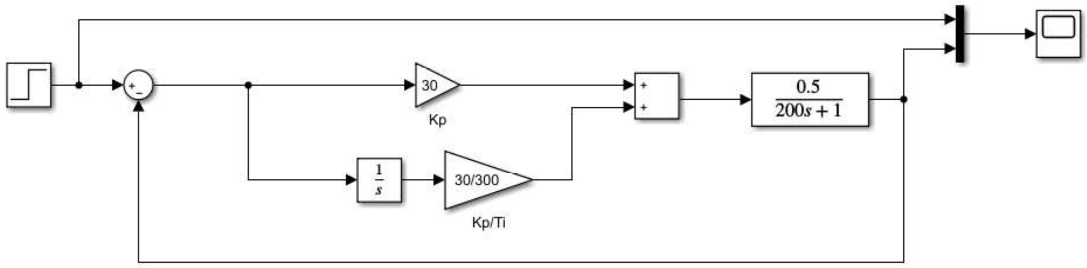

Figure 7.10: Simulink implementation of the PI controller, $K_p = 30$, $T_i = 300$.

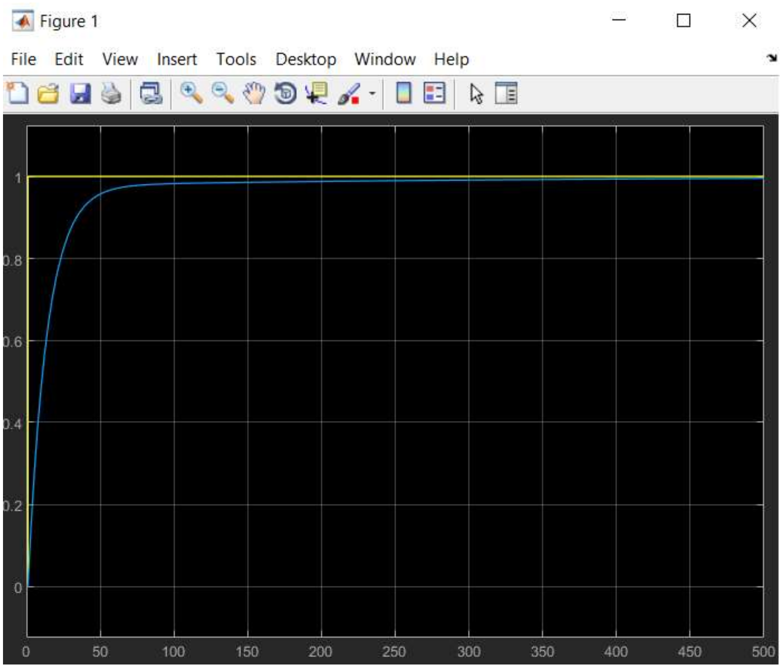

Figure 7.11: Comparison between $r(t)$ and $y(t)$ using the PI controller, $K_p = 30$, $T_i = 300$.

Question 7.14

Assume that the closed-loop system must have a natural frequency $\omega_0 = 0.01$ rad/s and a damping factor $\xi = 0.75$. How should T_i and K_p be selected to satisfy these specifications? Use Simulink to simulate the closed-loop response (use a simulation length of 500 seconds), and discuss how the zero at $s = -1/T_i$ in $H_{cl}(s)$ affects the response.

Solution: We must put $H_{cl}(s)$, which we calculated at Question 7.11 in a standard form:

$$H_{cl}(s) = \frac{0.5K_p(1+sT_i)}{200T_i s^2 + (T_i + 0.5K_pT_i)\,s + 0.5K_p} = \frac{1+sT_i}{1+\frac{T_i+0.5K_pT_i}{0.5K_p}s + \frac{200T_i}{0.5K_p}s^2} = \frac{1+sT_i}{1+\frac{2\xi}{\omega_0}s+\frac{s^2}{\omega_0^2}} \quad (7.8)$$

The comparison between the transfer functions, taking into account $\xi = 0.75$ and $\omega_0 = 0.01$ rad/s, leads to the following system of equations:

$$\begin{cases} \frac{2\xi}{\omega_0} = 150 = \frac{T_i+0.5K_pT_i}{0.5K_p} \\ \frac{1}{\omega_0^2} = 10000 = \frac{200T_i}{0.5K_p} \end{cases}$$

Then we get:

$$10000 = \frac{400T_i}{K_p} \Rightarrow K_p = 0.04T_i$$

and:

$$150 = \frac{T_i + 0.5 \cdot 0.04T_i^2}{0.5 \cdot 0.04T_i} = 50 + T_i \Rightarrow T_i = 100 \Rightarrow K_p = 4$$

The simulation of the closed-loop response is shown in Fig. 7.12. Since the zero is in the left half-plane, its effect will be to speed up the response, while at the same time increasing the overshoot.

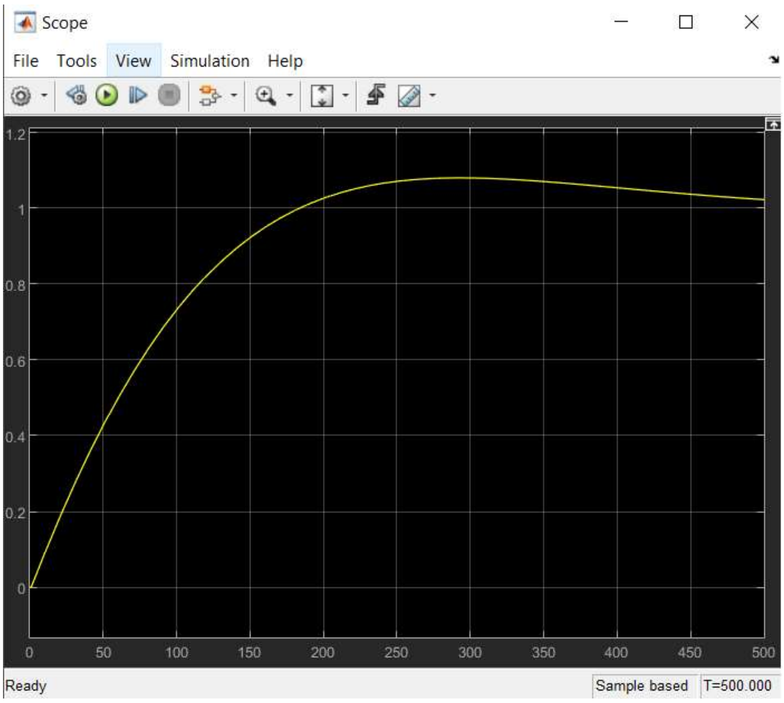

Figure 7.12: Simulation using the PI controller, $K_p = 4$, $T_i = 100$.