AUTOMATIC CONTROL

Lecture Notes

Tore Hägglund

Lund 2021

Department of Automatic Control
Lund University, Faculty of Engineering
Box 118
221 00 LUND

Contents

Preface

This compendium consists of lecture notes from the Basic Course in Automatic Control at Lund University. The compendium is intended to complement a control text book, exercise set, collection of formulae and laboratory exercise guidelines which are used in the course.

The course consists of 15 lectures, where the last one is a pure repetition. Because of this, the compendium contains notes from only the first 14 lectures.

Tore Hägglund

Lecture 1

Introduction—The PID Controller

In this lecture, we give a first introduction to automatic control and show how one can describe problems from this field. Subsequently we introduce the reader to one of the simpler and most commonly used controller—the PID controller.

1.1 What is Automatic Control?

Automatic control is the technology used to control processes in order to achieve desired behaviors. In many principal disciplines of science, such as physics and mechanics, we learn to understand how nature works. We observe, and mathematically describe, various phenomenons in our environment. In automatic control we use this knowledge to control these phenomenons.

From mechanics we have for instance obtained equations, which describe the velocity of a cart rolling down an inclined plane. In control theory the corresponding equations can be used to device an automatic cruise controller for motorcars, granting them constant speed, despite variations in the incline of the road.

Automatic control is used in a variety of ways. Here are some examples:

- The auto pilots of aircrafts and motorships are devices providing the vessels with constant bearings, despite disturbances in form of winds and currents.
- To keep a constant temperature in buildings, despite variations in the outdoor temperature and the number of persons in the building, is a classic control problem.
- In the process industry, e.g. in pulp mills, hundreds—or even thousands—of controllers maintain desired pressures, flows, temperatures, concentrations and levels.
- Modern cars contain many control systems for e.g. cruise control, anti-lock brake systems, power steering, emission reduction and air conditioning.
- Cameras contain control systems for e.g. auto focus and automatic exposure control.
- Automatic control is not limited to technological contexts. The human body, for instance, contains many control systems. One example is the temperature control, which ensures that the body temperature is held constant at 37° C, despite variations in the surrounding temperature and the work load of the body. Another example is the light control in the eye. The size of the pupil is automatically adjusted so that the illumination of the retina is held as constant as possible.

What is interesting, is that all these seemingly unrelated problems can be described and solved with a unified theory, taught in this course.

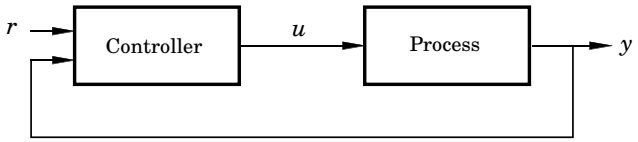

Figure 1.1 The simple feedback loop.

1.2 The Simple Feedback Loop

All the above stated control problems can be summarized by the block diagram shown in Figure 1.1. The control problems are solved by means of feedback.

The entity to be controlled is denoted y and referred to as the measurement signal or output. The signal used to affect the process is denoted u and referred to as the control signal or input. The signal constituting the reference value for the output is denoted r and called the reference value, or the setpoint. The notions of input and output are used in the literature, but are unfortunate, since they cause confusion. Among controller manufacturers, it is e.g. common to call the control signal output and the measurement signal input.

The purpose of the controller is to determine the control signal u such that the measurement signal y tracks the reference r as well as possible. The control problem would have been an easy one if the process was static, i.e. that there was a static relation between y and u such that

$$y(t) = f(u(t))$$

Most processes are, however, dynamic, i.e.

$$y(t) = f(u_{[-\infty,t]})$$

It is crucial to understand the dynamics of the process, in order to solve an automatic control problem. The next few lectures will treat process dynamics and different ways of representing them. The remainder of this lecture is, however, devoted to the controller and more specifically to describing the most commonly used controller, namely the PID controller.

1.3 The PID Controller

We shall now deduce the structure of the PID controller and thereby show that it is a natural extension of the very simplest controller, namely the on/off controller.

The On/Off Controller

The on/off controller is the simplest imaginable controller. Its control signal u is given by

$$u = \begin{cases} u_{\max} & e > 0 \\ u_{\min} & e < 0 \end{cases}$$

where e is the control error, i.e. the difference between the setpoint r and the measurement signal y.

$$e = r - y$$

The functionality of the on/off controller can also be described graphically, as shown in Figure 1.2.

A drawback with this controller is that it gives rise to oscillations in the control loop. In order for the controller to maintain a small difference between measurement signal and setpoint, it must constantly switch the control signal between the two levels $u_{\max}$ and $u_{\min}$. If we for instance control the speed of a car by means of the gas

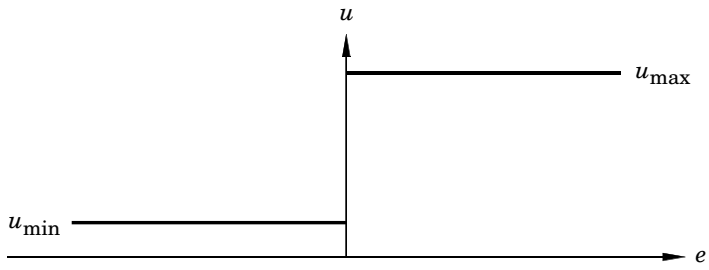

Figure 1.2 The control signal of the on/off controller.

pedal, while it can only take on the values "no gas" and "full gas", we will need to switch between these two values in order to keep the average speed at the setpoint. This is one way of driving, but the control action is poor.

The P Part

For large control errors it may be feasible to either release or fully depress the gas pedal. Consequently, the on/off controller performs well for large errors. The oscillations appear for small control errors and can be reduced by e.g. decreasing the controller gain for small control errors. This can be achieved by introducing a proportional band or a P controller. The control signal of the P controller is given by

$$u = \begin{cases} u_{max} & e > e_0 \\ u_0 + Ke & -e_0 \leq e \leq e_0 \\ u_{min} & e < -e_0 \end{cases}$$

where u_0 is the control signal corresponding to a zero control error and K is the gain of the controller. The P controller can also be described graphically, as shown in Figure 1.3.

The output of the P controller corresponds to that of the on/off controller for large control errors. For control errors of magnitude less than e_0, the control signal is, however, proportional to the control error.

For many controllers, the proportional band (PB) is given, rather than the gain. The relation between these two entities is given by

$$PB = \frac{100}{K}[\%]$$

The gain $K = 1$ hence corresponds to a proportional band $PB = 100\%$.

The P controller removes the oscillations, which were present during on/off control. Unfortunately this comes at the price. We are no longer granted a zero stationary error, or in other words, that the setpoint and measurement signal coincide

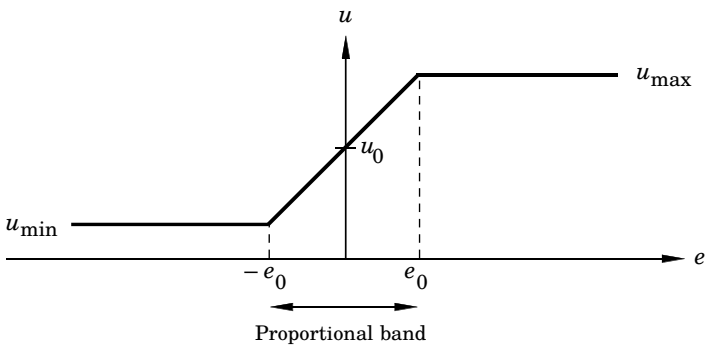

Figure 1.3 The control signal of the P controller.

when all signals in the control loop have reached constant values. This is easily realized by studying the control signal. For small control errors, the P controller works within its proportional band. The control error is then given by

$$e = \frac{u - u_0}{K}$$

In stationarity the control error becomes $e = 0$ if and only if at least one of the below criteria are fulfilled

1. K is infinitely large
2. $u_0 = u$

Alternative one, an infinite controller gain or a zero proportional band is equivalent to on/off control. This alternative is therefore not a good solution, since it leaves us with the initial oscillation problem. We are hence referred to alternative two, in order to eliminate the stationary control error. Here we can only eliminate the stationary control error if we can vary u_0 so that it becomes equal to control signal u for all values of setpoint r.

From the expression for the control error of the P controller we see that a higher controller gain K leads to a smaller control error. We also see that we minimize the maximal stationary control error by choosing u_0 in the center of the working range of the control signal. In most controllers, u_0 is consequently preset to $u_0 = 50\%$. In some controllers, it is possible to adjust the value of u_0. From the above discussion we see that u_0 should be chosen as close to the stationary value of u as possible.

The I Part

Rather than letting u_0 be a constant parameter, one can choose to adjust it automatically in order to achieve $u_0 = u$ when all signals in the control loop have reached constant values. This would eliminate the residual control error and is exactly what the integral part (I part) of a PI controller does. The control signal of a PI controller is given by

$$u = K\left(\frac{1}{T_i}\int e(t)dt + e\right)$$

where T_i is the integral time of the controller. The constant level u_0 of the P controller has thus been replaced by the term

$$u_0 = \frac{K}{T_i}\int e(t)dt$$

which is proportional to the *integral* of the control error. This is why the term is called the integral term or the integral part of the PID controller.

One can be convinced that the PI controller has the ability to eliminate residual control errors by studying the above control law. Assume that we have a stationary control error $e \neq 0$ despite the use of a PI controller. If the control error e is constant, the proportional part in the PI controller will also hold a constant value Ke. The integral part will, however, not be constant. It will increase or decrease, depending on the sign of the control error. If the control signal is changed, the measurement signal y of the process must sooner or later increase or decrease. Consequently, the error $e = r - y$ cannot be constant. Since this conflicts with the assumption of a stationary error, we have showed that we cannot have a non-zero stationary error, when the controller contains an integral part. The only occasion when all signals internal to the controller can be stationary, is when $e = 0$.

We have now showed that the PI controller solves the problem of a residual stationary error and that of oscillations resulting from on/off control. The PI controller is therefore a controller without any substantial shortcomings. It is generally sufficient when performance requirements are not extensive. Consequently, the PI controller is the by far most commonly used controller in industrial applications.

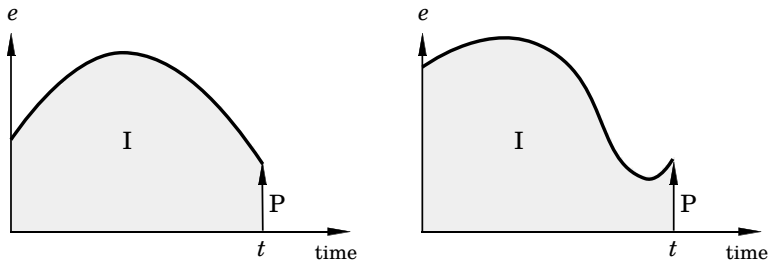

Figure 1.4 Two control cases where the output from a PI controller are equal at time t.

The D Part

One characteristic which limits the performance of the PI controller is that it only takes past and present control errors into account; it does not try to predict the future evolution of the control error. The problem is illustrated in Figure 1.4.

The two curves in Figure 1.4 show the evolution of the control error in the two cases. The P part of the controller is proportional to the control error at the present time instance t. This control error is equal for both figures. The integral part is proportional to the surface delimited by the control error curve. This implies that a PI controller yields the same control signal at time t for the two cases. An intelligent controller should, however, see that there is a big difference between the cases. In the left curve, the control error decreases rapidly and the control action should be deliberate, in order not to cause an overshoot. In the right curve, a decrease in the control error is followed by a sudden increase. Here, the controller should apply a large control signal in order to curtail the control error. The derivative part of the PID controller accomplishes exactly this type of compensation. It is proportional to the change rate of the control error, i.e. proportional to the time derivative of the error. See Figure 1.5.

The equation of the PID controller is given by

$$u = K\left(e + \frac{1}{T_i}\int e(t)dt + T_d\frac{de}{dt}\right)$$

where T_d is the derivative time of the controller.

The maximal benefit of the D part is obtained in cases where a lot can be earned by predicting the control error. This is the case for many temperature control applications. Due to the inertia of these systems it is necessary to abort heating in time. Slow heat conduction can otherwise result in rising temperatures, long after the seize of heating.

Everybody who has used a thick-bottomed pan for broiling has witnessed this phenomenon. It can take quite a while from the time instance when one turns down the temperature control knob, until the temperature in the pan actually begins to

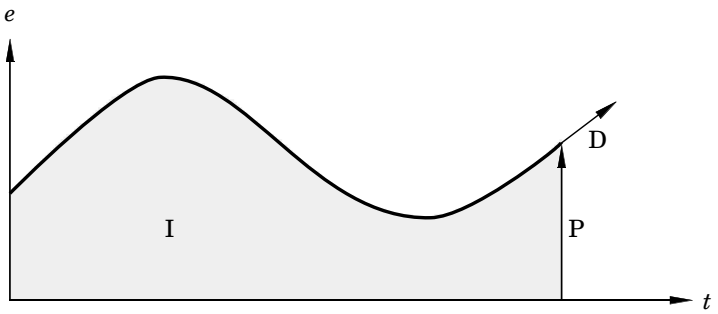

Figure 1.5 The integral part is proportional to the surface under the control error curve, the proportional part is proportional to the current control error and the derivative part is proportional to the change rate of the control error.

decrease. In the meanwhile the temperature can be subject to a significant overshoot if one is not careful with the temperature control.

The PID controller can be summarized by means of Figure 1.5. The proportional part provides the control signal with a contribution proportional to the current control error. The integral part is the memory of the PID controller. It is proportional to a weighted sum of all past control errors. Lastly, the derivative part tries to predict future control errors using the derivative of the current control error.

Lecture 2

Process Models

In the next few lectures we will discuss different ways to describe the dynamics of the process. There are two predominant ways to obtain a dynamical model of a process. One is to calculate the model through mass balances, energy balances, force and moment equilibria etc. The other way is to conduct experiments on the process. Most often, a combination of these two methods is used.

2.1 State Space Model

By writing down a suitable balance equation one can describe the process by means of a differential equation.

$$\frac{d^n y}{dt^n} + a_1 \frac{d^{n-1} y}{dt^{n-1}} + \ldots + a_n y = b_0 \frac{d^n u}{dt^n} + b_1 \frac{d^{n-1} u}{dt^{n-1}} + \ldots + b_n u \tag{2.1}$$

The above equation is linear. Most often, however, balance equations lead to non-linear differential equation. We treat this case in the next section.

The differential equation (2.1) can be written in state space form. If we introduce n states $x_1, x_2, \cdots, x_n$, Equation (2.1) can be written as a system of first order differential equations.

$$\begin{aligned}
\dot{x}_1 &= f_1(x_1, x_2, \ldots, x_n, u) \\
\dot{x}_2 &= f_2(x_1, x_2, \ldots, x_n, u) \\
&\vdots \\
\dot{x}_n &= f_n(x_1, x_2, \ldots, x_n, u) \\
\\
y &= g(x_1, x_2, \ldots, x_n, u)
\end{aligned}$$

Note that f_i and g are functions of the states and the control signal. They must not depend on the measurement signal y.

Introduce the vectors

$$x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix} \qquad f = \begin{pmatrix} f_1 \\ f_2 \\ \vdots \\ f_n \end{pmatrix}$$

This allows us to write the equation system in the compact form

$$\begin{aligned}
\dot{x} &= f(x, u) \\
y &= g(x, u)
\end{aligned}$$

EXAMPLE 2.1—FROM DIFFERENTIAL EQUATION TO STATE SPACE FORM
Assume that we have obtained the following differential equation, describing a process:

$$\ddot{y} + a_1\dot{y} + a_2 y = bu$$

Since this is a second-order differential equation, we have $n = 2$ and consequently we need two state variables to write the equation in state space form. These can be chosen in several ways. One possible choice would be

$$\begin{cases} x_1 = y \\ x_2 = \dot{y} \end{cases}$$

It yields the following state space description

$$\dot{x} = \begin{pmatrix} 0 & 1 \\ -a_2 & -a_1 \end{pmatrix} x + \begin{pmatrix} 0 \\ b \end{pmatrix} u$$

$$y = \begin{pmatrix} 1 & 0 \end{pmatrix} x$$

□

Generally, the state space description has the form

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

where A, B, C and D are matrices. It can only be written this way if $f(x, u)$ and $g(x, u)$ are linear.

The functions f and g being linear means that the process has the same dynamic properties independent of operation point. If the process is linear the super-position principle holds. I.e. if the control signals u_1 and u_2 result in the measurement signals y_1 and y_2, respectively, the super-position principle implies that the control signal $u_1 + u_2$ results in the measurement signal $y_1 + y_2$.

Figure 2.1 shows an example of a process which is *not* linear. The process consists of a tank from which fluid is pumped out. The purpose of the control is to govern the valve which affects the inflow to the tank, such that the fluid level is held constant. When the level is low, the cross section of the tank is small. This means that changes in the inflow result in relatively fast and substantial changes in the fluid level. The processes is therefore fast with a high gain for low fluid levels. When the level is high, the process becomes slow with a lower gain.

What do we do if f and g are nonlinear? One could of course describe the process using nonlinear differential equations, but these are significantly more complex to

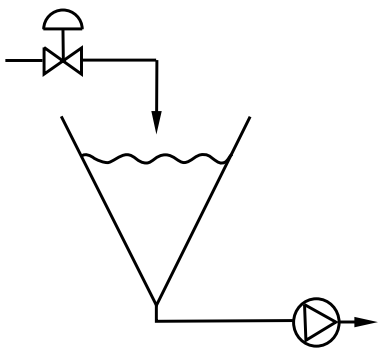

Figure 2.1 A nonlinear process.

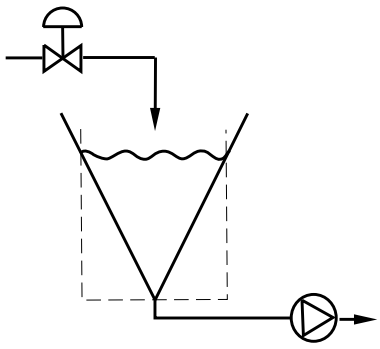

Figure 2.2 Linearization of a nonlinear process.

handle. Consequently, a common approach is to approximate the nonlinear equations by linear ones. The method is illustrated in Figure 2.2. If we know that we want to control the tank process around a certain level, we can assume that the tank has vertical walls, according to the figure. Thus we obtain a linear process. When we lie close to the desired level, the linear and nonlinear equations will hopefully exhibit similar properties.

2.2 Linearization

We will now demonstrate how one linearizes a system

$$\dot{x} = f(x, u)$$
$$y = g(x, u)$$

where f and g are nonlinear functions of x and u. The linearization consists of the following four steps:

1. Determine a stationary point (x_0, u_0) around which we shall approximate the system. For a stationary point it holds that

$$\dot{x}_0 = 0 \quad \Leftrightarrow \quad f(x_0, u_0) = 0$$

2. Make Taylor series expansions of f and g around (x_0, u_0). Keep only the first order terms.

$$f(x, u) \approx f(x_0, u_0) + \frac{\partial}{\partial x} f(x_0, u_0)(x - x_0) + \frac{\partial}{\partial u} f(x_0, u_0)(u - u_0)$$
$$g(x, u) \approx g(x_0, u_0) + \frac{\partial}{\partial x} g(x_0, u_0)(x - x_0) + \frac{\partial}{\partial u} g(x_0, u_0)(u - u_0)$$

   Note that $f(x_0, u_0) = 0$ and introduce the notion $y_0 = g(x_0, u_0)$.

3. Introduce the new variables

$$\Delta x = x - x_0$$
$$\Delta u = u - u_0$$
$$\Delta y = y - y_0$$

4. The state space equations in the new variables are given by

$$\dot{\Delta x} = \dot{x} - \dot{x}_0 = \dot{x} = f(x, u) \approx \frac{\partial}{\partial x} f(x_0, u_0)\Delta x + \frac{\partial}{\partial u} f(x_0, u_0)\Delta u = A\Delta x + B\Delta u$$
$$\Delta y = y - y_0 = g(x, u) - y_0 \approx \frac{\partial}{\partial x} g(x_0, u_0)\Delta x + \frac{\partial}{\partial u} g(x_0, u_0)\Delta u = C\Delta x + D\Delta u$$

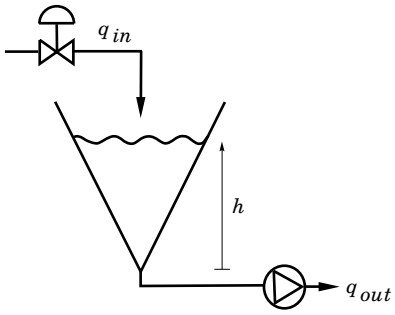

Figure 2.3 The nonlinear tank in Example 2.3.

Note that x and f are vectors. If we for instance deal with a second order system with the two states x_1 and x_2, a measurement signal y and a control signal u it holds that

$$x = \begin{pmatrix} x_1 \\ x_2 \end{pmatrix},\ f = \begin{pmatrix} f_1 \\ f_2 \end{pmatrix},\ \frac{\partial f}{\partial x} = \begin{pmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \end{pmatrix},\ \frac{\partial f}{\partial u} = \begin{pmatrix} \frac{\partial f_1}{\partial u} \\ \frac{\partial f_2}{\partial u} \end{pmatrix},\ \frac{\partial g}{\partial x} = \begin{pmatrix} \frac{\partial g}{\partial x_1} & \frac{\partial g}{\partial x_2} \end{pmatrix}$$

Example 2.2—Linearization of a nonlinear tank
Figure 2.3 shows the previously studied tank process, with some added quantity labels. Assuming that the tank walls form an angle that makes the diameter of the fluid surface equal to the height of the fluid, the fluid volume becomes

$$V = \frac{\pi h^3}{12}$$

A mass balance for the tank shows that the change in volume equals the inflow minus the outflow

$$\frac{dV}{dt} = q_{in} - q_{out}$$

We are interested in changes of the fluid level. These are obtained through

$$\frac{dV}{dt} = \frac{dV}{dh}\frac{dh}{dt} = \frac{\pi h^2}{4}\frac{dh}{dt}$$

i.e.

$$\frac{dh}{dt} = \frac{4}{\pi h^2}(q_{in} - q_{out})$$

Introduce the familiar notions

$$x = y = h, \quad u = q_{in}$$

and assume that q_{out} is constant. The resulting state space description is given by

$$\dot{x} = \frac{4}{\pi x^2}(u - q_{out}) = f(x, u)$$
$$y = x = g(x, u)$$

Now we linearize this system according to the method given above.

1. Stationary point.
$$f(x_0, u_0) = 0 \Leftrightarrow u_0 = q_{out}$$
This criterion simply means that the inflow must equal the outflow in stationarity. We obtain no restrictions on x_0, i.e. we can linearize around any given level.

2. Taylor series expansion.

$$f(x,u) \approx -\frac{8}{\pi x_0^3}(u_0 - q_{out})(x - x_0) + \frac{4}{\pi x_0^2}(u - u_0) = \frac{4}{\pi x_0^2}(u - u_0)$$
$$g(x,u) \approx y_0 + 1 \cdot (x - x_0) + 0 \cdot (u - u_0) = y_0 + (x - x_0)$$

3. New variables.

$$\Delta x = x - x_0$$
$$\Delta u = u - u_0$$
$$\Delta y = y - y_0$$

4. State space equations in the new variables.

$$\dot{\Delta x} = f(x,u) \approx \frac{4}{\pi x_0^2}\Delta u$$
$$\Delta y = g(x,u) - y_0 = \Delta x$$

We observe that the linearization in this case meant replacing the division by h^2 in the nominal case by a division by x_0^2, where x_0 is the level around which the system was linearized. □

2.3 The Transfer Function

The Laplace transformation of a time function $f(t)$ is defined as

$$\mathcal{L}(f(t))(s) = F(s) = \int_{0-}^{\infty} e^{-st} f(t)dt$$

where s is a complex variable. If $f(0) = 0$ the Laplace transform holds the pleasant property

$$\mathcal{L}\left(\frac{df(t)}{dt}\right)(s) = sF(s)$$

I.e. the derivative of the time function corresponds to a multiplication of the corresponding Laplace transformation by s. By induction it holds for higher order derivatives that

$$\mathcal{L}\left(\frac{d^n f(t)}{dt^n}\right)(s) = s^n F(s)$$

assuming that $f(0) = f'(0) = \ldots = f^{n-1}(0) = 0$. From now on we will assume that all these function values are equal to zero.

Example 2.3—Laplace transformation of a differential equation
Assume that the following differential equation describes a process.

$$\ddot{y} + a_1\dot{y} + a_2 y = b_1\dot{u} + b_2 u$$

Laplace transformation yields

$$(s^2 + a_1 s + a_2)Y(s) = (b_1 s + b_2)U(s)$$

This can also be written

$$Y(s) = \frac{b_1 s + b_2}{s^2 + a_1 s + a_2}U(s) = G(s)U(s)$$

□

The function $G(s)$ is called the transfer function of the system. Often, as in the above example, it consists of a polynomial fraction:

$$G(s) = \frac{Q(s)}{P(s)}$$

The zeros of the polynomial $Q(s)$ are called the zeros of the system. The zeros of the polynomial $P(s)$ are called the poles of the system. Soon we will see that the values of the poles are important for the behaviour of the system.

Example 2.4—The transfer function of the PID controller
In this example we will derive the transfer function of the PID controller. From Lecture 1 we get the equation of the PID controller:

$$u = K\left(e + \frac{1}{T_i}\int e(t)dt + T_d\frac{de}{dt}\right)$$

Laplace transformation gives

$$U(s) = K\left(E(s) + \frac{1}{sT_i}E(s) + T_d sE(s)\right) = K\left(1 + \frac{1}{sT_i} + T_d s\right)E(s)$$

Here, we have used the fact that integration corresponds to division with s, in the same way as derivation corresponds to multiplication with s.

So, the transfer function of the PID controller becomes

$$G_R(s) = K\left(1 + \frac{1}{sT_i} + T_d s\right)$$

□

The Relation between the State Space Model and the Transfer Function

Often one wants to alternate between working with the state space description and the transfer function. Therefore it is essential to be able to transit between these descriptions. We end this lecture by demonstrating how this can be achieved.

Translation from State Space Form → G(s) Start with the state space description

$$\begin{aligned}\dot{x} &= Ax + Bu\\ y &= Cx + Du\end{aligned}$$

Laplace transformation yields

$$\begin{aligned}sX(s) &= AX(s) + BU(s)\\ Y(s) &= CX(s) + DU(s)\end{aligned}$$

Solve for $X(s)$ in the first equation and subsequently eliminate $X(s)$ from the second one

$$\begin{aligned}X(s) &= (sI - A)^{-1}BU(s)\\ Y(s) &= C(sI - A)^{-1}BU(s) + DU(s)\end{aligned}$$

where I is the identity matrix. The transfer function thus becomes

$$G(s) = C(sI - A)^{-1}B + D$$

The denominator polynomial of the transfer function is given by $P(s) = \det(sI - A)$, which is the characteristic polynomial of the matrix A. Consequently, the poles of the system are identical to the eigenvalues of the matrix A.

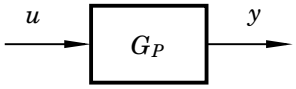

Figure 2.4 Block diagram for a process

Translation from G(s) → State Space Form There exists no unique solution to this problem. A degree n transfer function requires at least n states, but they can be chosen in many different ways. Further, it is possible to choose more than n states.

Sometimes the choice of states is naturally given by certain physical entities such as position, speed and acceleration. In other cases we have no physical references and are only interested in obtaining an arbitrary n^{th} order state space description. For these cases three forms are given in the collection of formulae: controllable form, observable form and diagonal form.

2.4 Block diagram representation

In Figure 1.1, a block diagram describing the simple feedback loop was presented. The block diagram is a convenient and efficient way to describe relations between signals in control systems. If the notations *Controller* and *Process* in Figure 1.1 are replaced by the transfer functions that describe the relations between the signals, it is possible to calculate relations between different signals in the diagram.

In Figure 2.4, a very simple block diagram is presented. It describes a process with input signal u, output signal y, and transfer function $G_P(s)$. The output signal from the block is given by the transfer function in the block multiplied by the input signal, i.e.

$$Y(s) = G_P(s)U(s)$$

Most block diagrams consist of three components; blocks with transfer functions, signals represented by arrows, and summations. In Figure 2.5, a block diagram describing the simple feedback loop is presented, with process transfer function G_P, controller transfer function G_R and the signals setpoint r, control signal u, process output y and control error e. The block diagram gives the following relations between the signals.

$$\begin{aligned} Y &= G_p U \\ U &= G_R E \\ E &= R - Y \end{aligned} \tag{2.2}$$

If one, e.g., wants to determine the transfer function between setpoint r and process output y, the other signals can be eliminated in the following way.

$$\begin{aligned} &Y = G_p U \\ &U = G_R E \\ &E = R - Y \end{aligned} \quad \longrightarrow \quad \begin{aligned} &Y = G_p U \\ &U = G_R(R - Y) \end{aligned} \quad \longrightarrow \quad Y = G_p G_R(R - Y)$$

Finally, the transfer function between r and y is

$$\begin{aligned} Y &= G_p G_R(R - Y) \\ (1 + G_p G_R)Y &= G_p G_R R \\ Y &= \frac{G_p G_R}{1 + G_p G_R} R \end{aligned} \tag{2.3}$$

After getting used to block diagram calculation, it is normally not necessary to write down the intermediate results like those in (2.2), but one write down the equations containing only the signals of interest directly, like the upper equation in

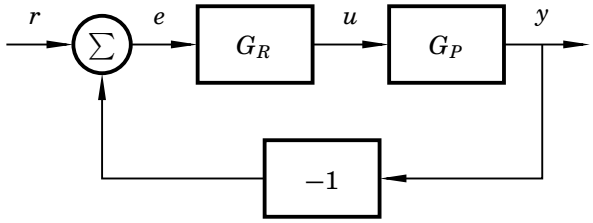

Figure 2.5 Block diagram for the simple feedback loop

(2.3). This is done by starting with the output signal and going through the diagram, against the arrows.

For more complicated systems, it is, however, often necessary to introduce intermediate variables and divide the calculations into subcalculations. It is then often useful to introduce intermediate variables after the summations, like e in Figure 2.5.

Lecture 3

Impulse- and Step Response Analysis

We begin this lecture by describing two additional process models, i.e. the impulse and step response. This is followed by an examination of the relation between the transfer function and the properties of the step response.

The state space description of a system can be written in the form

$$\dot{x} = Ax + Bu$$
$$y = Cx + Du$$

The solution to this system of differential equations is given by

$$y(t) = Ce^{At}x(0) + C\int_0^t e^{A(t-\tau)}Bu(\tau)d\tau + Du(t) \tag{3.1}$$

The output y can thus be described using three terms. The first term takes into account the initial state. This is commonly uninteresting from a control perspective except when the controller is initiated. The third term is called the direct term. It is often negligible in practical systems. The remaining second term consists of a weighted integral of the control signal. We shall utilize the solution (3.1) in order to study the impulse and step response. In the next lecture we will use the solution to analyze the frequency response.

3.1 The Impulse Response

Figure 3.1 shows an impulse response, i.e. the output y from a process when the input u is given by a short pulse.

Assume that the control signal is described as an ideal pulse, i.e. a Dirac function

$$u(t) = \delta(t)$$

If this control signal is introduced in Equation (3.1) and we assume that the initial state is given by $x(0) = 0$ we obtain

$$y(t) = C\int_0^t e^{A(t-\tau)}B\delta(\tau)d\tau + D\delta(t) = Ce^{At}B + D\delta(t) \equiv h(t) \tag{3.2}$$

The impulse response is also called the weighting function and is denoted $h(t)$. The reason is the following: By comparing the expression for the weighting function (3.2) to the general equation of the output, we see that Equation (3.1) can be rewritten as

$$y(t) = \int_0^t h(t-\tau)u(\tau)d\tau \tag{3.3}$$

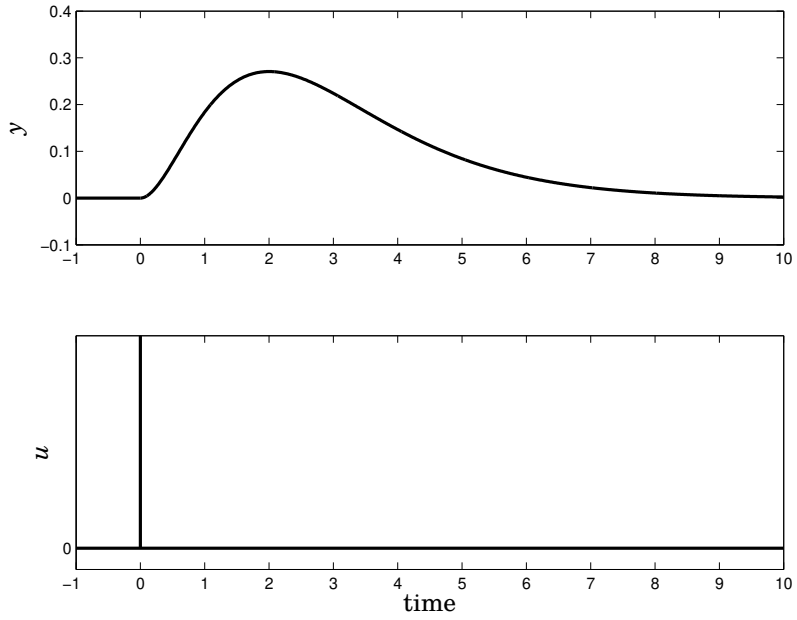

Figure 3.1 Impulse response experiment.

In other words, the weighting function tells which weights shall be assigned to old inputs when calculating the output.

The Laplace transformation of an impulse is obtained from the definition of the Laplace transform

$$U(s) = \int_{0-}^{\infty} e^{-st}\delta(t)dt = 1$$

This means that the Laplace transformations of the impulse response is given by the transfer function

$$G(s) = \frac{Y(s)}{U(s)} = Y(s)$$

Impulse response analysis is not very common in technological applications. However, it is common in medicine and biology. One can for instance inject a substance into the blood circulation and study the resulting uptake and secretion.

3.2 The Step Response

Figure 3.2 shows a step response, i.e. an output y from a process when the input u is given by a step. This is the by far most common experiment used to obtain a process model in technological applications.

Assume that the control signal is given by a step

$$u(t) = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases}$$

If we introduce this control signal in Equation (3.3) we obtain

$$y(t) = \int_0^t h(t-\tau')d\tau' = [\tau = t - \tau'] = \int_0^t h(\tau)d\tau \tag{3.4}$$

The step response is thus the integral of the impulse response. The Laplace transformation of a step is given by

$$U(s) = \int_{0-}^{\infty} e^{-st}dt = -\frac{1}{s}\left[e^{-st}\right]_0^{\infty} = \frac{1}{s}$$

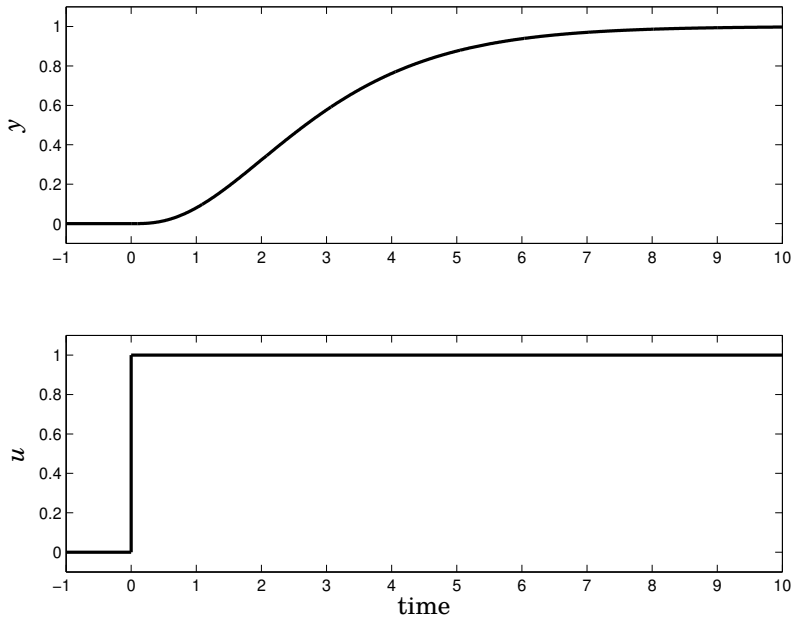

Figure 3.2 Step response experiment.

3.3 The Relation between Poles and the Step Response

In this section we will study the relationship between the transfer function and the step response. When the input $u(t)$ is a step, the Laplace transformation of the output $y(t)$ becomes

$$Y(s) = G(s)U(s) = G(s)\frac{1}{s}$$

One can obviously calculate the inverse transform of this expression in order to obtain the step response. In many situations it is, however, useful to be able to see the characteristics of the step response directly from the transfer function, without the need of an inverse transformation.

Two theorems which are useful when determining the initial and final properties of the step response are the

Initial Value Theorem:	$\lim_{t\to 0} f(t) = \lim_{s\to\infty} sF(s)$
Final Value Theorem:	$\lim_{t\to\infty} f(t) = \lim_{s\to 0} sF(s)$

It is essential to denote that these theorems must be used with precaution. Before utilizing them one must ensure that the limits really exist. This will be further explained in Lecture 7.

The final value theorem can be used in the following way to calculate the final value of the measurement signal y when the input u is given by a unit step.

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = \lim_{s\to 0} sG(s)\frac{1}{s} = G(0)$$

If the limit exists, the stationary value of the output is determined by the static gain $G(0)$ of the transfer function.

The characteristics of the step response are largely determined by the values of the poles of the transfer function. In this lecture we will see how the poles affect the properties of the step response for some simple processes. The influence of zeros are treated later in the course, in Lecture 12.

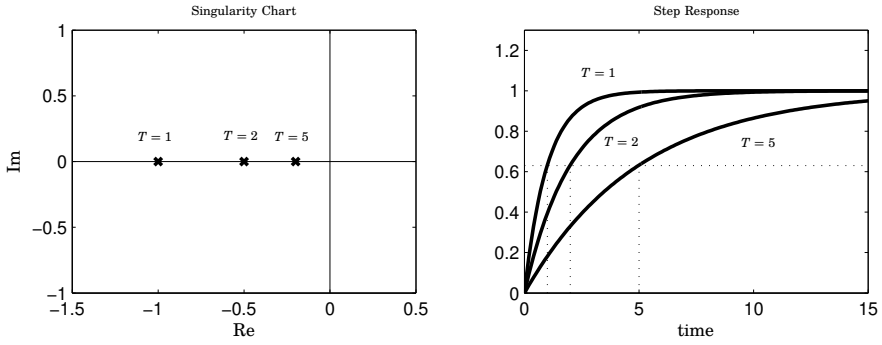

Figure 3.3 The placement of the poles and the step responses for the process $G(s) = K/(1+sT)$ as $K = 1$ and $T = 1, 2, 5$.

Example 3.1—First-order system
Assume that the transfer function of the process is given by

$$G(s) = \frac{K}{1+sT}$$

This process has a pole in $s = -1/T$. If the input of the process is a unit step, the Laplace transformation of the process output becomes

$$Y(s) = G(s)\frac{1}{s} = \frac{K}{s(1+sT)}$$

The output of the process is obtained by inverse transformation

$$y(t) = K\left(1 - e^{-t/T}\right)$$

If $T < 0$, i.e. if the pole lies in the right-half plane, the process is unstable and $y(t)$ grows out of bounds. From now on we assume that $T > 0$, i.e. that the pole lies in the left-half plane.

Figure 3.3 shows the step responses corresponding to three different values of T. The figure shows that a smaller T yields a faster step response. It also shows the pole of the process. The further into the left-half plane the pole lies, the faster the step response becomes.

In this case, the final value theorem gives

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = \lim_{s\to 0} \frac{sK}{s(1+sT)} = K$$

It says that the process output approaches $y = K$ as $t \to \infty$.

Parameter T is called the time constant of the process. At time $t = T$ the process output is given by

$$y(T) = K\left(1 - e^{-T/T}\right) = K\left(1 - e^{-1}\right) \approx 0.63K$$

Time constant T is thus the time it takes for the step response to reach up to 63% of its final value. This is illustrated by Figure 3.3.

Applying the initial value theorem on $y(t)$ tells us that $y(0) = 0$. It is, however, more interesting to study at which rate the process output changes in the initial phase, i.e. to study $\dot{y}(0)$.

$$\lim_{t\to 0} \dot{y}(t) = \lim_{s\to\infty} s \cdot sY(s) = \lim_{s\to\infty} \frac{s^2K}{s(1+sT)} = \frac{K}{T}$$

The shorter the time constant, i.e. the further into the left half plane the pole lies, the faster the initial change of the process output becomes. This is also illustrated by Figure 3.3. □

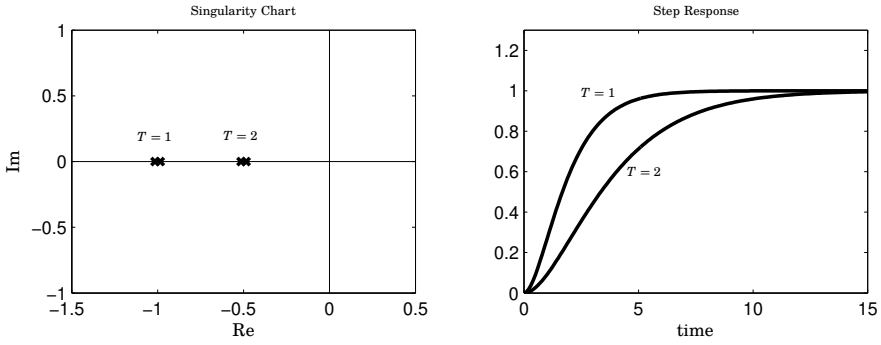

Figure 3.4 The pole placement and step response of the process $G(s) = K/(1+sT)^2$ as $K = 1$ and $T = 1, 2$.

Example 3.2—Second-order system, real poles
Assume that the transfer function of a process is given by

$$G(s) = \frac{K}{(1+sT_1)(1+sT_2)}$$

This process has two real poles in $s = -1/T_1$ and $s = -1/T_2$, respectively. If the input to the process is a unit step, the Laplace transformation of the process output is given by

$$Y(s) = G(s)\frac{1}{s} = \frac{K}{s(1+sT_1)(1+sT_2)}$$

By inverse transformation of this expression, we obtain the output of the process

$$y(t) = \begin{cases} K\left(1 - \dfrac{T_1 e^{-t/T_1} - T_2 e^{-t/T_2}}{T_1 - T_2}\right) & T_1 \neq T_2 \\ K\left(1 - e^{-t/T} - \dfrac{t}{T}e^{-t/T}\right) & T_1 = T_2 = T \end{cases}$$

From the first expression we see that when one of the time constants is significantly smaller than the other, the step response will approach that of a first-order process, as in Example 3.1. If, e.g., $T_1 \gg T_2$, the response will be like in Example 3.1 with $T \approx T_1$. If any pole lies in the right-half plane, the process is unstable and $y(t)$ grows out of bounds. We therefore assume that both poles lie in the left-half plane, i.e. that $T_1 > 0$ and $T_2 > 0$.

Figure 3.4 shows the step response of the process as $T_1 = T_2 = T$. Just as in the previous example, we observe that the step response becomes faster as the poles move further into the left-half plane. The final value theorem gives

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = \lim_{s\to 0} \frac{sK}{s(1+sT_1)(1+sT_2)} = K$$

It tells us that the output approaches $y = K$ as $t \to \infty$.

The initial value theorem applied to $\dot{y}(t)$ yields

$$\lim_{t\to 0} \dot{y}(t) = \lim_{s\to\infty} s \cdot sY(s) = \lim_{s\to\infty} \frac{s^2 K}{s(1+sT_1)(1+sT_2)} = 0$$

Hence the time derivative is zero, which is also evident from Figure 3.4.

It is easy to verify that the time derivative is zero for all systems where the number of poles minus the number of zeros is greater than one. □

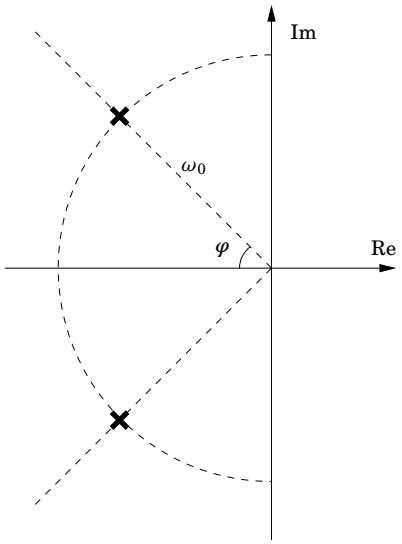

Figure 3.5 Interpretation of the parameters in the polynomial $s^2 + 2\zeta\omega_0 s + \omega_0^2$. The frequency ω_0 denotes the distance between the poles and the origin, whereas $\zeta = \cos\varphi$ is the relative damping of the system.

EXAMPLE 3.3—SECOND-ORDER SYSTEM, COMPLEX POLES

For a second order system with complex poles it is often convenient to write the transfer function in the form

$$G(s) = \frac{K\omega_0^2}{s^2 + 2\zeta\omega_0 s + \omega_0^2} \qquad 0 < \zeta < 1$$

Interpretations of the parameters ω_0 and ζ are given in Figure 3.5.

The parameter ω_0 denotes the distance between the poles and the origin. The parameter ζ is called the relative damping and is related to the angle φ in Figure 3.5 through

$$\zeta = \cos\varphi$$

The relative damping ζ gives the relation between the real and imaginary parts of the poles.

If the process input is a step, the Laplace transformation of the process output becomes

$$Y(s) = G(s)\frac{1}{s} = \frac{K\omega_0^2}{s(s^2 + 2\zeta\omega_0 s + \omega_0^2)}$$

By inverse transformation of this expression the following process output is obtained.

$$y(t) = K\left(1 - \frac{1}{\sqrt{1-\zeta^2}}e^{-\zeta\omega_0 t}\sin\left(\omega_0\sqrt{1-\zeta^2}t + \arccos\zeta\right)\right) \qquad 0 < \zeta < 1$$

The expression contains a term consisting of a sinusoid with decaying amplitude.

Figure 3.6 shows the step response of the process for some different values of ζ and ω_0. The parameter ω_0 determines the distance between the poles and the origin. Just as in the previous examples, the step response becomes faster as the poles are moved further into the left-half plane. We also see that the shape of the step response does not change as long as ζ is held constant.

The parameter ζ determines the ratio between the real and imaginary part of the poles. Figure 3.6 shows that the smaller ζ is, the less damped the step response becomes. We also see that the initial part of the step response is fairly consistent as long as ω_0 is held constant.

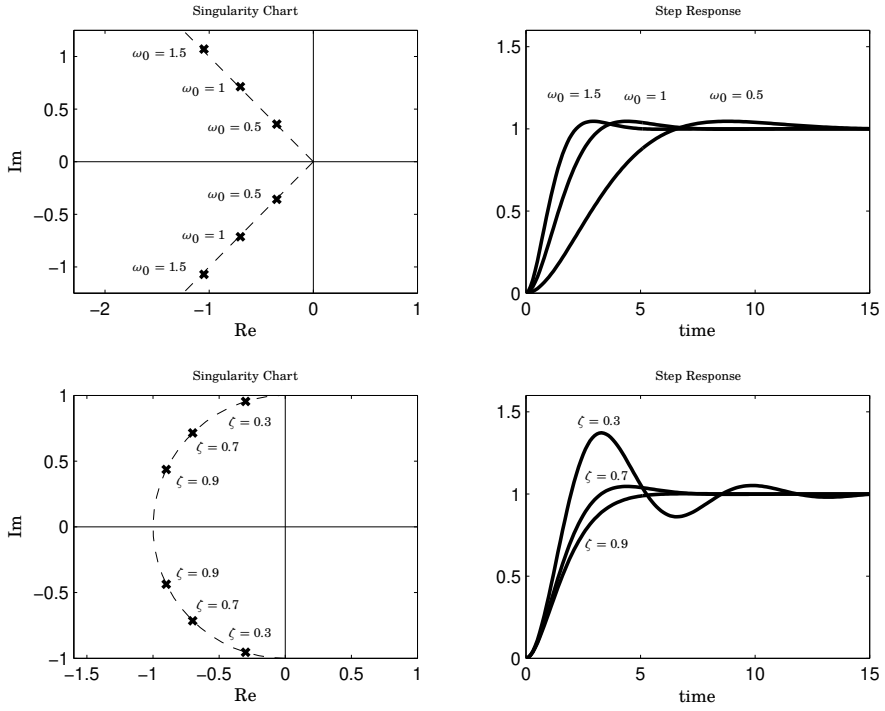

Figure 3.6 The placement of the poles and the step response of the process $G(s) = K/(s^2 + 2\zeta\omega_0 s + \omega_0^2)$ for $K = 1$. The two upper plots show the cases $\zeta = 0.7$ and $\omega_0 = 0.5, 1, 1.5$. The two lower plots show the cases $\omega_0 = 1$ and $\zeta = 0.3, 0.7, 0.9$.

In this example the final value theorem gives

$$\lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = \lim_{s\to 0} \frac{sK\omega_0^2}{s(s^2 + 2\zeta\omega_0 s + \omega_0^2)} = K$$

The theorem thus yields that the process output approaches $y = K$ as $t \to \infty$. This is also evident from Figure 3.6.

The initial value theorem applied to $\dot{y}(t)$ yields

$$\lim_{t\to 0} \dot{y}(t) = \lim_{s\to\infty} s \cdot sY(s) = \lim_{s\to\infty} \frac{s^2 K\omega_0^2}{s(s^2 + 2\zeta\omega_0 s + \omega_0^2)} = 0$$

Analogous to the case of real poles in the previous example, the time derivative is zero in this second-order case. □

Lecture 4

Frequency Analysis

In this lecture two additional ways of describing process dynamics are described. They are based on the frequency response and are normally illustrated graphically. After a thorough introduction of the Bode plot and the Nyquist curve, the lecture as well as the modelling part of the course are concluded by some examples showing the relation between the different process models descriptions, which have been introduced.

4.1 The Frequency Response

Figure 4.1 shows a frequency response, i.e. the output y from a process with a sinusoidal input u.

From the figure we see that the output becomes a sinusoid with the same frequency as the input, after the extinction of an initial transient. The only thing distinguishing the two signals after the initial transient is that they have different amplitude and that they are phase shifted. We shall now show that this is always the case.

Assume that the control signal is given by a sinusoid of frequency ω, i.e.

$$u(t) = \sin \omega t$$

Let $G(s)$ be the transfer function and $h(t)$ the impulse response of the studied process.

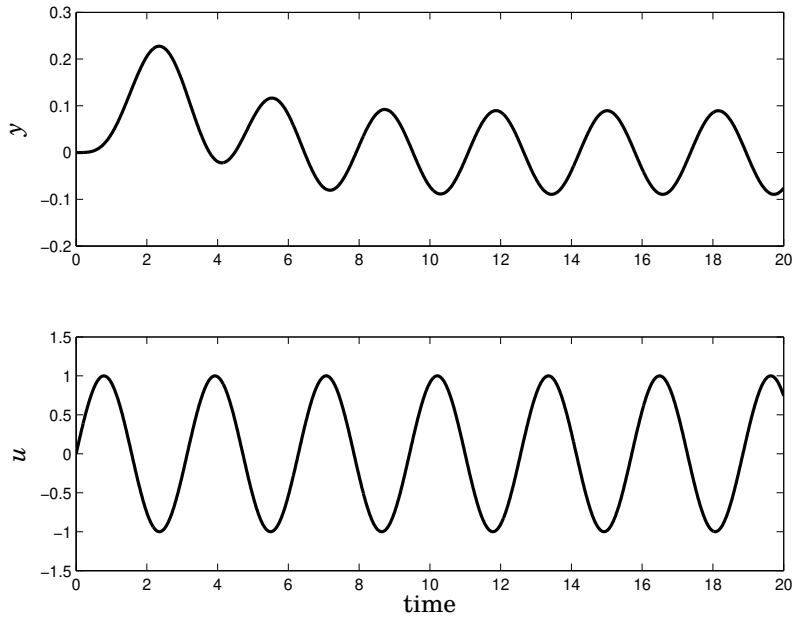

Figure 4.1 Frequency response experiment.

Since the Laplace transformation of an impulse is simply 1, it holds that

$$G(s) = \mathcal{L}(h(t))(s) = \int_{0-}^{\infty} e^{-st} h(t) dt$$

We exploit this in order to find which output we obtain when the input is given by a sinusoid. When the transient has died out, i.e. when $t \to \infty$, Equation (3.3) yields

$$\begin{aligned} y(t) &= \int_0^t h(t-\tau')u(\tau')d\tau' = [\tau = t - \tau'] = \int_0^t h(\tau)u(t-\tau)d\tau \\ &= \int_0^t h(\tau)\sin\omega(t-\tau)d\tau = \operatorname{Im}\int_0^t h(\tau)e^{i\omega(t-\tau)}d\tau \\ &= \operatorname{Im}\int_0^t h(\tau)e^{-i\omega\tau}d\tau e^{i\omega t} = [t \to \infty] = \operatorname{Im} G(i\omega)e^{i\omega t} \\ &= |G(i\omega)|\operatorname{Im} e^{i\arg G(i\omega)}e^{i\omega t} = |G(i\omega)|\sin(\omega t + \arg G(i\omega)) \end{aligned}$$

This means that when the control signal is given by $u(t) = \sin(\omega t)$, the measurement signal becomes $y(t) = a\sin(\omega t + \varphi)$, where

$$a = |G(i\omega)| \tag{4.1}$$

$$\varphi = \arg G(i\omega) \tag{4.2}$$

If we carry out a frequency analysis, i.e. let the control signal be a sinusoid with varying frequency, and measure the amplitude and phase shift of the measurement signal, we can thus determine the value of the transfer function for these frequencies. We obtain a table containing frequencies and their corresponding amplitudes and phase shifts. A table is, however, an inconvenient representation of the process dynamics. Therefore the table is usually represented graphically. This is done mainly in either of two ways; the Nyquist curve and the Bode plot.

4.2 The Nyquist Curve

The Nyquist curve is constituted of the complex number $G(i\omega)$ drawn in the complex plane for ω in $[0, \infty]$. Figure 4.2 shows a typical Nyquist curve.

Most processes have low-pass characteristics. This means that the measurement signal of the process is affected by low frequency inputs, whereas high frequency signals are damped out. Since the distance between the origin and points on the Nyquist curve describes the gain of process, it is normal that the Nyquist curve approaches the origin for high frequencies, as shown in Figure 4.2. The phase shift between in- and output does usually increase with the frequency. This is the explanation to the spiral shape of the curve in Figure 4.2.

The following example shows how one can compute the shape of the Nyquist curve, given the transfer function.

Example 4.1—Nyquist curve drawing
Assume that the process is described by the transfer function

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

We compute $G(i\omega)$ and separate into its real- and imaginary parts

$$G(i\omega) = \frac{1}{1+i\omega} = \frac{1-i\omega}{1+\omega^2} = \frac{1}{1+\omega^2} - i\frac{\omega}{1+\omega^2}$$

We see that the real part is positive, whereas the imaginary part is negative for all ω. In other words, the Nyquist curve will be contained in the fourth quadrant. Further, we see that $G(i\omega) \approx 1$ for small ω and $G(i\omega) \to 0$ as $\omega \to \infty$. The Nyquist curve is shown in Figure 4.3 □

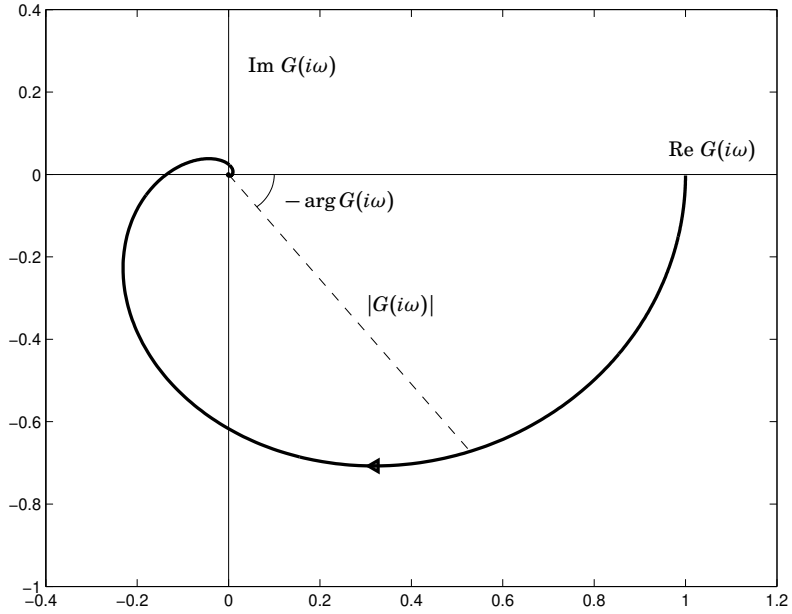

Figure 4.2 Nyquist curve.

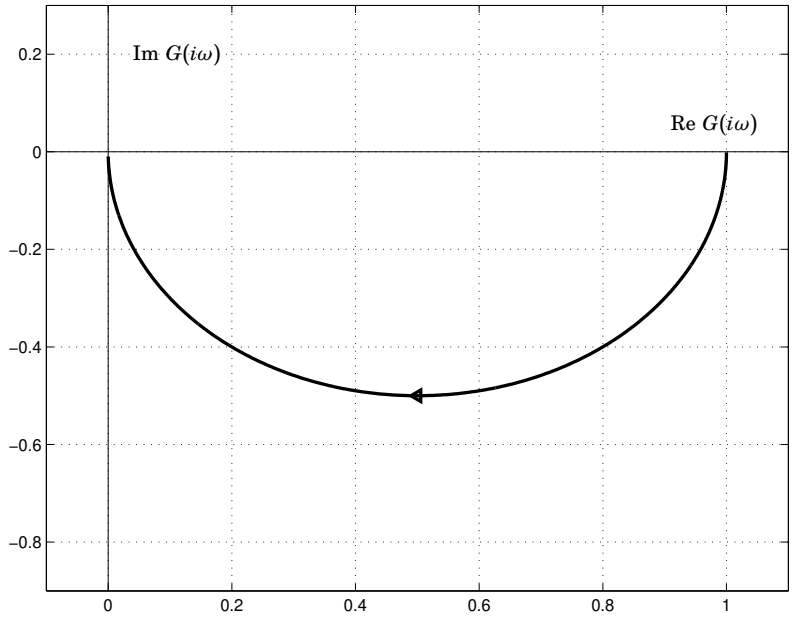

Figure 4.3 The Nyquist curve in Example 4.1.

4.3 The Bode Plot

The Bode plot features two curves, $|G(i\omega)|$ and $\arg G(i\omega)$ as functions of ω. Figure 4.4 shows the Bode plot of a typical process. The magnitude plot is drawn in a logarithmic scale, whereas the argument is drawn in a linear scale. The frequency axis is logarithmic.

The Bode plot of a process often looks as the one show in Figure 4.4. The low frequency gain is often constant and corresponds to the static gain of the process. As the frequency increases, the gain decreases and phase shift increases. In other words, the process has low pass characteristics as we have seen previously. There is, however, an exception as we will soon see.

The Bode plot has some properties which makes it easier to draw than the

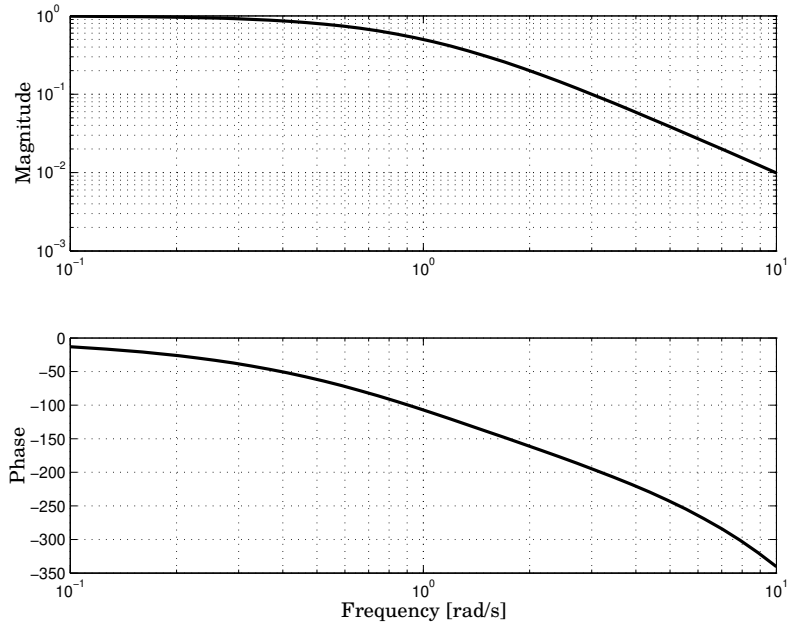

Figure 4.4 Bode plot.

Nyquist plot. Assume that we can factor the transfer function, e.g. as

$$G(s) = G_1(s)G_2(s)G_3(s)$$

The logarithms of the magnitude and argument, respectively, are given by

$$\log|G(i\omega)| = \log|G_1(i\omega)| + \log|G_2(i\omega)| + \log|G_3(i\omega)|$$
$$\arg G(i\omega) = \arg G_1(i\omega) + \arg G_2(i\omega) + \arg G_3(i\omega)$$

This means that the Bode plot of a transfer function is given by the sum of the Bode plots of its factors. This, in terms, enables us to draw all Bode plots which correspond to products of less complex transfer functions, for which we have already drawn the Bode plots. We shall study five sample transfer functions, into which all transfer functions in this course can be factored. These sample transfer functions are

1. K
2. s^n
3. $(1+sT)^n$
4. $\left(1 + 2\zeta s/\omega_0 + (s/\omega_0)^2\right)^n$
5. e^{-sL}

where K, T, ζ, ω_0 and L are real numbers and n is an integer.

1. Bode Plot of $G(s) = K$

The magnitude and argument of the transfer function $G(s) = K$ are given by

$$\log|G(i\omega)| = \log K$$
$$\arg G(i\omega) = 0$$

Both the gain and argument are independent of ω. The Bode plot is thus made up by two horizontal lines. This is shown in Figure 4.5 where Bode plots corresponding to three values of K are shown.

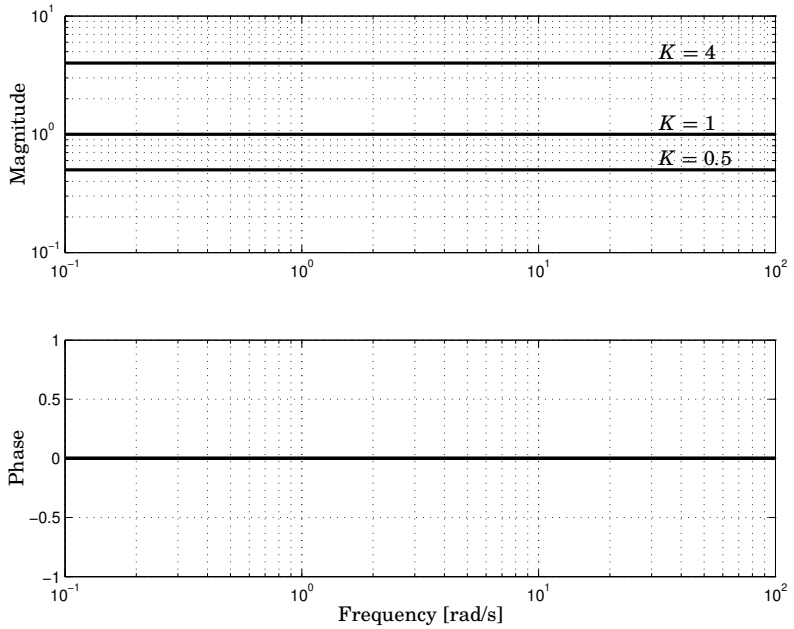

Figure 4.5 Bode plot of $G(s) = K$, where $K = 0.5, 1$ and 4.

2. Bode Plot of $G(s) = s^n$

The magnitude and argument of the transfer function $G(s) = s^n$ are given by

$$\log|G(i\omega)| = \log|i\omega|^n = n\log\omega$$
$$\arg G(i\omega) = n\arg(i\omega) = n\frac{\pi}{2}$$

The magnitude plot is a straight line with slope n, due to the logarithmic scales. The argument is independent of ω and thus forms a horizontal line. Figure 4.6 shows three Bode plots corresponding to some different values of n.

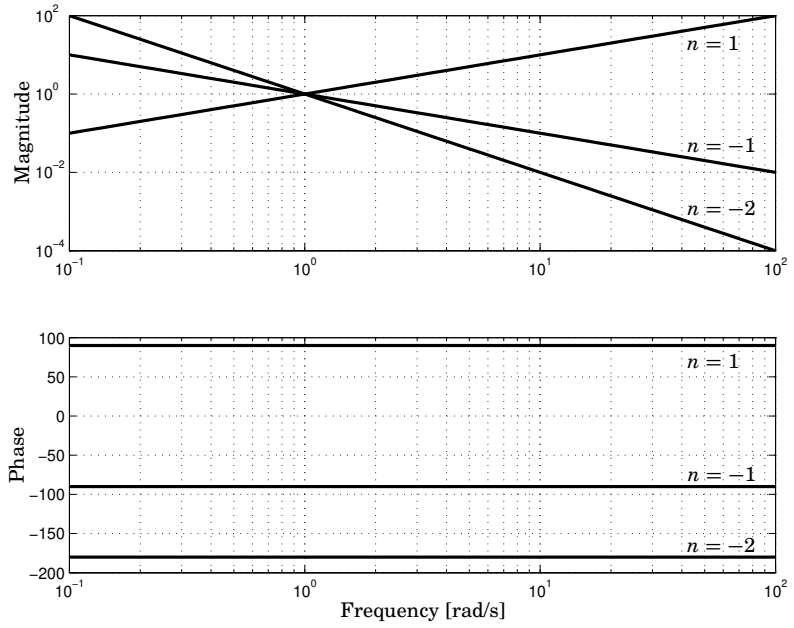

Figure 4.6 Bode plots of $G(s) = s^n$, where $n = 1, -1$ and -2.

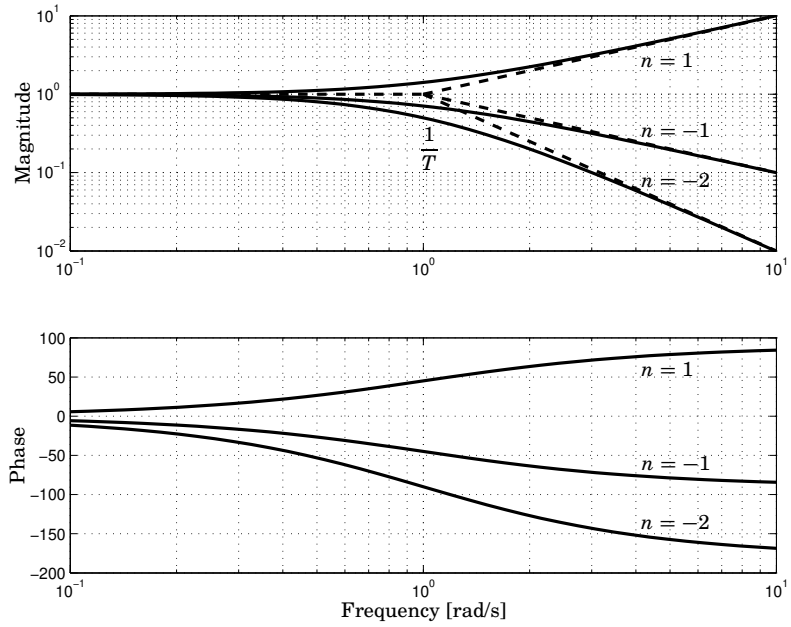

Figure 4.7 Bode plot of $G(s) = (1+sT)^n$, where $T = 1$ and $n = 1, -1$ och -2.

3. Bode Plot of $G(s) = (1+sT)^n$

The magnitude and argument of the transfer function $G(s) = (1+sT)^n$ are given by

$$\log|G(i\omega)| = \log|1+i\omega T|^n = n\log\sqrt{1+\omega^2T^2}$$
$$\arg G(i\omega) = n\arg(1+i\omega T) = n\arctan(\omega T)$$

For small values of ω the functions are given by

$$\log|G(i\omega)| \to 0$$
$$\arg G(i\omega) \to 0$$

For large values of ω the functions are given by

$$\log|G(i\omega)| \to n\log\omega T$$
$$\arg G(i\omega) \to n\frac{\pi}{2}$$

These two asymptotes, the low-frequency and high-frequency asymptotes, are shown in Figure 4.7 together with the Bode plots corresponding to some different values of n. The intersection between the low- and high frequency asymptotes is given by

$$\log\omega T = 0$$

This frequency is called the corner frequency and is given by $\omega = 1/T$.

4. Bode Plot of $G(s) = \left(1 + 2\zeta s/\omega_0 + (s/\omega_0)^2\right)^n$

The low-frequency asymptote of this transfer function is given by $G(i\omega) \approx 1$, i.e.

$$\log|G(i\omega)| \to 0$$
$$\arg G(i\omega) \to 0$$

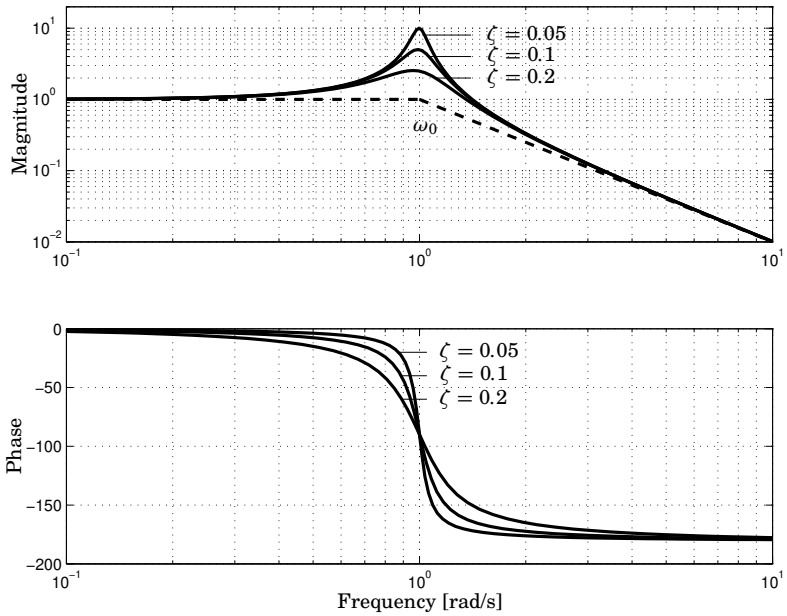

Figure 4.8 Bode plot of $G(s) = \omega_0^2/(s^2 + 2\zeta\omega_0 s + \omega_0^2)$, where $\omega_0 = 1$ and $\zeta = 0.05, 0.1, 0.2$.

For large ω the high-frequency asymptote is given by $G(i\omega) \approx (i\omega/\omega_0)^{2n} = (-1)^n(\omega/\omega_0)^{2n}$, i.e.

$$\log|G(i\omega)| \to 2n \log \frac{\omega}{\omega_0}$$
$$\arg G(i\omega) \to n\pi$$

Figure 4.8 shows the Bode plots for some different values of the parameter ζ. When $\zeta < 1/\sqrt{2}$ there will be a resonance peak close to frequency ω_0. The magnitude of the peak increases when ζ decreases.

5. Bode Plot of $G(s) = e^{-sL}$

This transfer function describes a pure time delay. This means that the output is identical to the input, except that it has been delayed by a time L, $y(t) = u(t - L)$. If one sends a sinusoid through such a process, it outputs a sinusoid with the same amplitude, but with a phase shift which is larger for higher frequencies. For the transfer function $G(s) = e^{-sL}$ the magnitude and argument become

$$\log|G(i\omega)| = \log|e^{-i\omega L}| = 0$$
$$\arg G(i\omega) = \arg e^{-i\omega L} = -\omega L$$

Figure 4.9 shows the Bode plot for some different choices of the delay L.

Drawing the Bode Plot of a composite Transfer Function

Finally we shall illustrate how to draw a Bode plot of a higher order transfer function by factoring it into the sample transfer functions treated above.

Example 4.2—Bode plot drawing

We want to draw the Bode plot of the transfer function

$$G(s) = \frac{100(s+2)}{s(s+20)^2}$$

The Bode plot is shown together with asymptotes in Figure 4.10. If one intends to

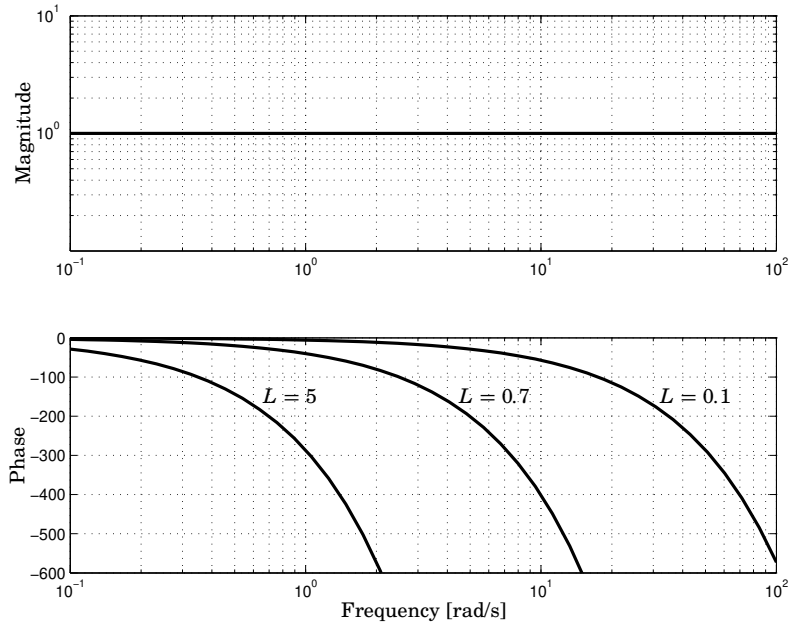

Figure 4.9 Bode plot of $G(s) = e^{-Ls}$ for $L = 5, 0.7$ and 0.1

draw the Bode plot by hand, the first step is to factor the transfer function into a product of the sample transfer functions which we have previously studied.

$$G(s) = \frac{100(s+2)}{s(s+20)^2} = 0.5 \cdot s^{-1} \cdot (1 + 0.5s) \cdot (1 + 0.05s)^{-2}$$

For low frequencies, the two transfer functions to the right are almost one. The ramaining parts form the low frequency asymptote

$$G(s) \to \frac{0.5}{s}$$

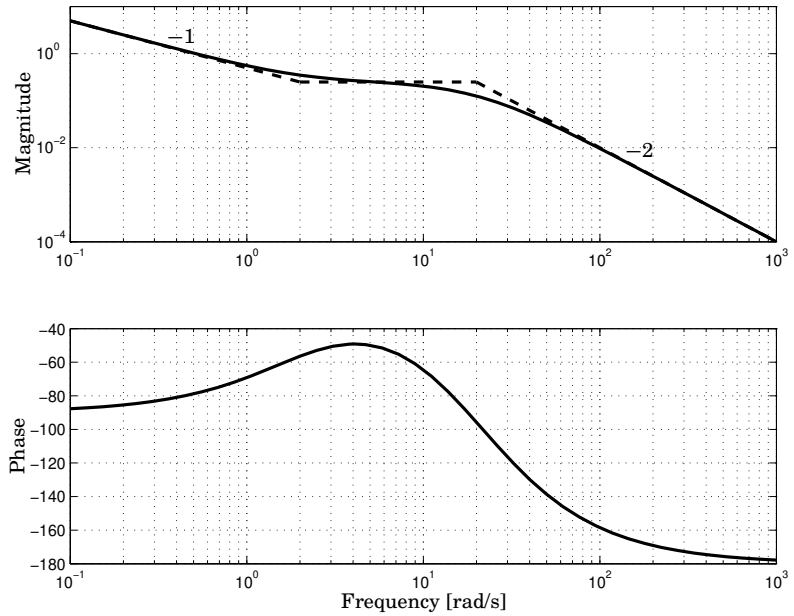

Figure 4.10 Bode plot of $G(s) = \frac{100(s+2)}{s(s+20)^2}$

We begin by drawing the magnitude curve. The low frequency asymptote is a straight line with slope -1. The vertical placement of the line is determined by evaluating the transfer function for a value of $s = i\omega$. For e.g. $\omega = 1$ we obtain $|G(i\omega)| = 0.5$.

The transfer function has two corner frequencies. At $\omega = 2$ the curve breaks up once and we obtain an asymptote with slope 0. At $\omega = 20$ the curve breaks down twice and we obtain an asymptote with slope -2. The high frequency asymptote is given by

$$G(s) \to \frac{100}{s^2}$$

Figure 4.10 shows both the asymptotes and the detailed curve. The phase curve starts at $-90°$ since the low frequency asymptote is an integrator, $G(s) = 0.5/s$. Would the process only have consisted of this integrator, the phase would have remained constantly $-90°$ for all frequencies. Now there is, however, first a corner point at $\omega = 2$, which increases the phase. If there would have been no additional dynamics, the phase would have increased to $0°$ for high frequencies. At $\omega = 20$ we do have another corner point caused by a pole pair. Consequently, the phase decreases and approaches $-180°$ for high frequencies. □

4.4 The Relation between Model Descriptions

We have now introduced several ways of representing process dynamics: state space model, Bode plot, Nyquist curve, step and impulse response and the transfer function. We shall end the modelling part of the course by showing the relation between these representations for some different types of processes.

Single-Capacitive Processes Figure 4.11 shows the singularity plot (poles and zeros), the step response, the Nyquist curve and the Bode plot of a single capacitive—or first order—process. This is a process type with simple dynamics which is easy to control. If we imagine a mercury thermometer with a very thin glass wall, which is immersed into boiling water, we get a response in the mercury column which bears a form corresponding to a single-capacitive process.

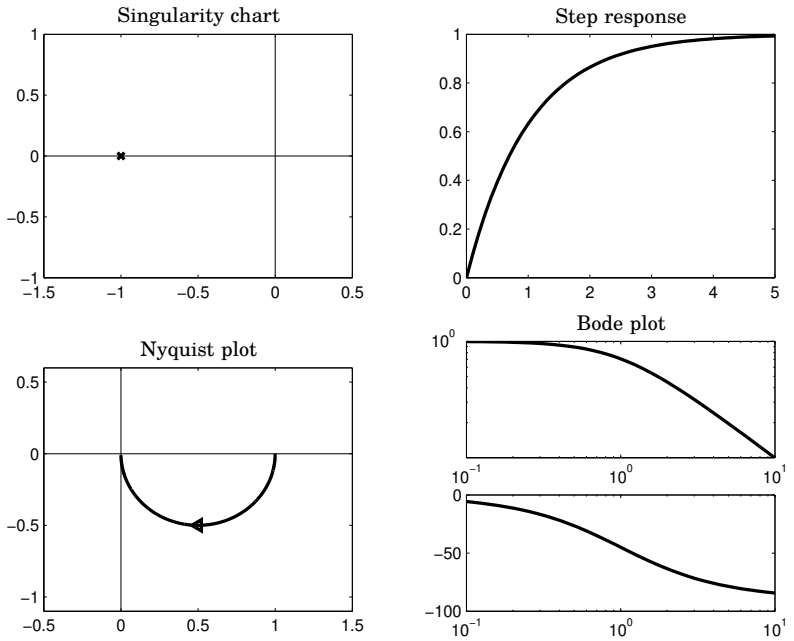

Figure 4.11 Different model descriptions of a single-capacitive process, $G(s) = 1/(s+1)$.

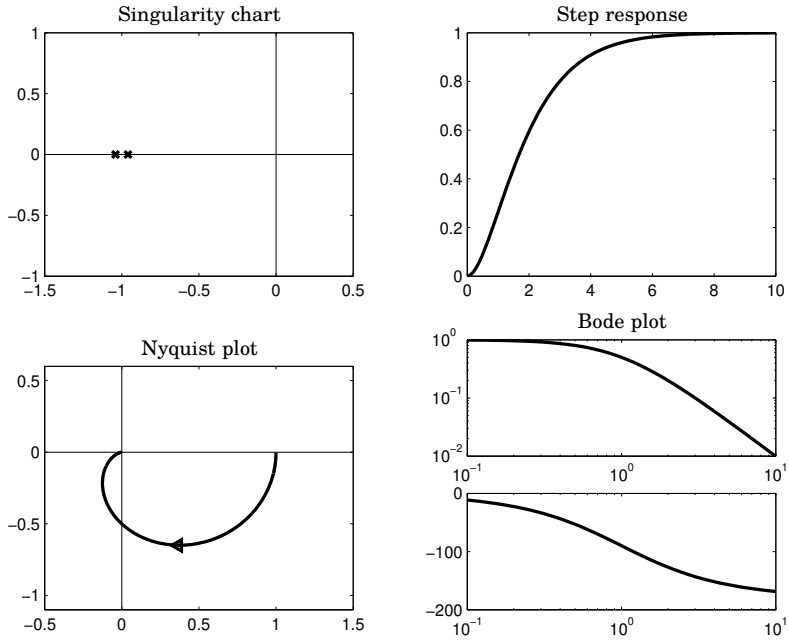

Figure 4.12 Different model descriptions of the multi-capacitive process $G(s) = 1/(s+1)^2$

Multi-capacitive Processes Figure 4.12 shows model descriptions of a multi-capacitive process, in this case a second-order process.

This is a very common process type. Let us conduct the same experiment as with the single-capacitive process, but with the difference that we now immerse the thermometer into cold water. The cold water container is then placed on a heater. The water temperature will rise in a manner characteristic for a multi-capacitive "capacitances" which are heated. The difference between single and multi-capacitive processes increases with the addition of capacitances. This would have been the case if e.g. the mercury thermometer had a thick glass wall or if the heating plate would have been cold when the experiment begun.

Integrating Processes Figure 4.13 shows model descriptions of an integrating process. If the process is integrating, the step response will not converge towards a stationary level, but rather grow linearly. Examples of such processes are level control in tanks, pressure control in sealed vessels, concentration control in containers without inflow and temperature control in well isolated ovens. If one opens the inlet valve, the level will increase linearly given that the outflow remains unaffected. In common for all these processes is that they involve some sort of accumulation. In the level and pressure control cases it is an accumulation of mass, in the temperature control case it is an accumulation of energy.

Oscillative Processes Figure 4.14 shows model descriptions of oscillative processes. These processes have two complex poles. They are further characterized by a step response which oscillates around its final stationary value. Oscillative processes occur mainly in mechanical contexts, when elastic materials are used, e.g. in servo axles, spring constructions etc.

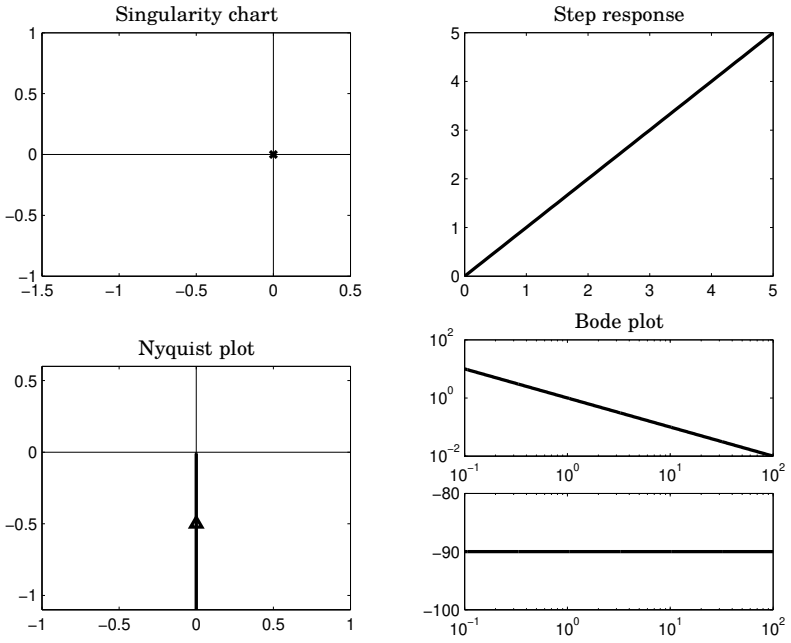

Figure 4.13 Different model descriptions of an integrating process, $G(s) = 1/s$.

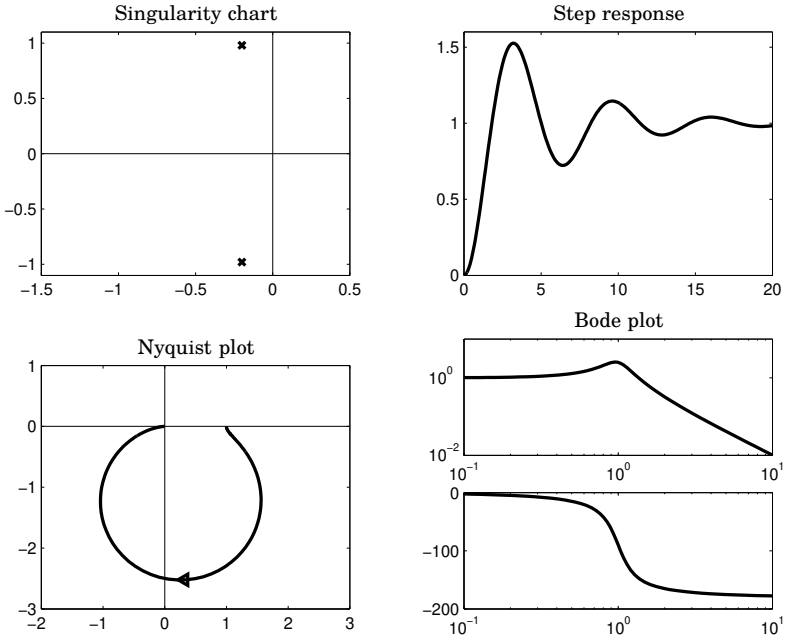

Figure 4.14 Different model descriptions of the oscillative process $G(s) = 1/(s^2 + 0.4s + 1)$.

Delay Processes Figure 4.15 shows model descriptions of a process with a time delay. In this case it is a process with a 1 s delay and a time constant of 0.1 s. Figure 4.15 contains no singularity plot, since the delay cannot be represented in such a plot. This process can not be described in state-space form either.

Delays arise e.g. when materials are transported in pipes or on conveyors. If we for instance measure the pH value in a fluid being transported through a pipe, where the addition of a the substance of which we want to measure the concentration is made upstream with respect to the sensor, we are subject to a delay. It corresponds to the time it takes the fluid to flow from the point of addition to the sensor. Another well known example is the temperature control in a shower. Since a change in

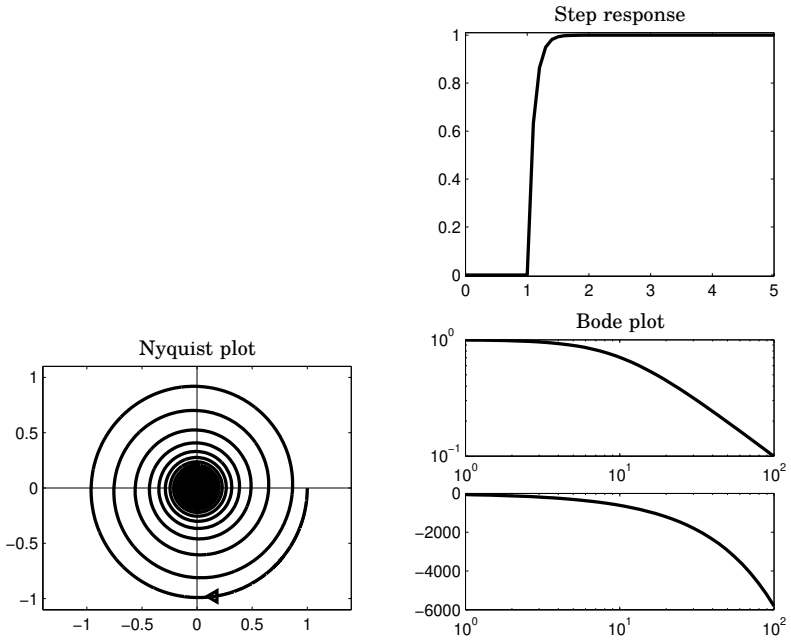

Figure 4.15 Different model descriptions of a process with a delay, $G(s) = e^{-s}/(0.1s + 1)$.

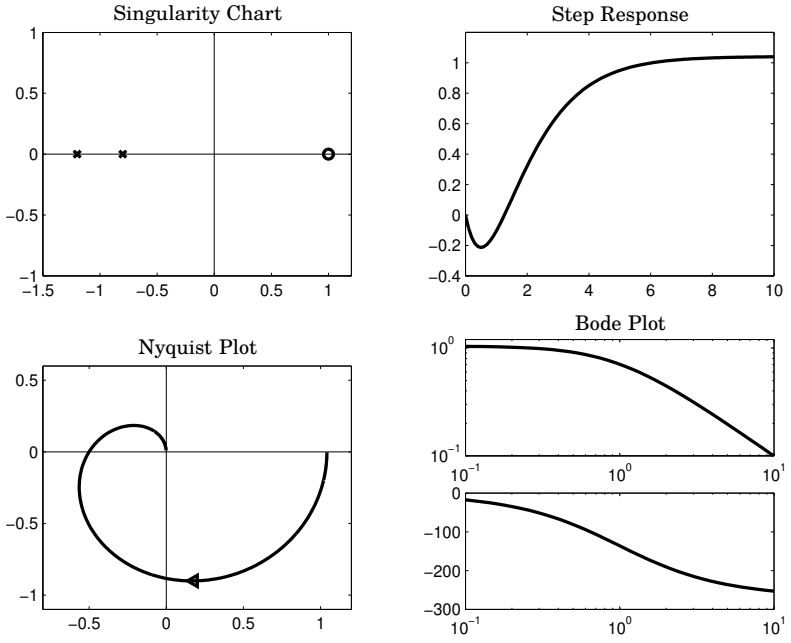

Figure 4.16 Different model descriptions of a process with inverse step response, $G(s) = (1 - s)/((s + 0.8)(s + 1.2))$.

the mix between cold and warm water is not noticeable until the water has been transported through the shower hose, we are subject to a delay corresponding to the time this transportation takes.

Processes with Inverse Responses Figure 4.16 shows model descriptions of a process with an inverse response, i.e. a process where the step response initially goes in the ”wrong” direction. Typical for these processes is that they have zeros in the right-half plane. This will be further investigated later in the course. Processes of this type are relatively difficult to control.

In the process industry a well known example of a process with inverse response

arises when controlling the dome level in steam boilers. If one, for instance, would like to increase the dome level by increasing the flow of water into the dome, the first reaction will be that the water in the dome is cooled. This results in less steam bubbles in the water and consequently the dome level decreases. First after a while, when the water has been heated anew, will the level increase.

Another example of a process with an inverted response is a reversing car. If we intend to pocket-park a car, its center of gravity is first moved in the wrong direction before we can steer it into the pocket.

Lecture 5

Feedback and Stability

We will now leave the process modelling and enter the analysis part of the course. First the properties of feedback are introduced through an example. The example we have chosen is the control of a steam engine. A fundamental requirement is that the control must succeed in keeping the controlled process stable. We define stability and subsequently some methods to determine whether a control loop is stable.

5.1 Feedback—The Steam Engine

The steam engine is one of the earliest example of a control application. The control objective is to maintain a constant angular speed, despite load variations. If one, for instance, lets the steam engine drive a saw, one wants to avoid a change in rotational speed as logs are fed in.

In order to control the steam engine, one must first of all measure its angular speed. This was achieved by a mechanism shown in Figure 5.1.

When the angular speed is increased, the two spheres are forced outwards by the centrifugal force. The vertical position of the shaft thus constitutes a measurement of the angular speed. The shaft was connected to a valve which controlled the steam outflow. Due to its working principle the device is known as a centrifugal governor.

The Uncontrolled Steam Engine

We start by investigating the uncontrolled process, i.e. the steam engine itself. Figure 5.2 shows a block diagram of the process. We call this the open-loop system.

The output is the angular speed ω, which we want to keep constant. The angular speed is affected by two momenta. The driving momentum M_d is determined by us by varying the steam outflow from the engine. We also have a load momentum M_l, which varies with applied load.

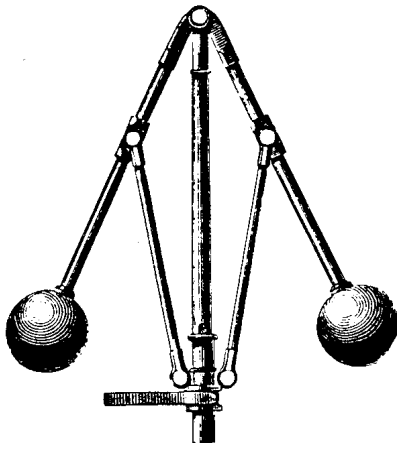

Figure 5.1 Measurement of the angular speed was achieved using the centrifugal governor.

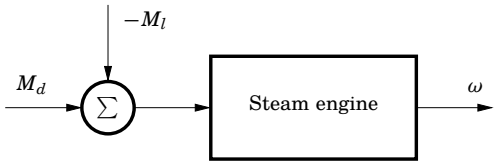

Figure 5.2 Block diagram of the uncontrolled steam engine.

A simple mathematical model of the steam engine is obtained from the momentum equation

$$J\dot{\omega} + D\omega = M_d - M_l \tag{5.1}$$

where J is the momentum of inertia and D is a damping coefficient.

The stationary angular speed ω_s, i.e. the constant angular speed corresponding to constant momenta, is obtained by letting $\dot{\omega} = 0$.

$$\omega_s = \frac{M_d - M_l}{D}$$

We see that we can control the angular speed by choosing a suitable driving momentum M_d, but that it will vary if the load M_l or the damping D are varied. The uncontrolled steam engine is thus sensitive to both loads and process variations.

The dynamical evolution of the system can be analysed by solving the differential Equation (5.1). If we assume that the steam engine is initially at rest, i.e. $\omega = 0$, and that it is started at time $t = 0$, the angular speed is given by

$$\omega(t) = \frac{M_d - M_l}{D}\left(1 - e^{-Dt/J}\right) = \omega_s\left(1 - e^{-Dt/J}\right)$$

The step response is shown in Figure 5.3 and the system settles with a time constant $T = J/D$. The rise time is an alternative measure of the speed of the step response. There are several parallel definitions of the rise time. Sometimes it is defined as the inverse of the maximal time derivative of the measurement

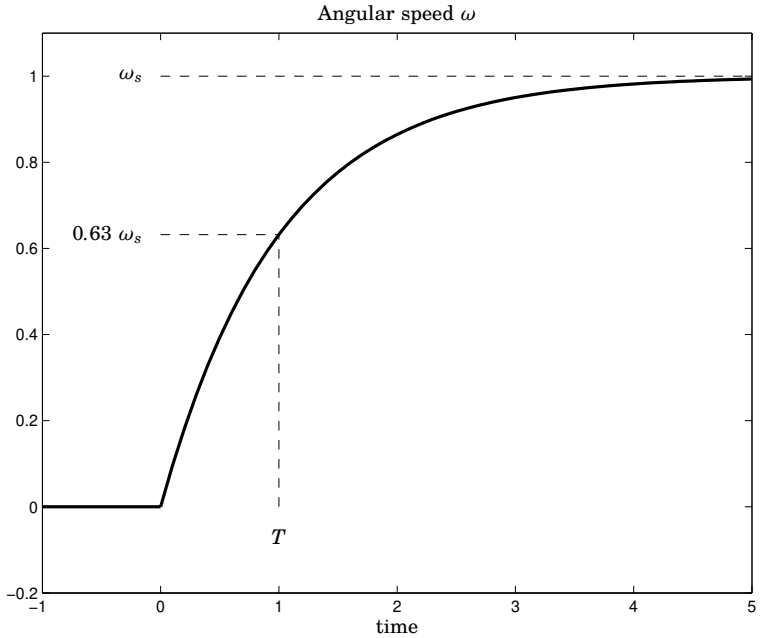

Figure 5.3 The step response of the uncontrolled steam engine. The process parameters are given by $J = D = 1$. The time T is the time constant of the process.

signal. Other definitions include the time it takes for the measurement signal to go between 5% and 95%, or between 10% and 90%, of its final value. The rise time is alternatively referred to as the solution time of the system.

P Control of the Steam Engine

Let us now control the steam engine according to Figure 5.4. This is referred to as the closed loop system. We start by studying a P controller and let the driving momentum be given by

$$M_d = K(\omega_r - \omega)$$

where ω_r is the setpoint value for the angular speed and K is the gain of the controller.

If this controller is applied to the process in Equation (5.1) we obtain the equation of the closed loop system

$$J\dot{\omega} + D\omega = K(\omega_r - \omega) - M_l$$

i.e.

$$J\dot{\omega} + (D + K)\omega = K\omega_r - M_l \tag{5.2}$$

In stationarity, i.e. when $\dot{\omega} = 0$, it holds that

$$\omega_s = \frac{K}{D + K}\omega_r - \frac{1}{D + K}M_l$$

By choosing the controller gain K high, the angular speed ω can be held close to its setpoint value ω_r. We also see that the feedback makes the angular speed less sensitive to variations in both load M_l and the process parameter D.

In order to study the dynamical evolution of the system, we can solve Equation (5.2). If the system is initially at rest, i.e. $\omega = 0$, the solution becomes

$$\omega(t) = \frac{K\omega_r - M_l}{D + K}\left(1 - e^{-(D+K)t/J}\right) = \omega_s\left(1 - e^{-(D+K)t/J}\right)$$

The time constant is given by $T = J/(D + K)$. A high gain K hence results in a faster settling than in the uncontrolled case. Figure 5.5 shows the settling process for some different K, while the process is subject to load disturbances. (In Figures 5.5 and 5.6 the angular speed takes on negative values. This does not mean that the steam engine is reversing, but is explained by the fact that we study deviations from its equilibrium, which we have conveniently defined to be $\omega = 0$.) We can thus establish that the feedback has yielded a faster system which is less sensitive to process variations and load disturbances.

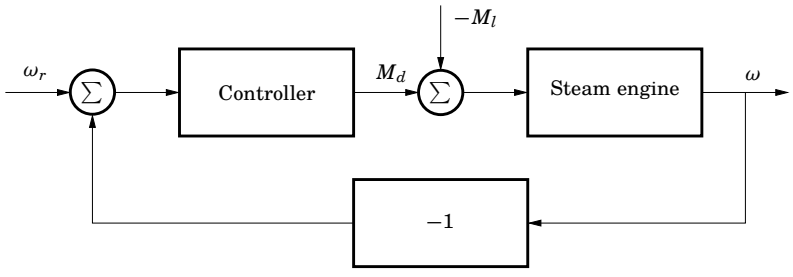

Figure 5.4 Block diagram of the controlled steam engine.

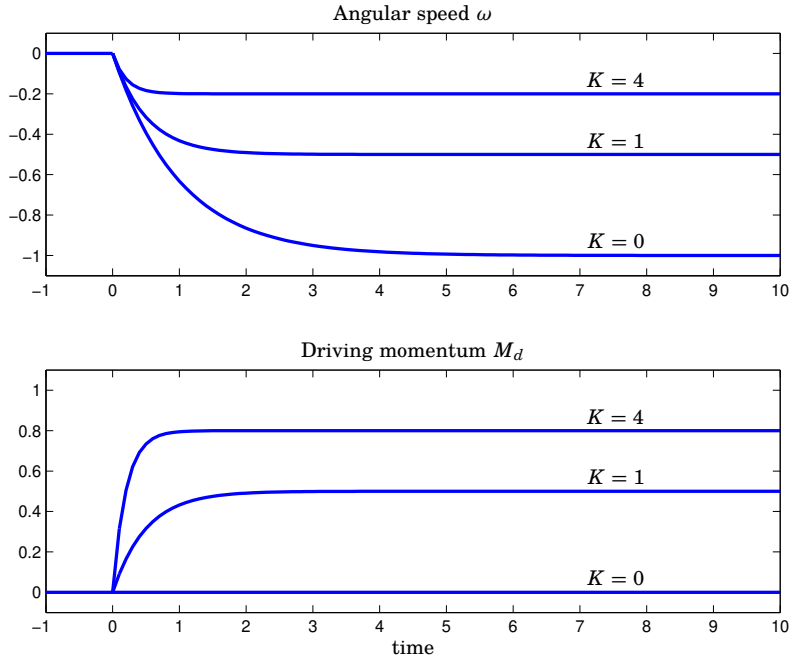

Figure 5.5 P control of the steam engine with gains $K = 0$, 1 and 4. The case $K = 0$ corresponds to the uncontrolled open loop system. The process parameters are given by $J = D = 1$.

PI Control of the Steam Engine

Although the P control improved the properties of the steam engine, the problem with a difference between the angular speed and its setpoint ω_r in stationarity remains. Motivated by this we introduce the PI controller

$$M_d = K(\omega_r - \omega) + \frac{K}{T_i}\int_0^t (\omega_r - \omega)dt$$

where T_i is the integral time. If this controller is applied to the process in Equation (5.1), the following equation is obtained.

$$J\dot{\omega} + D\omega = K(\omega_r - \omega) + \frac{K}{T_i}\int_0^t (\omega_r - \omega)dt - M_l \tag{5.3}$$

By deriving the equation we get rid of the integral and obtain the pure differential equation

$$J\ddot{\omega} + (D + K)\dot{\omega} + \frac{K}{T_i}\omega = K\dot{\omega}_r + \frac{K}{T_i}\omega_r - \dot{M}_l$$

If we assume that the angular speed setpoint and load are constant, it holds that $\dot{\omega}_r = \dot{M}_l = 0$ and we obtain the stationary angular speed

$$\omega_s = \omega_r$$

In stationarity we thus obtain the desired angular speed when utilizing the PI controller. This can be seen in Figure 5.6, which shows the control performance under load disturbances for some different choices of controller parameters. The figure also shows that the dynamics are more complex than previously. The characteristic equation of the closed-loop system is given by

$$Js^2 + (D + K)s + \frac{K}{T_i} = 0$$

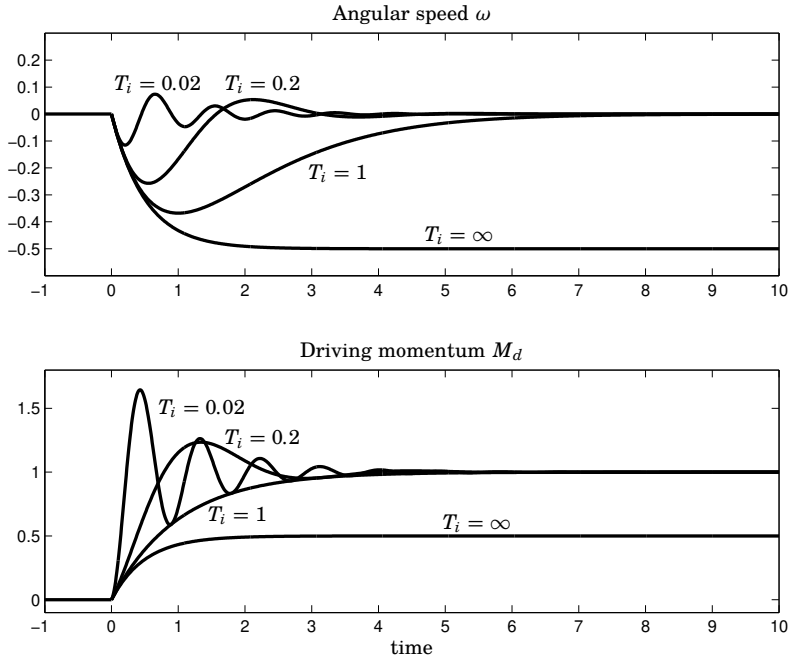

Figure 5.6 PI control of the steam engine with gain $K = 1$ and integral times $T_i = 0.02, 0.2, 1$ and ∞. The case $T_i = \infty$ corresponds to P control. The process parameters are $J = D = 1$.

i.e.

$$s^2 + \frac{D + K}{J}s + \frac{K}{JT_i} = 0 \tag{5.4}$$

In the P control case the closed-loop system had only one pole. The location of the pole was moved as the controller gain K was altered and the control performance was improved with increased gains. In practice, however, we most often deal with higher-order dynamics and excessively high gains lead to poor control.

In the case of PI control we deal with a second-order system with two poles. They can be either real or complex. We see from Equation (5.4) that by altering K and T_i an arbitrary characteristic polynomial, and thereby arbitrary pole placement, can be achieved. This design method is referred to as pole placement.

Figure 5.6 shows that a short integral time T_i leads to oscillations and stability problems. In the following lectures we will discuss how one can choose controller parameters in order to fulfil the specifications on the closed loop system.

Summary

This introductory example has shown us some important properties of feedback. Feedback gives us great advantages. We can keep the controlled variable closer to its setpoint by reducing the sensitivity towards process and load variations. We can also affect the speed of the system, so that we achieve a fast settling after disturbances and setpoint changes.

We have additionally seen that we can have great variations in the settling and that it is thus important how one chooses the closed-loop dynamics by means of the controller. How this is done will be investigated throughout the rest of the course.

5.2 Stability - Definitions

A fundamental requirement in control applications is that we achieve to keep the controlled process stable. We begin by defining the stability concept and then proceed by introducing different methods to conclude whether a process is stable. For

simplicity, the definitions are given for the process dynamics, but they are general and are valid for the closed-loop system as well

For the definitions we choose to consider the system in its state-space form, i.e.

$$\begin{aligned}\dot{x} &= Ax + Bu\\ y &= Cx + Du\end{aligned}$$

> **Asymptotic Stability** A linear system is asymptotically stable if $x(t) \to 0$ as $t \to \infty$ for all initial states if $u(t) = 0$.

Here we consider the deviation from an equilibrium. This means that $u(t) = 0$ corresponds to the state $x(t) = 0$. The definition tells us that independent of their initial values, the state variables will return to their equilibria. Observe that the stability concept is unrelated to the input and output. Stability is a property of the system itself and has nothing to do with how we affect or measure it.

> **Stability** A linear dynamic system is stable if $x(t)$ is bounded for all initial states if $u(t) = 0$.

This definition is weaker than the definition of asymptotic stability, which means that an asymptotically stable system also is stable. We now no longer demand that the state returns to the equilibrium, but only that it remains bounded.

> **Instability** A linear dynamic system is unstable if there exists an initial state, which results in an unbounded state $x(t)$ when $u(t) = 0$.

Example 5.1—Stability in the scalar case
We first investigate the stability concept in the scalar case where we only have one state variable. Since $u(t) = 0$ and the measurement signal is irrelevant in the definitions, it is sufficient to study the equation

$$\begin{aligned}\dot{x}(t) &= ax(t)\\ x(0) &= x_0\end{aligned}$$

with solution

$$x(t) = x_0 e^{at}$$

We obtain three cases depending on the sign of a.

$a < 0$	Asymptotically stable
$a = 0$	Stable
$a > 0$	Unstable

Figure 5.7 shows the responses for the three cases. □

In other words, the sign of a determines the stability in the scalar case.

Example 5.2—Stability in the diagonal case
We now investigate the stability concept in the case when the matrix A is diagonal, i.e. when the state-space equations are given by

$$\begin{aligned}\dot{x}(t) &= \begin{pmatrix} a_1 & & & 0\\ & a_2 & & \\ & & \ddots & \\ 0 & & & a_n \end{pmatrix} x(t) = Ax(t)\\ x(0) &= x_0\end{aligned}$$

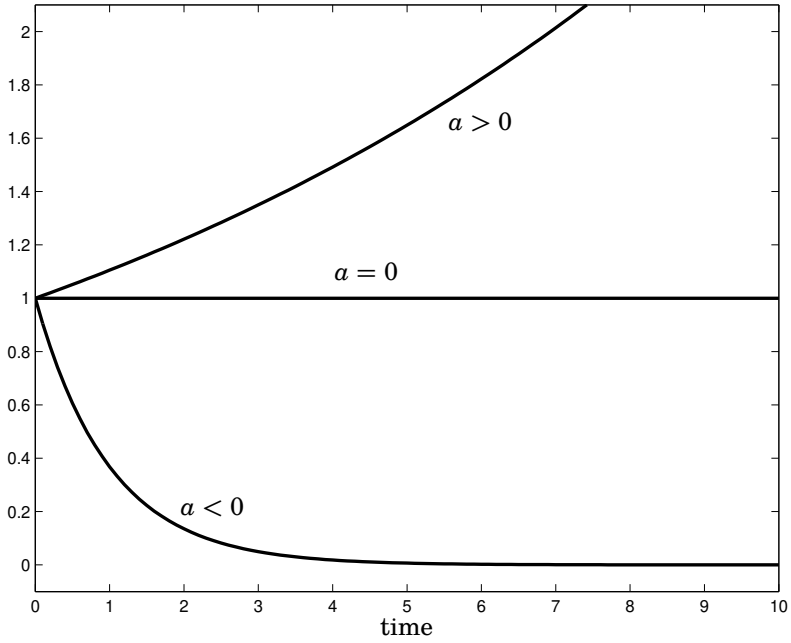

Figure 5.7 Solutions for different choices of the parameter a in Example 5.1 for $x_0 = 1$.

Since every state variable has an equation corresponding to the scalar case

$$\dot{x}_i(t) = a_i x_i(t)$$

where a_i is an eigenvalue of the A matrix, the signs of the eigenvalues determine the stability. We obtain the following cases:

1. If all eigenvalues of the A matrix have negative real parts, the system is asymptotically stable.
2. If any eigenvalue of the A matrix has positive real part, the system is unstable.
3. If all eigenvalues of the A matrix have negative or zero real parts, the system is stable.

□

In the general case, when the A matrix is not diagonal, the rule in Example 5.2 still applies, with the exception that non-positive eigenvalues are an insufficient criterion for stability. It can, however, be shown that the system is stable if any purely imaginary eigenvalues are unique.

5.3 Stability Analysis

From the stability definition we see that stability can be determined from the eigenvalues of the A matrix, which are equivalent to the poles of the transfer function, cf. Lecture 2. In what lies ahead we put our focus on asymptotic stability. The stability analysis thus becomes a matter of investigating whether the roots of the characteristic polynomial of the A matrix, which are identical to the denominator polynomial of the transfer function, lie in the left-half plane.

It can be useful to know the stability criteria for second and third order systems by heart. For a second-order polynomial parametrized as

$$s^2 + a_1 s + a_2$$

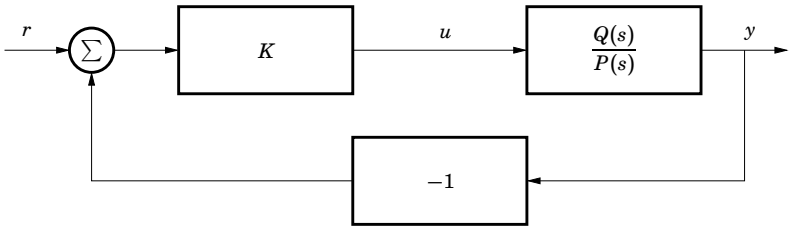

Figure 5.8 The control loop, being analyzed by means of the root-locus method.

the roots lie in the left half plane if and only if (iff) the coefficients a_1 and a_2 are both positive. For a third-order polynomial

$$s^3 + a_1 s^2 + a_2 s + a_3$$

it is also demanded that all coefficients are positive. It is, however, further demanded that

$$a_1 a_2 > a_3$$

For higher-order polynomial computer aid is practical to determine whether the roots lie in the left-half plane.

The Root-Locus Method

In control contexts, one often wants to investigate how the properties of the control loop are affected when a parameter, e.g. a controller parameter, is varied. One way of doing this is by means of the root-locus method. It shows how the poles are translated when a control-loop parameter is varied. The case studied here is described by Figure 5.8. It could be a process with transfer function $Q(s)/P(s)$, which is controlled by a P controller with gain K, where we want to know how the poles of the closed-loop system are affected as K varies. It could also be a more complicated controller, where all dynamics except the gain K are included in the process polynomials $Q(s)$ and $P(s)$. The closed-loop transfer function is given by

$$Y(s) = \frac{KQ(s)}{P(s) + KQ(s)} R(s)$$

We see that the closed-loop zeros coincide with the open-loop zeros. They are given by the nominator polynomial $Q(s)$ and are unaffected by K. The poles will, however, vary with K. The characteristic equation of the closed-loop system is obtained from the nominator polynomial of the transfer function.

$$P(s) + KQ(s) = 0$$

Our task is to investigate how the poles move as K is varied within the interval $[0, \infty]$. We begin by studying the end points. For $K = 0$, the characteristic equation becomes

$$P(s) = 0$$

This implies that the poles of the closed-loop system are identical to the poles of the open-loop system for $K = 0$. For $K \to \infty$ we obtain the equation

$$Q(s) = 0$$

This means that the poles of the closed-loop system will approach its zeros as $K \to \infty$. If the number of poles and zeros are equal, all poles will approach a zero. Normally there are, however, more poles than zeros. The ”left over” poles will then approach infinity. We shall now study a simple example.

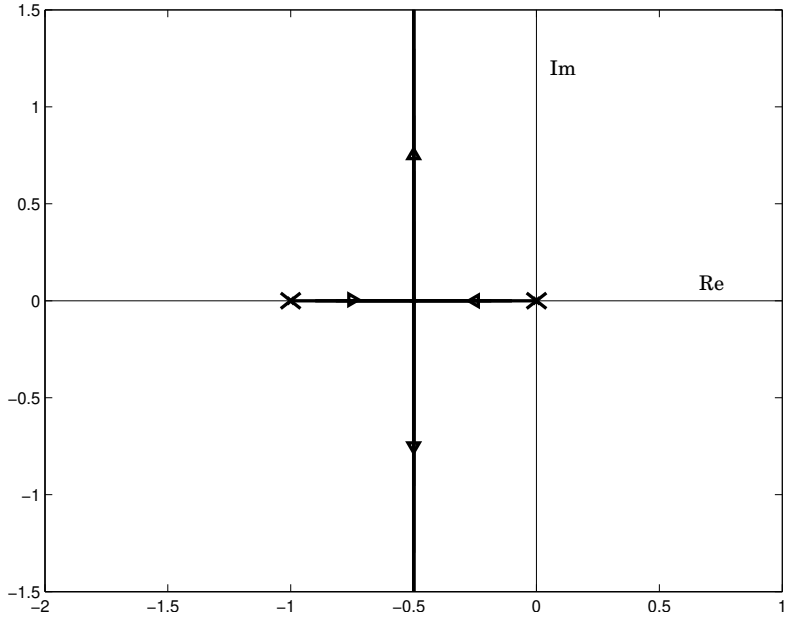

Figure 5.9 Root locus in Example 5.3.

Example 5.3—Root locus o f a second-order system
Assume that we have a control problem according to Figure 5.8 where the process transfer function is given by

$$\frac{Q(s)}{P(s)} = \frac{1}{s(s+1)}$$

The closed-loop characteristic equation is given by

$$P(s) + KQ(s) = s(s+1) + K = 0$$

Since this is a second-order equation, it can easily be solved by hand. We obtain the two poles

$$s = -\frac{1}{2} \pm \sqrt{\frac{1}{4} - K}$$

The root locus is shown in Figure 5.9. For $K = 0$ we observe that the poles lie in 0 and -1, i.e. they correspond to the poles of the open loop system. When K is increased, the two poles will first approach each other along the real axis. As $K = 1/4$ they will both lie in $s = -0.5$. For $K > 1/4$ we obtain a complex-conjugated pole pair with constant real parts and growing imaginary parts as $K \to \infty$. □

For a second-order system the root locus can be drawn by hand. For higher-order systems this becomes complicated and consequently computer aid is recommended. In the following example the root locus of a third order system is studied.

Example 5.4—Root locus of a third-order system
Assume that we have a control problem according to Figure 5.8 where the process transfer function is given by

$$\frac{Q(s)}{P(s)} = \frac{1}{s(s+1)(s+2)}$$

Additionally, the closed-loop characteristic equation is given by

$$s(s+1)(s+2) + K = s^3 + 3s^2 + 2s + K = 0$$

The root locus is shown in Figure 5.10. For $K = 0$ the poles lie in 0, -1 and -2, i.e.

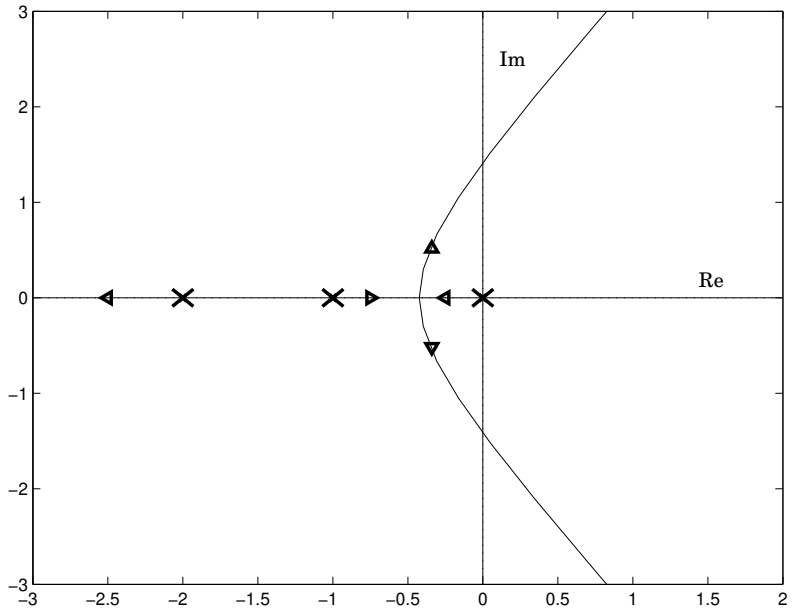

Figure 5.10 The root locus in Example 5.4.

they correspond to the poles of the open loop system. All poles will approach infinity as $K \to \infty$, since there are no zeros. Along their path the poles change properties according to the following:

$0 < K < 0.4$	Three poles in the left-half plane.
$0.4 < K < 6$	One real and two complex poles in the left-half plane.
$K = 6$	A real pole in the left-half plane and two imaginary poles.
$6 < K$	One real pole in the left half plane and two complex poles in the right half plane.

The properties of the control loop vary with the placement of the poles. Figure 5.11 shows step responses corresponding to different values of K. For $0 < K < 0.4$ we have real poles and monotone step responses. For $K > 0.4$ the poles are complex, which results in oscillating responses. The closer to the imaginary axis the poles get, the less damped the responses become. For $K = 6$ the system has reached its stability boundary and for $K > 6$ the system is unstable. □

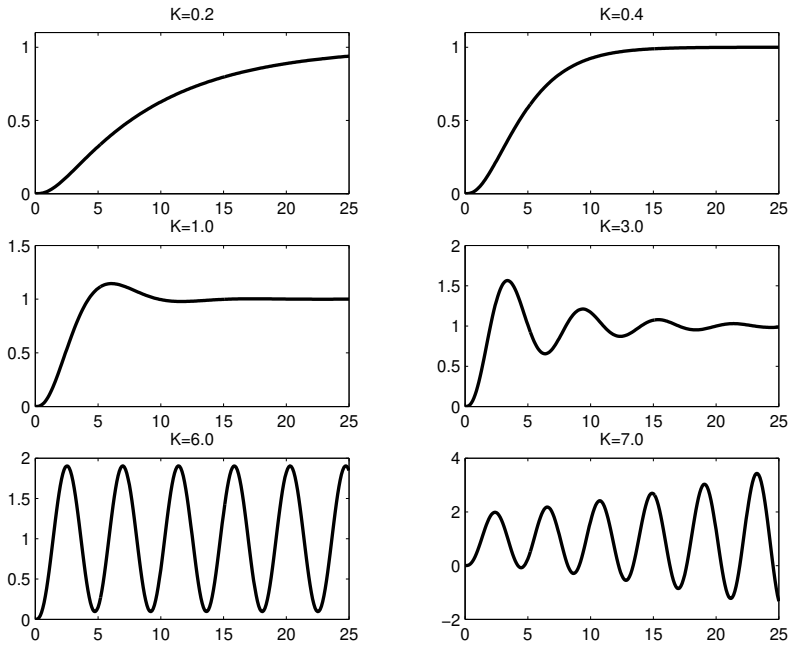

Figure 5.11 Step responses of the system in Example 5.4.

Lecture 6

The Nyquist Criterion and Stability Margins

In the previous lecture we have shown how to determine whether a process or control loop is stable by investigating the poles of its transfer function or the eigenvalues of its A matrix. This lecture will introduce an additional method for determining stability, namely the Nyquist criterion. In this method frequency descriptions, given by either the Bode- or Nyquist plot, are analyzed. Subsequently, we will define some measures of stability margins.

6.1 The Nyquist Criterion

In order to deduce the Nyquist criterion it is required that one is accustomed with Cauchy's argument principle, which is described at the end of this lecture. It will here suffice to give an intuitive explanation to the Nyquist criterion.

Figure 6.1 shows the block diagram of a simple feedback loop where $G_0 = G_R G_P$ is the open-loop transfer function, i.e. the product of the process transfer function G_P and the controller transfer function G_R. There is a switch in the figure, which enables us to cut the feedback. When the switch is in position 1, the control loop functions normally. However, when the switch is in position 2, the feedback is broken and a sinusoid is applied to the open-loop transfer function.

Assume that the switch is in position 2. Under the assumption that the loop transfer function is stable, the signal e will also become a sinusoid. As we introduced the frequency analysis in Lecture 4, we showed that this error signal is given by

$$\begin{aligned} e(t) &= -|G_0(i\omega)| \sin(\omega t + \arg G_0(i\omega)) \\ &= |G_0(i\omega)| \sin(\omega t + \arg G_0(i\omega) + \pi) \end{aligned}$$

Let us choose the frequency of the sinusoid such that $\arg G_0(i\omega) = -\pi$ and denote this frequency ω_0. We then obtain

$$e(t) = |G_0(i\omega_0)| \sin(\omega_0 t)$$

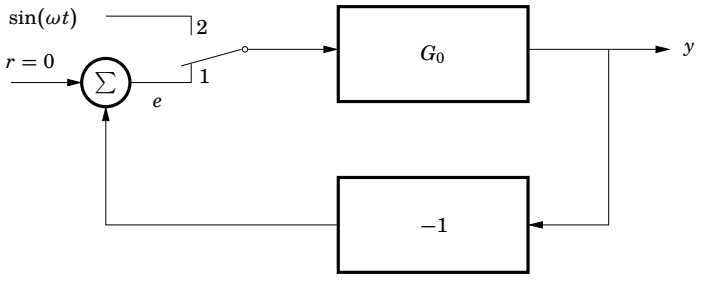

Figure 6.1 The simple feedback loop, being analyzed by means of the Nyquist criterion.

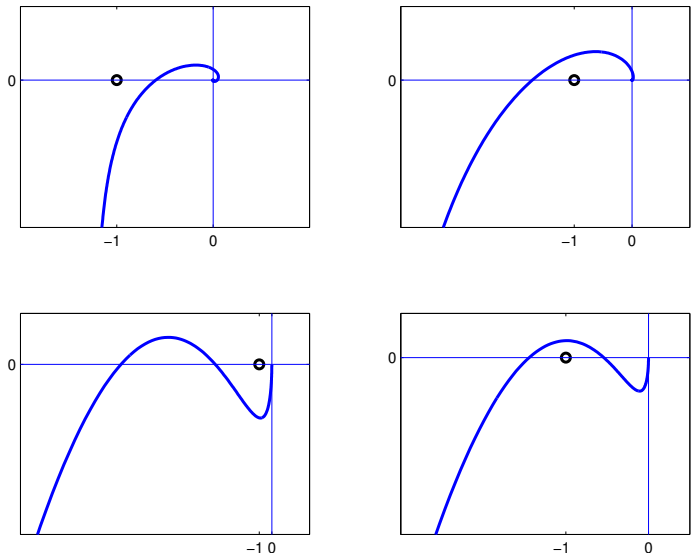

Figure 6.2 Nyquist curves for four different loop transfer functions. According to the Nyquist criterion the two leftmost systems are stable, whereas the two systems to the right are unstable.

Let us further assume that $|G_0(i\omega_0)| = 1$. Then we will obtain the signal

$$e(t) = \sin(\omega_0 t)$$

i.e. the same signal which is sent into the system. If the switch is now toggled from position 2 to position 1, the signals in the control circuit will not be affected. The control loop is caught in a self-induced oscillation. In other words we lie on the stability boundary.

Correspondingly, one can imagine what will happen if $|G_0(i\omega_0)| \neq 1$. Assume that $|G_0(i\omega_0)| > 1$. Then the signal $e(t)$ will have the same frequency and phase as the input, but the amplitude will be larger. If one toggles the switch from position 2 to position 1 under these circumstances, the amplitude in the control loop will grow and an unstable loop is obtained. Analogously, a gain $|G_0(i\omega_0)| < 1$ will imply a decreasing amplitude and a stable control loop.

We can summarize the stability investigation in the following way: Investigate the magnitude of the loop transfer function at the frequency ω_0 for which $\arg G_0(i\omega) = -\pi$. Depending on the magnitude we obtain one of the following cases.

$|G_0(i\omega_0)| < 1$ Stable.

$|G_0(i\omega_0)| = 1$ Stability boundary.

$|G_0(i\omega_0)| > 1$ Unstable.

This intuitive reasoning is unfortunately not always true. It assumes, e.g., that signal components with other freuqencies than ω_0 are damped out, which is not always the truth. It was Nyquist that showed the shortcomings in this reasoning, and therafter he formulated his criterion:

The Nyquist Criterion Assume that the loop transfer function has no poles in the right-half plane, and that possible poles on the imaginary axis are uniqe. Given this, the system is asymptotically stable if the point -1 lies to the left of the Nyquist curve as it is traversed from $\omega = 0$ to $\omega = \infty$.

Figure 6.2 shows Nyquist curves for some different loop transfer functions and the Nyquist criterion interpretation of these.

One advantage of the Nyquist criterion, as opposed to previously introduced stability investigation methods, is that it is applicable also when the system contains

a delay. The Nyquist criterion, however, cannot be used when the system contains poles in the right half plane. For this case one is referred to Cauchy's argument principle, which is described at the end of this lecture.

EXAMPLE 6.1—THIRD-ORDER SYSTEM
We apply the Nyquist criterion on the system, which we previously studied by means of the root-locus method.

$$G_0(s) = \frac{K}{s(s+1)(s+2)}$$

Figure 6.3 shows the Nyquist curve of G_0. As seen, the Nyquist curve starts in the third quadrant. Due to the integrator, it has infinite gain for low frequencies. As ω increases, the Nyquist curve approached the origin. At the frequency ω_0 the Nyquist curve crosses the negative real axis and enters the second quadrant.

In order to use the Nyquist criterion we must first determine the frequency ω_0. This can be done by evaluating $G_0(i\omega)$ and then determining the value of ω for which the Nyquist curve intersects the negative real axis.

$$\begin{aligned} G_0(i\omega) &= \frac{K}{i\omega(1+i\omega)(2+i\omega)} = \frac{-Ki(1-i\omega)(2-i\omega)}{\omega(1+\omega^2)(4+\omega^2)} \\ &= \frac{-Ki(2-\omega^2-3i\omega)}{\omega(1+\omega^2)(4+\omega^2)} = \frac{-3K}{(1+\omega^2)(4+\omega^2)} - i\frac{K(2-\omega^2)}{\omega(1+\omega^2)(4+\omega^2)} \end{aligned}$$

The expression shows that the imaginary part is zero for $\omega = \omega_0 = \sqrt{2}$. The next step in the stability investigation is to determine *where* the Nyquist curve intersects the real axis. This intersection is given by

$$G_0(i\sqrt{2}) = -\frac{3K}{3 \cdot 6} = -\frac{K}{6}$$

The Nyquist curve lies to the right of the point -1 for $K < 6$. The Nyquist criterion thus yields the same stability criterion as obtained when we studied the root locus. □

6.2 Stability Margins

We have now discussed stability concepts and different methods to determine stability. In practice it is not sufficient to determine whether a process is stable. In

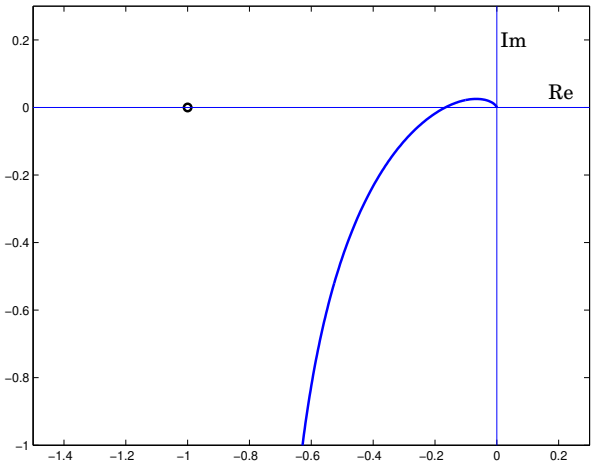

Figure 6.3 The Nyquist curve in Example 6.1, drawn for the case $K = 1$.

addition one wants to know the margins towards the stability limit. We will now introduce three common margins, namely the gain margin, the phase margin and the delay margin.

Gain and Phase Margin

The gain and phase margins are easily defined using the Nyquist plot, see Figure 6.4.

For simplicity we assume that the Nyquist curve of the open loop transfer function G_0 is strictly decreasing, both in magnitude and argument. The gain margin is denoted A_m and determines by how much the gain can be be increased without reaching instability. This margin is read at the frequency ω_0, where the phase shift is π, i.e. $\arg G_0(i\omega_0) = -\pi$. The gain margin is thus given by

$$A_m = 1/\left|G_0(i\omega_0)\right|$$

The phase margin is denoted φ_m and determines by how much the phase shift can be decreased without passing the stability limit. The phase margin can be determined by observing the phase shift in the Nyquist curve at the frequency ω_c, where the magnitude is unity, i.e. $|G_0(i\omega_c)| = 1$. The frequency ω_c is known as the cross-over frequency. The phase margin is given by

$$\varphi_m = \pi + \arg G_0(i\omega_c)$$

The gain and phase margins can also be read from the Bode plot, see Figure 6.5. The critical point -1 in the Nyquist plot corresponds to two horizontal lines in the Bode plot. One line corresponds to the magnitude $|G_0(i\omega)| = 1$ while the other is corresponding to the argument $\arg G_0(i\omega) = -\pi$. The gain margin is obtained as the distance between the line $|G_0(i\omega)| = 1$ and the magnitude curve at the frequency ω_0. The phase margin is obtained as the distance between the line $\arg G_0(i\omega) = -\pi$ and the phase curve.

It is important to maintain reasonable stability margins, since it allows for some variations in process dynamics. This is, however, not the only reason. The stability margins and the distance to the critical point -1 are also decisive for the control performance. If the stability margins are inadequate, jerky and poorly damped control is obtained. On the other hand, large stability margins result in slow control.

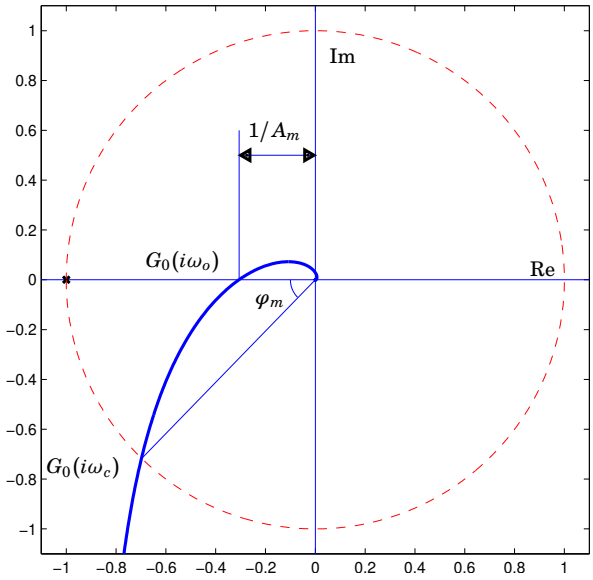

Figure 6.4 Determination of the phase margin φ_m and the gain margin A_m in the Nyquist plot.

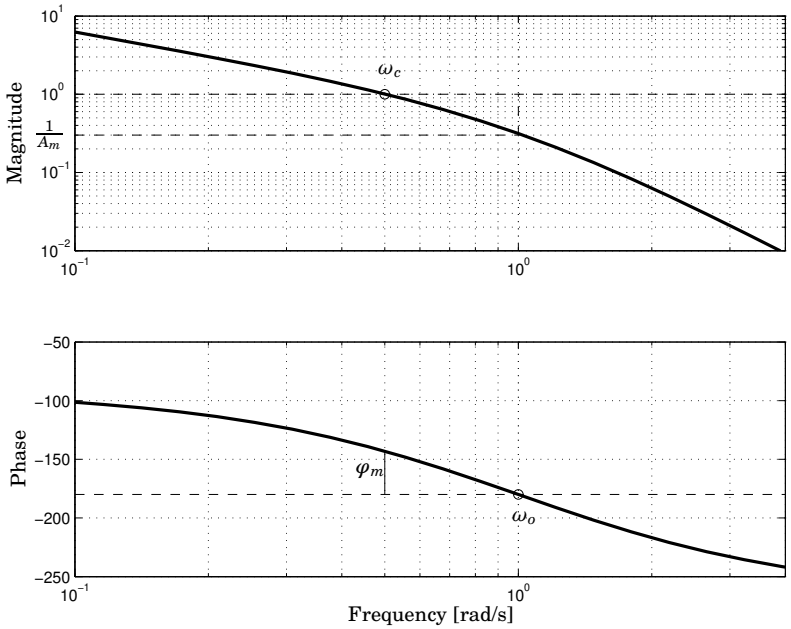

Figure 6.5 Determining the phase margin φ_m and the gain margin A_m in the Bode plot.

Customary rules of thumb prescribe gain and phase margins within the intervals $A_m \in [2, 6]$ and $\varphi_m \in [45°, 60°]$, respectively.

Delay Margin

The delay margin determines the length of an added delay required to drive the control loop unstable. The delay margin has no interpretations as a distance in the Nyquist plot, as we have seen for the gain and phase margins.

Assume that the open-loop transfer function $G_0(s)$ is augmented with a delay. The new loop transfer function thus becomes

$$G_0^{new}(s) = e^{-sL} G_0(s)$$

where L is the delay. The gain and phase shift of the new transfer function are given by

$$|G_0^{new}(i\omega)| = |G_0(i\omega)|$$
$$\arg G_0^{new}(i\omega) = \arg G_0(i\omega) - \omega L$$

The gain is thus not affected by the delay, while the phase shift decreases. Assume that the nominal loop transfer function G_0 has cross-over frequency ω_c, i.e. that $|G_0(i\omega_c)| = 1$, and that the corresponding phase margin is denoted φ_m. Since G_0^{new} has the same gain as G_0, the cross-over frequency of G_0^{new} will also be ω_c. The phase margin will, however, decrease since the phase shift has decreased. The new phase margin becomes

$$\varphi_m^{new} = \varphi_m - \omega_c L$$

If the delay is excessive, the phase margin vanishes and the closed-loop system becomes unstable. This occurs when

$$\omega_c L = \varphi_m$$

This gives us the following bound on how long delays can be before causing an unstable system

$$L_m = \frac{\varphi_m}{\omega_c}$$

The delay L_m is known as the delay margin and is a robustness margin in the same way as the gain margin A_m and the phase margin φ_m.

Obviously, we cannot allow delays close to L_m. The limit

$$\omega_c L < 0.2$$

is a good rule of thumb which guarantees a phase margin decrease of at most 12°. The equation also reveals how this criterion can be met. Either the delay L must be kept sufficiently short or one has to limit the cross-over frequency ω_c, i.e. limit the speed of the system.

6.3 Cauchy's Argument Principle (not included in the course)

We shall now briefly summarize Cauchy's argument principle. Subsequently, we will see how it can be used in a control context to determine stability.

Assume that we have a function $F(s)$ which is analytic, except for a in a finite number of poles $p_1,\ p_2, \ldots, p_P$, on a complex set bounded by the curve C. See Figure 6.6. Further assume that the function has a finite number of zeros $n_1,\ n_2, \ldots, n_N$ inside C.

Now investigate how the curve C is mapped by the function F and especially study how many times the curve $F(C)$ encircles the origin. Observe that the curve has orientation and hence that the number of encirclements can be both positive (counter clockwise) and negative (clockwise). Let P and N denote the number of poles and zeros, respectively, enclosed by C. Cauchy's argument principle states that

$$N - P = \frac{1}{2\pi}\Delta_C \arg F(s)$$

Put in words, the formula says that the number of zeros (N) minus the number of poles (P) equals the number of times the curve $F(C)$ encircles the origin. In Figure 6.6 $N = 4$, $P = 2$ and $F(C)$ encircles the origin twice in the positive direction.

A Control Theory Application We shall now see how Cauchy's argument principle can be applied in order to determine the stability of a control loop. We study the simple feedback loop shown in Figure 6.1. The closed-loop transfer function is given by

$$G(s) = \frac{G_0(s)}{1 + G_0(s)}$$

The location of the poles are given by the denominator polynomial of $G(s)$. We thus define $F(s)$ as

$$F(s) = 1 + G_0(s)$$

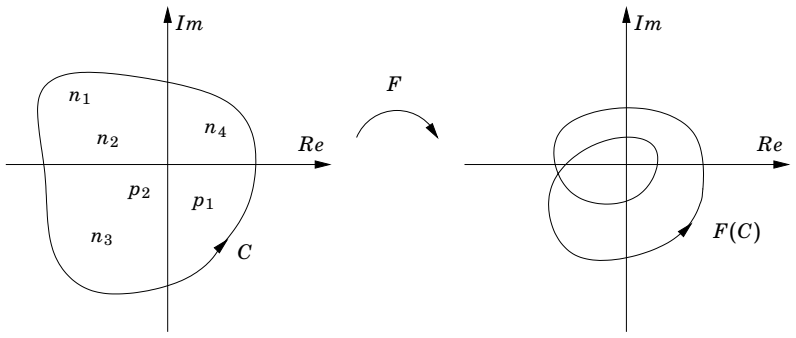

Figure 6.6 Mapping illustrating Cauchy's argument principle.

Since we are interested in knowing if there are any poles in the right half plane, we would prefer to choose C such that it encloses the right half plane. The curve we choose is shown in Figure 6.7. If we let $R \to \infty$ and $r \to 0$, the curve C will enclose the right half plane. The reason for introducing the small half circle is that there must not lie any poles or zeros on C while poles in the origin are commonly occurring. This is true for e.g. the case when the controller contains an integrator.

Since we have chosen $F(s) = 1 + G_0(s)$ it holds that

$$\begin{aligned} N &= \text{Number of zeros of } 1 + G_0 \text{ enclosed by } C. \\ &= \text{Number of poles of } G \text{ enclosed by } C. \\ P &= \text{Number of poles of } 1 + G_0 \text{ enclosed by } C. \\ &= \text{Number of poles of } G_0 \text{ enclosed by } C. \end{aligned}$$

Let us investigate the last equation in order to realize its validity. If we assume that G_0 is the fraction between two polynomials we have that

$$G_0(s) = \frac{Q(s)}{P(s)} \qquad \Rightarrow \qquad 1 + G_0(s) = \frac{P(s) + Q(s)}{P(s)}$$

Since $G_0(s)$ and $1 + G_0(s)$ share a common denominator polynomial, they also have the same poles. Cauchy's argument principle thus states that

$$N - P = \text{Number of times } 1 + G_0(C) \text{ encircles the origin.}$$

Rather than studying how many times $1 + G_0$ encircles the origin, we choose to study how many times G_0 encircles the point -1. The procedure is summarized below.

1. Draw $G_0(s)$ as the curve C is traversed.
2. Count the number of times, n, which the curve encircles the point -1.
3. Cauchy's argument principle yields $N - P = n$.
4. If $N = 0$ there are no poles in the right half plane and consequently the closed-loop system is asymptotically stable.

The Nyquist criterion is a special case of Cauchy's argument principle. It treats the case when the open loop transfer function does not have any poles in the right half plane, i.e. the case when $P = 0$. Cauchy's argument principle then states that the closed-loop system is stable ($N = 0$) if $G_0(C)$ does not enclose the point -1. For this case it is sufficient to study the Nyquist curve. Rather than evaluating the map of the entire curve C, it will suffice to study the map of the positive imaginary axis.

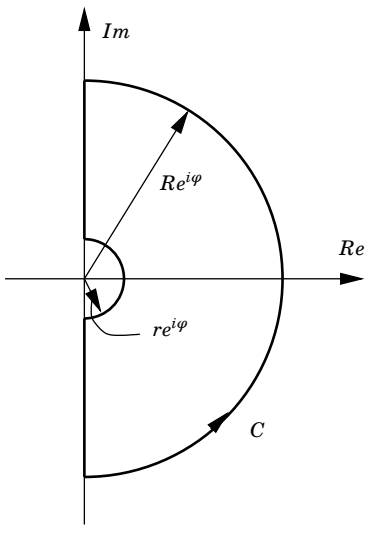

Figure 6.7 The curve C in an automatic control application of Cauchy's argument principle.

EXAMPLE 6.2—THIRD ORDER SYSTEM
We apply Cauchy's argument principle to the same system which was previously studies in Example 6.1, i.e.

$$G_0(s) = \frac{K}{s(s+1)(s+2)}$$

The first step is to determine $G_0(C)$. In order to do this we divide the curve C into segments. To begin with we compute the map of the positive imaginary axis, i.e. the Nyquist curve. We have already done this in Example 6.1 and the curve is shown in Figure 6.3. We do not have to compute the map of the negative imaginary axis, since it is simply the Nyquist curve mirrored in the real axis.

The map of the large half circle is given by

$$G_0(Re^{i\varphi}) \to 0, \quad R \to \infty$$

The large half circle is thus mapped onto the origin. The small half circle yields

$$G_0(re^{i\varphi}) \to \frac{K}{2re^{i\varphi}} = \frac{K}{2r}e^{-i\varphi}, \quad r \to 0 \qquad \varphi : \frac{\pi}{2} \to 0 \to -\frac{\pi}{2}$$

The small half circle is thus mapped onto a half circle with infinitely large radius. It starts with argument $-\pi/2$, passes 0 and ends with argument $\pi/2$.

Figure 6.8 shows $G_0(C)$. The figure also shows an enlargement of the interesting part around the point -1 for the case $K = 1$.

The Nyquist curve thus lies to the right of the point -1 for $K < 6$. We have once again arrived at the same stability criterion. For $K > 6$ the curve will encircle the point -1 twice, which according to Cauchy's argument principle tells us that there are two poles in the right half plane. We came to this conclusion in the last lecture, as we studied the root locus of the system. □

Cauchy's argument principle and the Nyquist criterion are significantly more complicated methods than those presented previously. On the other hand they provide an important insight into and understanding of control theory. E.g. the methods show that the properties of the control loop in the frequency range around $-180°$ are critical for stability, robustness and performance of the closed-loop system.

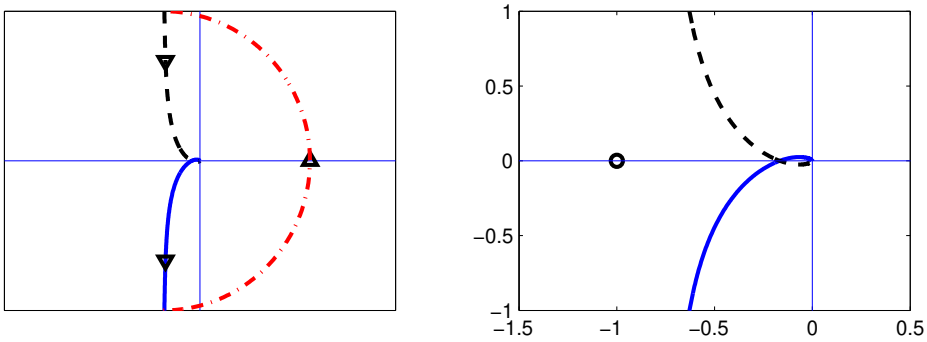

Figure 6.8 The mapping $G_0(C)$ from Example 6.2 is shown to the left. To the right we see an enlargement of the area around the origin.

Lecture 7

The Sensitivity Function and Stationary Errors

At the end of the previous lecture we defined three different stability margins. We will begin this lecture by introducing the sensitivity function, which can be interpreted in terms of a stability margin. Then we will treat another demand, which is usually put on the control loop: we expect the control error to vanish in stationarity.

7.1 The Sensitivity Function

In a simple feedback loop, with open-loop transfer function $G_0 = G_R G_P$, where G_P is the process transfer function and G_R is the transfer function of the controller, the sensitivity function is given by

$$S = \frac{1}{1 + G_P G_R} \tag{7.1}$$

The sensitivity function is a transfer function which can be interpreted in several ways. We shall here present three of its interpretations.

The Sensitivity Function as a Stability Margin

Figure 7.1 shows the interpretation of the sensitivity function in the Nyquist plot.

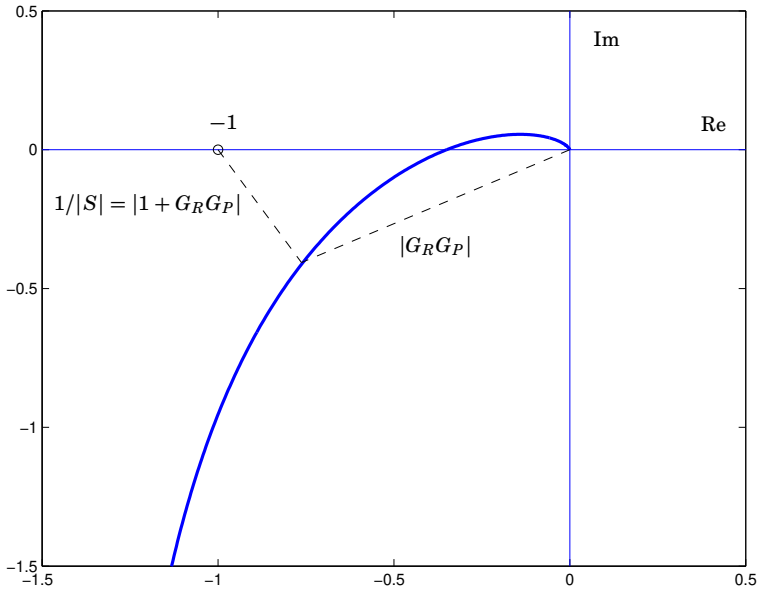

Figure 7.1 The sensitivity function interpreted in the Nyquist plot.

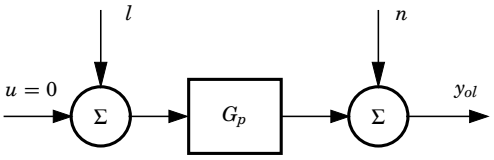

Figure 7.2 Block diagram of the open-loop system.

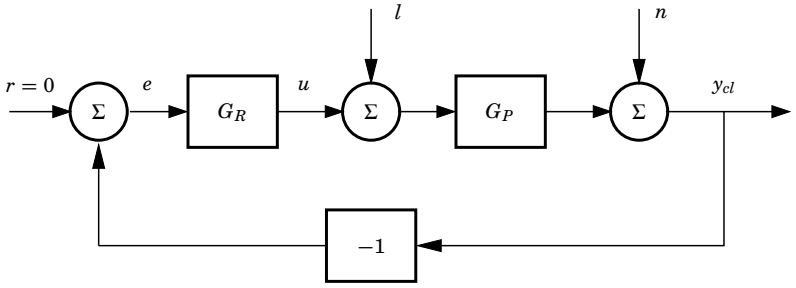

Figure 7.3 Block diagram of the closed-loop system.

Every point on the Nyquist curve is given by the value of the loop transfer function, i.e. $G_R(i\omega)G_P(i\omega)$, evaluated at the corresponding frequency. The distance between the point -1 and the corresponding point on the Nyquist curve is thus $|1 + G_R(i\omega)G_P(i\omega)|$. Since the sensitivity function is given by Equation (7.1), we see that $1/|S(i\omega)|$ is the distance to the critical point -1. A small value of the sensitivity function yields a robust system, since it implies a large margin to the stability limit.

The maximum value of the sensitivity function is denoted M_s.

$$M_s = \max_{\omega} |S(i\omega)|$$

Figure 7.1 shows that M_s is given by the inverse of the shortest distance to the Nyquist curve. The value of M_s is therefore an interesting stability margin, in the same way as the gain, phase and delay margins, which were defined in the previous lecture. Often, one tries to keep M_s in the range $[1.2, 2]$.

The Sensitivity Function as a Measure of Disturbance Rejection

Figure 7.2 shows the block diagram of a process with transfer function G_P, being subject to a load disturbance l and measurement noise n. Disturbances may affect the process in many ways, but they are normally represented by these two disturbances. Load disturbances enter at the process input in the same way as the control signal. People that enter a room generate heat in the same was as heating radiators and are therefore load disturbances for the temperature control. Measurement disturbances affect the measurement signal. They are often electrical disturbances of high frequency and are thefore often called noise. When the control signal is $u = 0$, the process output becomes

$$Y_{ol}(s) = N(s) + G_P(s)L(s)$$

Assume that we close the loop, in which the controller is described by G_R, according to Figure 7.3. Since the setpoint is $r = 0$, the output from the control loop becomes

$$Y_{cl}(s) = \frac{1}{1 + G_R(s)G_P(s)}N(s) + \frac{G_P(s)}{1 + G_R(s)G_P(s)}L(s)$$

The ratio between the outputs of the open and closed-loop systems becomes

$$\frac{Y_{cl}(s)}{Y_{ol}(s)} = \frac{1}{1 + G_P(s)G_R(s)} = S(s). \tag{7.2}$$

Hence, the sensitivity function describes how disturbances are affected by the feedback. Disturbances with frequency ω, for which $|S(i\omega)| < 1$, are rejected, while disturbances for which $|S(i\omega)| > 1$ are amplified by the feedback.

The Sensitivity Function as a Measure of Sensitivity to Modelling Errors

A third interpretation of the sensitivity function is obtained by investigating the effects of modelling errors on the measurement signal. Assume that we have deduced a model G_p of the process to be controlled. Realistically this model is not an exact description of the real process, which we have chosen to denote G_P^0. The closed-loop systems obtained using the model and the true transfer function, respectively, become

$$Y = \frac{G_R G_P}{1 + G_R G_P} R \qquad Y^0 = \frac{G_R G_P^0}{1 + G_R G_P^0} R$$

where Y and Y^0 are the measurement signals obtained for the model and the true system.

The relative error in the measurement signal is

$$\frac{Y^0 - Y}{Y} = \frac{\frac{G_R G_P^0}{1 + G_R G_P^0} R - \frac{G_R G_P}{1 + G_R G_P} R}{\frac{G_R G_P}{1 + G_R G_P} R} = \frac{G_P^0(1 + G_R G_P) - G_P(1 + G_R G_P^0)}{G_P(1 + G_R G_P^0)}$$

$$= \frac{1}{1 + G_R G_P^0} \cdot \frac{G_P^0 - G_P}{G_P} = S^0 \cdot \frac{G_P^0 - G_P}{G_P}$$

where S^0 denotes the sensitivity function of the *true* system, i.e. the sensitivity function obtained using G_P^0 instead of G_P in Equation (7.1). We can summarize the expression as

$$\frac{Y^0 - Y}{Y} = S^0 \cdot \frac{G_P^0 - G_P}{G_P}$$

We can now observe that the sensitivity function is a transfer function from the relative modelling error to the relative error in measurement signal. Once again we see that it is favorable to have a small sensitivity function. Modelling errors affect the measurement signal less at frequencies for which the sensitivity function is small. Figure 7.1 shows that the sensitivity function achieves a maximum at the frequency approximately corresponding to the phase shift $-180°$. In other words, this is the frequency around which it is important to have an accurate model of the process to be controlled.

7.2 Stationary Errors

We shall now investigate stationary errors and the properties required by a controller in order to avoid them. The problem which we shall investigate is illustrated in Figure 7.4. We study the simple feedback loop with two inputs, the setpoint r and a load disturbance l. This gives us the two transfer functions

$$Y = \frac{G_R G_P}{1 + G_R G_P} R + \frac{G_P}{1 + G_R G_P} L$$

The measurement signal is thus affected both by changes in the setpoint value and load disturbances. In many contexts one chooses to isolate these two cases.

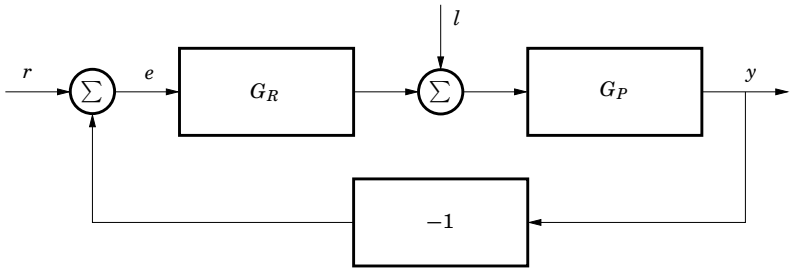

Figure 7.4 The simple feedback loop.

The Servo Problem The servo problem is the case for which $l = 0$, i.e. the case where we are only interested in the setpoint tracking properties of the controller. This is commonly occurring in motor control, e.g. in vehicles and industry robots. When considering the servo problem, the transfer function is given by

$$Y = \frac{G_R G_P}{1 + G_R G_P} R = \frac{G_0}{1 + G_0} R \tag{7.3}$$

where $G_0 = G_R G_P$ denotes the open-loop transfer function.

The Regulator Problem The regulator problem is another name of the case $r = 0$. Here we only study the effect of load disturbances. This problem is commonly occurring in process industry where setpoint values are often constant during long periods of time, whereas loads are continuously changing. In this context the transfer function is given by

$$Y = \frac{G_P}{1 + G_R G_P} L \tag{7.4}$$

We shall treat the servo and regulator problems independently, beginning with the servo problem.

Stationary Errors—The Servo Problem

The servo problem control error can be obtained by means of Equation (7.3).

$$E(s) = R(s) - Y(s) = \frac{1}{1 + G_0(s)} R(s)$$

The final-value theorem is used to determine the control error $e(t)$ as $t \to \infty$.

$$e_\infty = \lim_{t \to \infty} e(t) = \lim_{s \to 0} sE(s)$$

Observe that the final-vale theorem can only be applied when the limit exists. This is the case if and only if the signal $sE(s)$ has all its poles in the left-half plane. The proof of the final-value theorem is given by the definition of the Laplace transform:

$$\lim_{s \to 0}(sE(s) - e(0)) = \lim_{s \to 0} \int_0^\infty e^{-st} \dot{e}(t) dt = \int_0^\infty \dot{e}(t) dt = \lim_{t \to \infty}(e(t) - e(0))$$

We will now study an example before proceeding with the general case.

Example 7.1—Stationary error in step and ramp setpoint tracking
Assume that the process and controller are described by the transfer functions

$$G_P = \frac{1}{s(1 + sT)} \qquad G_R = K$$

The process model could e.g. describe an electric motor with current as input and shaft angle as output. The controller is a P controller with gain K. Together they yield the open-loop transfer function

$$G_0 = G_R G_P = \frac{K}{s(1+sT)}$$

Assume that the setpoint is constituted by a step

$$r(t) = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases}$$

The Laplace transformation of a step is given by

$$R(s) = \frac{1}{s}$$

The control error thus becomes

$$E(s) = \frac{1}{1+G_0(s)} R(s) = \frac{s(1+sT)}{s(1+sT)+K} \cdot \frac{1}{s}$$

By means of the final-value theorem we can now compute the stationary error

$$e_\infty = \lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} \frac{s(1+sT)}{s(1+sT)+K} = 0$$

The limit exists under the assumption that the parameters K and T are positive. We thus see that there is no persisting control error, despite the use of only a P controller. The reason for this will be revealed soon as we will study the general case. But first we investigate the case where the setpoint is a ramp

$$r(t) = \begin{cases} t & t \geq 0 \\ 0 & t < 0 \end{cases}$$

The Laplace transformation of a ramp is given by

$$R(s) = \frac{1}{s^2}$$

The control error thus becomes

$$E(s) = \frac{1}{1+G_0(s)} R(s) = \frac{s(1+sT)}{s(1+sT)+K} \cdot \frac{1}{s^2}$$

This yields the stationary error

$$e_\infty = \lim_{t\to\infty} e(t) = \lim_{s\to 0} sE(s) = \lim_{s\to 0} \frac{1+sT}{s(1+sT)+K} = \frac{1}{K}$$

It is true also for this case that the limit exists if the parameters K and T are positive. When the setpoint is given by a ramp we hence have a persisting control error. Its magnitude is inversely proportional to the controller gain K. □

The General Case We shall now study the general case. We assume that the loop transfer function is given in the form

$$G_0(s) = \frac{K}{s^n} \cdot \frac{1 + b_1 s + b_2 s^2 + \cdots}{1 + a_1 s + a_2 s^2 + \cdots} e^{-sL} = \frac{KB(s)}{s^n A(s)} e^{-sL}$$

This covers most of the processes and controllers we will encounter. It contains a low-frequency part given by K/s^n, a delay L and two polynomials with static gains 1, i.e. $A(0) = B(0) = 1$.

We let the setpoint be given by

$$r(t) = \begin{cases} t^m/m! & t \geq 0 \\ 0 & t < 0 \end{cases}$$

where m is a positive integer. The Laplace transformation of the setpoint is given by

$$R(s) = \frac{1}{s^{m+1}}$$

The control error for this general case becomes

$$E(s) = \frac{1}{1 + G_0(s)} R(s) = \frac{s^n A(s)}{s^n A(s) + KB(s)e^{-sL}} \cdot \frac{1}{s^{m+1}}$$

The stationary error can be computed by means of the final-value theorem.

$$e_\infty = \lim_{t \to \infty} e(t) = \lim_{s \to 0} sE(s) = \lim_{s \to 0} \frac{A(s)}{s^n A(s) + KB(s)e^{-sL}} \cdot \frac{s^n}{s^m}$$

However, remember that this necessitates the existence of a limit. Since $A(0) = B(0) = 1$ and $e^{-sL} \to 1$ as $s \to 0$ it holds that

$$e_\infty = \lim_{s \to 0} \frac{1}{s^n + K} s^{n-m}$$

We hence see that the stationary error is determined entirely by the low-frequency properties of the loop transfer function, i.e. K and n, as well as the property m of the setpoint. We obtain the following cases, depending on the relation between m and n:

$n > m$	$e_\infty = 0$
$n = m = 0$	$e_\infty = \frac{1}{1+K}$
$n = m \geq 1$	$e_\infty = \frac{1}{K}$
$n < m$	Limit does not exist.

The table shows that it is the number of integrators, n, in the loop transfer function, which determine how fast setpoints can be tracked without a persisting control error. In Example 7.1 the loop transfer function contained one integrator, i.e. $n = 1$. For step changes in setpoint we have $m = 0$ resulting in the stationary control error $e_\infty = 0$. For ramp changes in setpoint we have $m = 1$, which according to the above table results in $e_\infty = 1/K$. This is notably the same results which we arrived at in Example 7.1.

Stationary Errors—The Regulator Problem

We shall now investigate stationary errors in the context of the regulator problem, i.e. the case where the setpoint value is constant while the load disturbances vary. Since we assume the setpoint value to be $r = 0$, Equation (7.4) yields the measurement signal

$$Y(s) = \frac{G_P(s)}{1 + G_R(s)G_P(s)} L(s)$$

We can utilize the final value theorem in the same manner as before in order to determine the stationary error. Since $r = 0$, we might as well study the measurement signal directly rather than the control error. The final value theorem yields

$$y_\infty = \lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s)$$

We begin investigating the regulator problem by studying an example.

Example 7.2—Stationary errors in the regulator problem

Assume that the process and controller are given by the transfer functions

$$G_P = \frac{1}{1+sT} \qquad G_R = \frac{K}{s}$$

The process model can e.g. describe the relationship between current and angular speed in an electric motor. The controller is an I controller, i.e. a pure integrator. Together they yield the loop transfer function

$$G_0 = G_R G_P = \frac{K}{s(1+sT)}$$

which is identical to the loop transfer function studied in Example 7.1.

Assume that the load disturbance is given by a step, i.e.

$$l(t) = \begin{cases} 1 & t \geq 0 \\ 0 & t < 0 \end{cases}$$

with Laplace transformation $L(s) = 1/s$. The measurement signal then becomes

$$Y(s) = \frac{G_P(s)}{1 + G_R(s)G_P(s)} L(s) = \frac{s}{s(1+sT)+K} \cdot \frac{1}{s}$$

By means of the final value theorem we can compute the stationary measurement signal.

$$y_\infty = \lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = \lim_{s\to 0} \frac{s}{s(1+sT)+K} = 0$$

The limit exists under the assumption that the parameters K and T are positive. Just as in Example 7.1 we will not obtain any persisting control error.

Now assume that the process and controller are given by the transfer functions

$$G_P = \frac{1}{s(1+sT)} \qquad G_R = K$$

Compared to the previous case, we have now added an integrator to the process and removed an integrator from the controller. The loop transfer function is hence unaffected. The measurement signal at a step load disturbance becomes

$$Y(s) = \frac{G_P(s)}{1 + G_R(s)G_P(s)} L(s) = \frac{1}{s(1+sT)+K} \cdot \frac{1}{s}$$

By means of the final value theorem we obtain the stationary measurement signal

$$y_\infty = \lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = \lim_{s\to 0} \frac{1}{s(1+sT)+K} = \frac{1}{K}$$

Despite the fact that we have the same transfer function as previously, we still obtain a persisting control error. The integrator in the process does not contribute to eliminating the stationary error caused by the load disturbance. □

The example shows that the location of the integrator is of relevance. In the context of the servo problem it is the number of integrators in the loop transfer function, which determines which setpoints can be tracked without persisting control errors. In the context of the regulator problem it is the number of integrators in the *controller*, which determines which load disturbances can be eliminated by the feedback controller. We now show this by studying the general case.

The General Case Assume that the process and controller are given by the transfer functions

$$G_P(s) = \frac{K_P B_P(s)}{s^p A_P(s)} e^{-sL} \qquad G_R(s) = \frac{K B_R(s)}{s^r A_R(s)}$$

where $A_P(0) = B_P(0) = A_R(0) = B_R(0) = 1$. The load disturbances are given in the form

$$L(s) = \frac{1}{s^{m+1}}$$

The measurement signal of the general example becomes

$$Y(s) = \frac{G_P(s)}{1+G_R(s)G_P(s)} L(s) = \frac{s^r A_R K_P B_P e^{-sL}}{s^{r+p} A_P A_R + K B_R K_P B_P e^{-sL}} \cdot \frac{1}{s^{m+1}}$$

By means of the final value theorem we can now compute the stationary error.

$$y_\infty = \lim_{t\to\infty} y(t) = \lim_{s\to 0} sY(s) = \lim_{s\to 0} \frac{A_R K_P B_P e^{-sL}}{s^{r+p} A_P A_R + K B_R K_P B_P e^{-sL}} \cdot \frac{s^r}{s^m}$$

Remember that the limit must exist, in order for us to do this. Since $A_P(0) = B_P(0) = A_R(0) = B_R(0) = 1$ and $e^{-sL} \to 1$ as $s \to 0$ it holds that

$$y_\infty = \lim_{s\to 0} \frac{K_P}{s^{r+p} + K K_P} s^{r-m}$$

We thus see that the number of integrators, r, internal to the controller determines whether there will be any persisting control error. We obtain the following cases depending on the relation between m and r.

$r > m$	$y_\infty = 0$
$r = m = 0, \quad p = 0$	$y_\infty = \dfrac{K_P}{1+KK_P}$
$r = m = 0, \quad p \geq 1$	$y_\infty = \dfrac{1}{K}$
$r = m \geq 1$	$y_\infty = \dfrac{1}{K}$
$r < m$	The limit does not exist.

In Example 7.2 we studied steps in the load disturbances, i.e. $m = 0$. The first design example involved one integrator in the controller, i.e. $r = 1$. According to the above table this results in $y_\infty = 0$, which is confirmed by the example. In the second design example the controller lacked integrators. There was, however, an integrator in the process, i.e. $r = 0$ and $p = 1$. According to the table this yields the stationary error $y_\infty = 1/K$, which was also confirmed by the example.

The above conducted analysis shows why it is often desirable to have an integrator in the controller. The reason being that it guarantees the elimination of a stationary control error otherwise caused by steps in either setpoint or load disturbances.

Lecture 8

State Feedback

We shall now move on to the synthesis of control systems. This will be carried out in several of the model descriptions, which we introduced in the beginning of the course. We begin by designing controllers in state space.

8.1 State Feedback

We assume that the process to be controlled is given in state space form, i.e.

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{8.1}$$

For simplicity we assume that the process lacks a direct term, i.e. that the matrix $D = 0$. This is a realistic assumption since processes with a direct term are uncommon.

The transfer function of the process is given by

$$Y(s) = C(sI - A)^{-1}BU(s)$$

where the denominator polynomial

$$\det(sI - A)$$

is the characteristic polynomial of the process.

We further assume that we can measure all process states. This is obviously an unrealistic assumption in most cases. In the next chapter we will, however, see that this assumption can be relaxed and that the states can be computed from the only signals we normally posses over, i.e. the control and measurement signals. The controller structure is shown in Figure 8.1. The controller equation becomes

$$u = k_r r - k_1 x_1 - k_2 x_2 - \cdots - k_n x_n = k_r r - Kx \tag{8.2}$$

where the vectors K and x are given by

$$K = \begin{pmatrix} k_1 & k_2 & \cdots & k_n \end{pmatrix} \qquad x = \begin{pmatrix} x_1 \\ x_2 \\ \vdots \\ x_n \end{pmatrix}$$

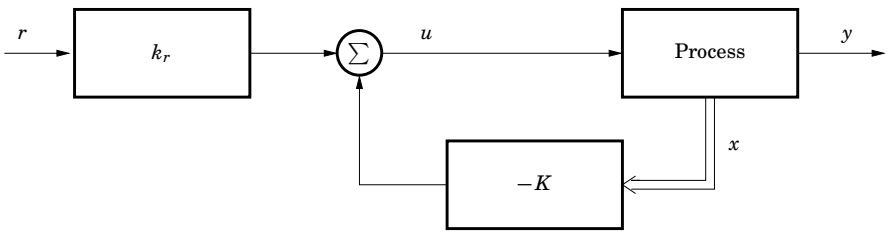

Figure 8.1 State feedback.

In state feedback the control signal is hence a linear combination of the state variables and the setpoint.

The Closed-Loop System If we combine the control law (8.2) with the process model (8.1) the state-space description of the closed-loop system is obtained.

$$\begin{aligned} \dot{x} &= (A - BK)x + Bk_r r \\ y &= Cx \end{aligned} \tag{8.3}$$

Here, the setpoint r is our new input. The corresponding transfer function is given by

$$Y(s) = C(sI - (A - BK))^{-1} Bk_r R(s)$$

where the characteristic polynomial has become

$$\det(sI - (A - BK))$$

The state feedback has changed the matrix A of the open-loop system (8.1) into $A - BK$, which is the corresponding matrix for the closed-loop system (8.3). Since K can be chosen arbitrarily, we have a certain freedom in determining the eigenvalues of this matrix.

The controller parameter k_r is apparently unaffected by the pole placement of the closed-loop system. To begin with we shall choose k_r such that the static gain of the closed loop system becomes unity in order to achieve $y = r$ in stationarity. Later we will introduce integral action to achieve this goal.

We will now illustrate the synthesis method by an example.

Example 8.1—State feedback control of an electric motor
The transfer function of an electric motor is given by

$$G_P(s) = \frac{100}{s(s + 10)}$$

where the current is the input and the shaft angle is the output. The transfer function can be divided into two parts according to Figure 8.2, where we have also marked the angular speed. If we introduce the angle and the angular speed as states, the process can be described in state-space form as

$$\begin{aligned} \dot{x}_1 &= -10x_1 + 100u \\ \dot{x}_2 &= x_1 \\ y &= x_2 \end{aligned}$$

which can be written in the matrix form

$$\dot{x} = \begin{pmatrix} -10 & 0 \\ 1 & 0 \end{pmatrix} x + \begin{pmatrix} 100 \\ 0 \end{pmatrix} u$$

$$y = \begin{pmatrix} 0 & 1 \end{pmatrix} x$$

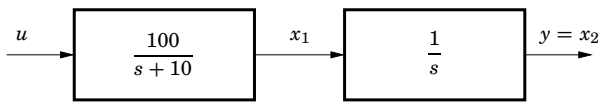

Figure 8.2 Block diagram of the motor in Example 8.1. The state x_1 corresponds to the angular speed, while the state x_2 represents the angle.

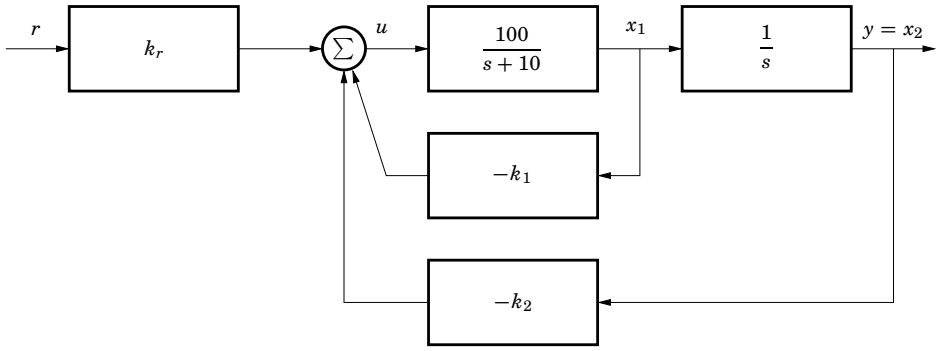

Figure 8.3 The state feedback motor control in Example 8.1.

Now we establish feedback connections from the angle and the angular speed according to Figure 8.3. The control law

$$u = k_r r - k_1 x_1 - k_2 x_2 = k_r r - Kx$$

yields the closed-loop system

$$\dot{x} = \begin{pmatrix} -10 - 100k_1 & -100k_2 \\ 1 & 0 \end{pmatrix} x + \begin{pmatrix} 100 \\ 0 \end{pmatrix} k_r r$$

$$y = \begin{pmatrix} 0 & 1 \end{pmatrix} x$$

The characteristic polynomial becomes

$$\begin{aligned} \det(sI - (A - BK)) &= \begin{vmatrix} s + 10 + 100k_1 & 100k_2 \\ -1 & s \end{vmatrix} \\ &= s^2 + (10 + 100k_1)s + 100k_2 \end{aligned}$$

Since we can achieve an arbitrary second-order characteristic polynomial by choosing the parameters k_1 and k_2, the poles of the closed-loop system can be places freely. Assume that the characteristic polynomial of the closed-loop system is given by

$$s^2 + 2\zeta\omega s + \omega^2$$

This yields the following controller parameters

$$k_1 = \frac{2\zeta\omega - 10}{100} \qquad k_2 = \frac{\omega^2}{100}$$

Now it remains to determine the parameter k_r such that $y = r$ in stationarity. One could always determine k_r by calculating the closed-loop transfer function and thereafter assure that $G(0) = 1$, i.e. that the static gain becomes unity. It is, however, often more efficient to investigate the stationary relation as $\dot{x} = 0$ directly in the state-space description. For our example it holds that

$$\dot{x} = 0 = \begin{pmatrix} -10 - 100k_1 & -100k_2 \\ 1 & 0 \end{pmatrix} x + \begin{pmatrix} 100 \\ 0 \end{pmatrix} k_r r$$

$$y = \begin{pmatrix} 0 & 1 \end{pmatrix} x$$

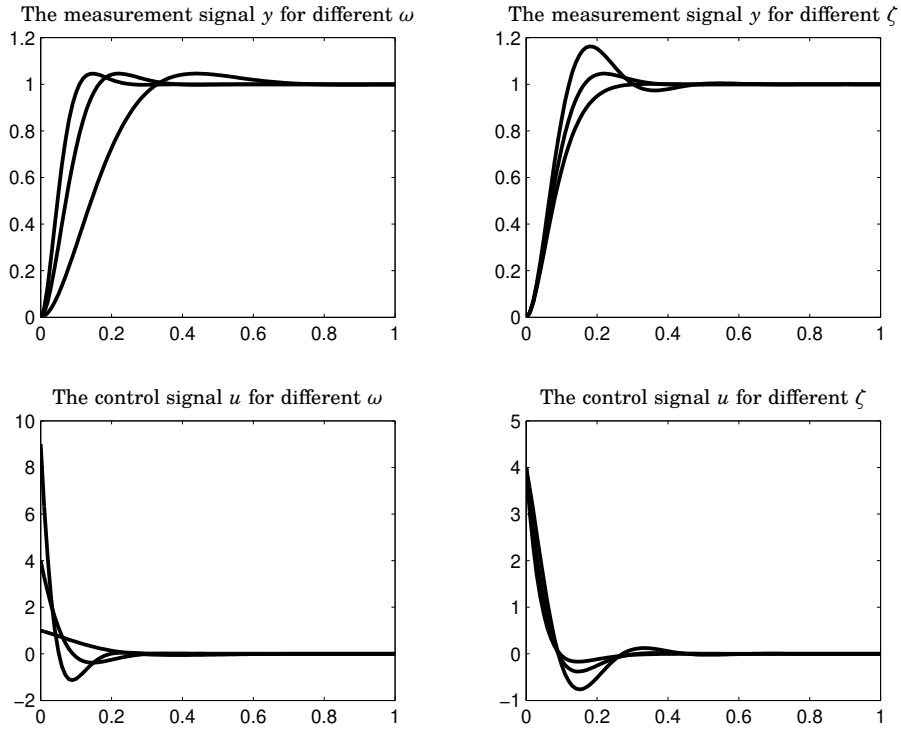

Figure 8.4 State feedback control of the motor in Example 8.1. The figures to the left show the controller corresponding to $\zeta = 0.7$ and $\omega = 10,\ 20,\ 30$, where the fastest control action is achieved by the highest frequency. The figures to the right show the control performance for $\omega = 20$ and $\zeta = 0.5,\ 0.7,\ 0.9$, where the most damped control action corresponds to the largest relative damping.

The second state-space equation implies that $x_1 = 0$ in stationarity. That this must be the case is obvious, since x_1 represents the angular speed. By inserting $x_1 = 0$ into the first equation and exploiting that $y = x_2$ we observe that $y = r$ if

$$k_r = k_2$$

We have now completed our synthesis. Figure 8.4 shows step responses corresponding to different choices of the design parameters ζ and ω. The figure shows that the design parameter ω is an efficient way to specify the speed of a system and that the relative damping ζ is a good measure of the damping of the system. □

8.2 Controllability

In the previous example we could place the poles of the closed-loop system arbitrarily by means of the state feedback. An interesting question is whether this is always possible. We begin the investigation by an example.

Example 8.2—Controllability
Assume that the process we are about to control is described by the following equations

$$\dot{x} = Ax + Bu = \begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix} x + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u$$

$$y = Cx + Du$$

Since the A matrix is diagonal, one can directly observe its eigenvalues and thereby the poles of the process, which lie in -1 and -2. State feedback yields the matrix

$$A - BK = \begin{pmatrix} -1 - k_1 & -k_2 \\ 0 & -2 \end{pmatrix}$$

This, in terms, gives us the characteristic polynomial of the closed-loop system

$$\det(sI - (A - BK)) = (s + 1 + k_1)(s + 2)$$

Here we see that we cannot place the poles arbitrarily. By means of k_1 we can move the pole which initially lays in -1. However, there is no way for us to move the pole in -2. We also see that the parameter k_2 does not appear in the characteristic polynomial. Hence we do not benefit from measuring x_2 and involve the measurement in the feedback.

The cause of our problem is clearly visible in the state-space description. Its second equation is given by

$$\dot{x}_2 = -2x_2$$

and is unaffected by the control signal u. Consequently, this state is not controllable. □

This reasoning leads us to the concept of controllability, which is defined in the following way:

> A state vector x_0 is said to be controllable if there exists a control signal which brings x from the origin to x_0 in a finite time. A system is controllable if all its states are controllable.

If a system is controllable it is possible to place its poles arbitrarily using state feedback. As seen from the definition, controllability is unrelated to the output y. The definition concerns the state vector and the control signal. In the state-space description (8.1) we thus see that the controllability is determined by the A and B matrices.

Whether a system is controllable can be determined by studying the controllability matrix, which is defined as

$$W_s = \begin{pmatrix} B & AB & A^2B & \cdots & A^{n-1}B \end{pmatrix}$$

where n is the degree of the system. One can show that a system is controllable if and only if the controllability matrix W_s consists of n linearly independent columns. In the non-controllable case the columns of W_s tell which, if any, of the states are controllable. This is illustrated in the examples below.

We investigate the controllability of the two systems studied in examples previously in this lecture.

Example 8.3—Controllability
In Example 8.1 an electric motor described by the equations

$$\dot{x} = \begin{pmatrix} -10 & 0 \\ 1 & 0 \end{pmatrix} x + \begin{pmatrix} 100 \\ 0 \end{pmatrix} u$$

$$y = \begin{pmatrix} 0 & 1 \end{pmatrix} x$$

was studied. The controllability matrix of this process is given by

$$W_s = \begin{pmatrix} B & AB \end{pmatrix} = \begin{pmatrix} 100 & -1000 \\ 0 & 100 \end{pmatrix}$$

The columns of this matrix are linearly independent. One way to determine whether the columns of a quadratic matrix are linearly independent is to investigate if the determinant is non-zero. Since the columns of W_s are linearly independent, the process is controllable. We observed this already in Example 8.1, since the poles could be arbitrarily placed.

Let us now investigate the controllability of the system in Example 8.2. There the A and B matrices were given by

$$A = \begin{pmatrix} -1 & 0 \\ 0 & -2 \end{pmatrix} \qquad B = \begin{pmatrix} 1 \\ 0 \end{pmatrix}$$

This yields the controllability matrix

$$W_s = \begin{pmatrix} B & AB \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ 0 & 0 \end{pmatrix}$$

The columns of this matrix are linearly dependent and $\det W_s = 0$. The system in Example 8.2 is thus not controllable. The columns of W_s also show that x_1 is a controllable state, while x_2 is not controllable. □

A Controllability Example

Controllability is an abstract concept. Processes are often constructed so that controllability is achieved for the states which one wishes to control. Despite this, the concept of controllability is still important and it is good to develop an intuitive understanding of it. We will return to this later in the course.

In order to increase the understanding of the controllability concept, we will now investigate a few physical processes regarding their controllability. The processes to be investigated are shown in Figure 8.5 and are made up by interconnected water tanks and the corresponding flows.

By writing down the mass balance equations for the individual tanks, we obtain a dynamical model. If the level in a tank is denoted x it holds that

$$\dot{x} = q_{in} - q_{out}$$

where q_{in} is the inflow and q_{out} the outflow from the tank. All tanks are equipped with a hole in their bottom which makes the inflow approximately proportional to the level in the corresponding tank. The inflow of a tank is made up by either of the outflow from a higher tank or an auxiliary flow u, which is our control signal.

Process A First consider process A in Figure 8.5. In this case the controlled flow enters the upper tank. The following mass balances are obtained

$$\dot{x}_1 = u - ax_1$$
$$\dot{x}_2 = ax_1 - ax_2$$

where x_1 and x_2 denote the levels in the upper and lower tank, respectively. The first equation tells us that the inflow of the upper tank is given by u whereas the outflow is proportional to the level in the tank. From the second equation we obtain that the inflow of the lower tank equals the outflow of the upper tank and that the outflow of the lower tank is proportional to the level in the tank. The dynamics of process A are thus described by

$$\dot{x} = Ax + Bu = \begin{pmatrix} -a & 0 \\ a & -a \end{pmatrix} x + \begin{pmatrix} 1 \\ 0 \end{pmatrix} u$$

As a consequence, the controllability matrix is given by

$$W_s = \begin{pmatrix} B & AB \end{pmatrix} = \begin{pmatrix} 1 & -a \\ 0 & a \end{pmatrix}$$

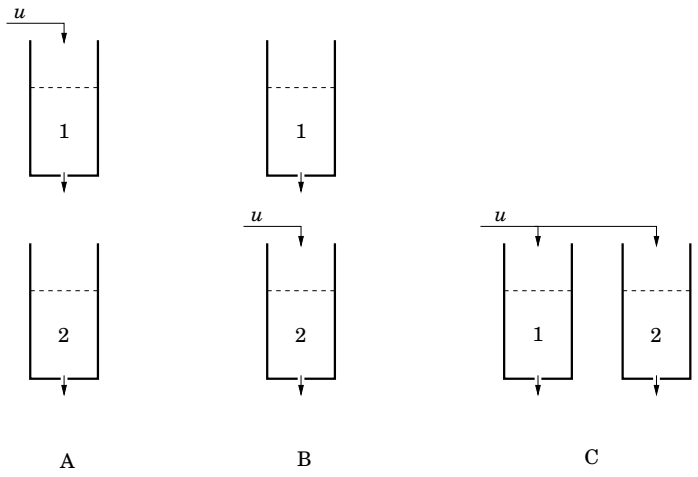

Figure 8.5 Several connections of water tanks leading to different cases regarding controllability.

The columns of this matrix are linearly independent, and the determinant is $\det W_s = a \neq 0$. Hence, the system is controllable. The columns of the controllability matrix are shown graphically in Figur 8.6, where it's obvious that they are linearly independent. This means that we can control both levels to take on arbitrary values, in accordance with the definition of controllability. Observe that the definition does not require that we should be able to maintain arbitrary constant levels. From studying the equations one realizes that this is only possible for levels which satisfy

$$u = ax_1 = ax_2$$

In stationarity we thus have the same level in both tanks. Finally we must consider the validity of the equations. In reality it holds that the levels cannot take on negative values, neither can the control signal be negative given that there is no pump which could suck water from the tank. As a consequence we have $x_1 \geq 0$, $x_2 \geq 0$ and $u \geq 0$.

Process B Now consider process B, shown in Figure 8.5. In this case the controlled flow enters the lower tank. This infers that we can no longer control the level in the upper tank. Let us now confirm this intuitively drawn conclusion by applying the above introduced analysis method.

The following balance equations constitute a model of process B

$$\dot{x}_1 = -ax_1$$
$$\dot{x}_2 = u + ax_1 - ax_2$$

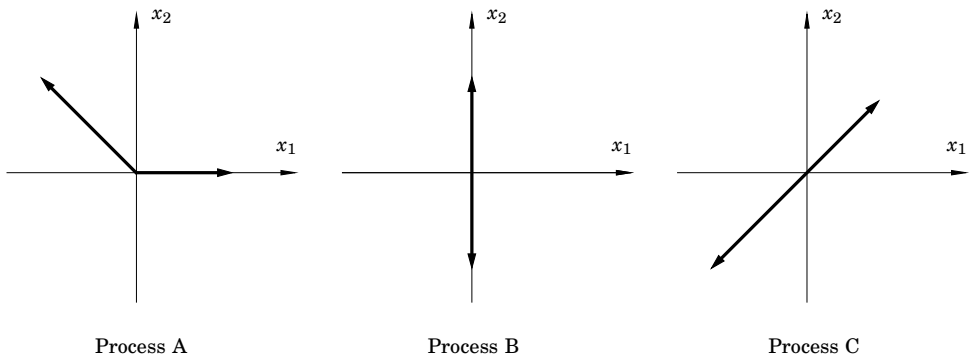

Figure 8.6 The columns of the controllability matrix in the three examples.

The process dynamics can therefore be written

$$\dot{x} = Ax + Bu = \begin{pmatrix} -a & 0 \\ a & -a \end{pmatrix} x + \begin{pmatrix} 0 \\ 1 \end{pmatrix} u$$

The controllability matrix is thus

$$W_s = \begin{pmatrix} B & AB \end{pmatrix} = \begin{pmatrix} 0 & 0 \\ 1 & -a \end{pmatrix}$$

The columns of the matrix are linearly dependent, the determinant is given by $\det W_s = 0$ and the system is hence not controllable. The columns of W_s further show that the level in the lower tank is controllable, as opposed to the upper tank level. This is also obvious from Figur 8.6.

Process C In process C, shown in Figure 8.5, the flow enters both tanks, such that half of the flow enters tank 1 whereas the other half enters tank 2.

The following balance equations hold for tank C

$$\dot{x}_1 = 0.5u - ax_1$$
$$\dot{x}_2 = 0.5u - ax_2$$

The dynamics of process C can therefore be written

$$\dot{x} = Ax + Bu = \begin{pmatrix} -a & 0 \\ 0 & -a \end{pmatrix} x + \begin{pmatrix} 0.5 \\ 0.5 \end{pmatrix} u$$

The controllability matrix becomes

$$W_s = \begin{pmatrix} B & AB \end{pmatrix} = \begin{pmatrix} 0.5 & -0.5a \\ 0.5 & -0.5a \end{pmatrix}$$

The columns of the matrix are linearly dependent, the determinant is $\det W_s = 0$ and the system is hence not controllable. The construction is such that if the levels are initially equal, they will remain equal. This can also be observed from the columns of W_s, and the illustration in Figur 8.6.

Lecture 9

Kalman Filtering

In the previous lecture the synthesis part of the course began by introducing state feedback. This method came with the price of several serious limitations, which we will eliminate during this lecture. Firstly, a premises of the state feedback is that we can measure all process states. This is, however, normally not possible. In this lecture we will describe the Kalman filter, which is a method to obtain an estimate of the state vector from the control and measurement signals.

Another serious limitation was that the previously introduced state feedback controller lacks integral action, which consequently may result in stationary control errors.

9.1 Integral Action

We begin by introducing integral action in the controller. Figure 8.1 shows the block diagram of the nominal state feedback. The parameter k_r was chosen such that the static gain from setpoint r to measurement signal y was unity. This did, however, not guarantee $y = r$ in stationarity. A load change will e.g. bring the measurement signal away from the setpoint.

Figure 9.1 shows how the state feedback can be augmented with an integral term. First we form the control error e, which we have not used in the context of state feedback previously, as the difference between the setpoint r and the measurement y. Then we integrate the control error and multiply it with a gain, which we in this context denote k_i. Finally we add the contribution from the integral term to the control signal u.

To add the integral term to the control signal in this way is no stranger than adding the integral term to a P controller and thereby arriving at a PI controller.

The integral term will naturally affect the position of the closed-loop poles. In order to use the same methodology as in the previous lecture we will describe the integral term as part of the state feedback.

We begin by reviewing the state-space representation of the process.

$$\begin{aligned} \dot{x} &= Ax + Bu \\ y &= Cx \end{aligned} \tag{9.1}$$

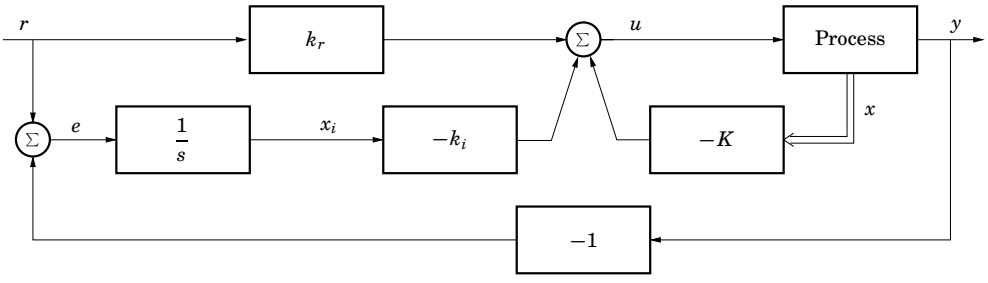

Figure 9.1 State feedback with integral action.

Now introduce the state x_1 according to Figure 9.1. This yields the equation

$$x_i = \int (r - y)dt \qquad \Rightarrow \qquad \dot{x}_i = r - y = r - Cx$$

If we augment the state vector x with the integral state x_i such that

$$x_e = \begin{pmatrix} x \\ x_i \end{pmatrix}$$

the augmented system can be written

$$\begin{aligned} \dot{x}_e &= \begin{pmatrix} \dot{x} \\ \dot{x}_i \end{pmatrix} = \begin{pmatrix} A & 0 \\ -C & 0 \end{pmatrix} x_e + \begin{pmatrix} B \\ 0 \end{pmatrix} u + \begin{pmatrix} 0 \\ 1 \end{pmatrix} r = A_e x_e + B_e u + B_r r \\ y &= \begin{pmatrix} C & 0 \end{pmatrix} x_e = C_e x_e \end{aligned} \tag{9.2}$$

Since we want to write the closed-loop system on state-space form, control signal u must be eliminated from Equation (9.2). The control signal is given by

$$u = k_r r - Kx - k_i x_i = k_r r - K_e x_e$$

where

$$K_e = \begin{pmatrix} K & k_i \end{pmatrix}$$

If this expression is introduced in Equation (9.2), the state-space form of the closed-loop system becomes:

$$\dot{x}_e = (A_e - B_e K_e)x_e + (B_e k_r + B_r)r$$

$$y = C_e x_e$$

We have hence augmented the state space system with a state which represents the integral of the control error and thus arrived at a controller with integral action. In stationarity it holds that $\dot{x}_e = 0$ and thereby that $\dot{x}_i = r - y = 0$.

The parameters in the vector K_e are chosen so that we obtain a desired closed-loop pole placement, just as previously. Here the poles are given by the characteristic polynomial

$$\det(sI - (A_e - B_e K_e))$$

We no longer need the parameter k_r in order to achieve $y = r$ in stationarity. The parameter does not affect the poles of the closed-loop system, only its zeros. It can therefore be chosen so that the system obtains desired transient properties at setpoint changes. We shall come back to zero placement in a later lecture.

9.2 Observability

So far we have assumed that all process states have been directly measurable. Normally this is not the case. However, if one can estimate the state vector by studying the control and measurement signals, one could close the loop over the estimates rather than the real states.

In Example 8.1 we utilized state feedback from two states in order to control an electric motor. The measurement signal was the angle of the motor shaft and the other state was given by the angular speed of the shaft. If the angular speed cannot be directly measured, it can be estimated by e.g. deriving the angle measurement.

Before describing how to estimate the state vector, we will ask ourselves the principal question whether it is generally possible to estimate the state vector merely by studying u and y. The answer is that it is possible if the system is *observable*. Observability is defined in the following way:

> A state vector $x_0 \neq 0$ is not observable if the output is $y(t) = 0$ when the initial state vector is $x(0) = x_0$ and the input is given by $u(t) = 0$. A system is observable if it lacks non-observable states.

As seen from the definition, observability has nothing to do with the control signal u. Rather, it concerns the state vector and the measurement signal. In the state-space description (9.1) we see that this implies that observability is determined solely by A and C.

Whether a system is observable can be determined by studying the observability matrix which is defined in as

$$W_o = \begin{pmatrix} C \\ CA \\ CA^2 \\ \vdots \\ CA^{n-1} \end{pmatrix}$$

Here n is the degree of the system. One can show that the system is observable if and only if the observability matrix W_o has n linearly independent rows. If x_0 is a non-observable state, it fulfills the equation

$$W_o x_0 = 0$$

As seen, observability and the observability matrix have strong resemblance to controllability and the controllability matrix, which we studied in the previous lecture. We will nu study an observability example.

Example 9.1—Observability
A process is described by the following equations:

$$\dot{x} = \begin{pmatrix} -1 & 0 \\ 0 & -1 \end{pmatrix} x$$

$$y = \begin{pmatrix} 1 & -1 \end{pmatrix} x$$

We can e.g. imagine that the equations describe the process in Figure 9.2, where the states are given by the tank levels. The measurement signal is the difference between the tank levels. The system lacks input and we study only the state evolution after various initial states.

We shall now investigate the observability of the process. The observability matrix is

$$W_o = \begin{pmatrix} C \\ CA \end{pmatrix} = \begin{pmatrix} 1 & -1 \\ -1 & 1 \end{pmatrix}$$

The rows of this matrix are not linearly independent, the determinant is $\det W_o = 0$ and the system is not observable. From the equation

$$W_o x_0 = 0$$

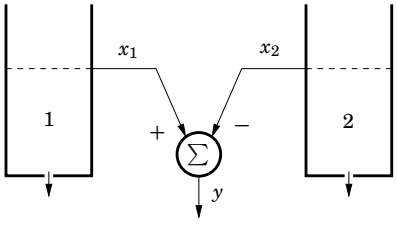

Figure 9.2 A physical interpretation of the process dynamics in Example 9.1.

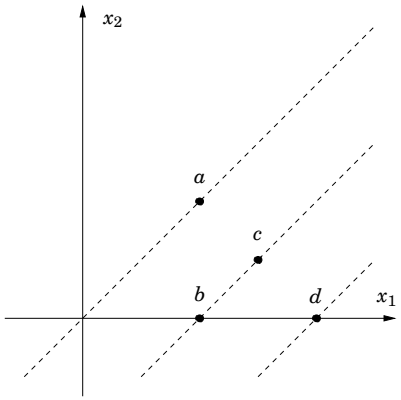

Figure 9.3 Different initial states in Example 9.1.

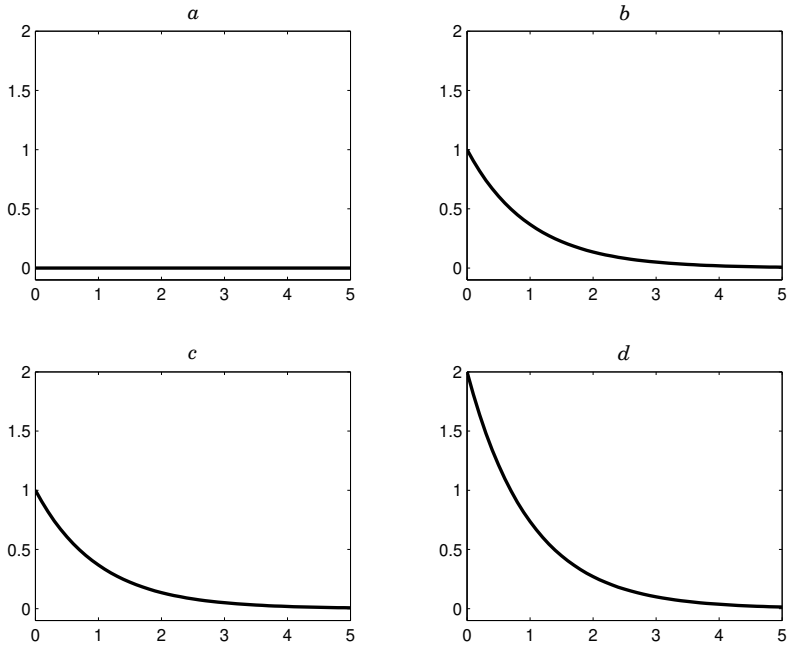

Figure 9.4 The measurement y corresponding to the initial states in Figure 9.3.

we see that the non-observable states can be written

$$x_0 = \begin{pmatrix} a \\ a \end{pmatrix}$$

If we return to the definition of observability we can justify that this is correct. If the initial levels in the two tanks are equal they will empty in exactly the same way and the measurement remains $y = 0$.

Figure 9.3 shows four initial states for the levels in the two tanks. The corresponding outputs y are shown in Figure 9.4

The initial state a is a non-observable state. The measurement is consequently $y = 0$. The initial state b yields a response in the measurement. However, this is the same response obtained for the initial state c. All initial states which have the same difference between the levels x_1 and x_2 will yield the same response in the measurement signal y and it is hence not possible to tell them apart. The initial state d yields a response which differs from the others since the difference between x_1 and x_2 is not the same as for the other three cases. □

9.3 Kalman Filtering

If the system is observable it is possible to estimate the state vector by studying the input u and the output y. The most common way of doing this is to filter the in- and outputs through a Kalman filter. We shall deduce the Kalman filter in two steps. As a first step we shall see what happens if we try to estimate x by simply simulating the process.

State Estimation through Simulation

The prerequisite of state estimation is that the process is known and given in state space form and that the control signal u as well as the measurement signal y are available. We can simulate the process in the following way:

$$\dot{\hat{x}} = A\hat{x} + Bu$$

Here $\hat{x}$ is the estimate of the state vector x. We drive the simulation with the same control signal u as we apply to the real process. Will this work? Will the estimated state vector $\hat{x}$ converge to x?

Introduce the estimation error

$$\tilde{x} = x - \hat{x}$$

The derivative of the estimation error becomes

$$\dot{\tilde{x}} = \dot{x} - \dot{\hat{x}} = Ax + Bu - A\hat{x} - Bu = A\tilde{x}$$

Whether the estimation error approaches zero depends on the matrix A. If A has all its eigenvalues in the left half plane, i.e. if the process is asymptotically stable, the estimation error will converge to zero. The convergence rate depends on where in the left half plane the eigenvalues lie. Estimating the state vector by simulating the process might thus work in some cases. For the estimation we use the control signal u, while we ignore the information available through the measurement y. State estimation by derivation of y has a corresponding shortcoming. If one for instance estimates the angular speed of the motor in Example 8.1 by deriving the angle measurement, the information in the control signal is not exploited. The Kalman filter is, however, a method which makes use of both the control and measurement signals in order to estimate the state vector.

Kalman Filtering

The Kalman filter estimates the state vector in the following way:

$$\begin{aligned} \dot{\hat{x}} &= A\hat{x} + Bu + L(y - \hat{y}) \\ \hat{y} &= C\hat{x} \end{aligned} \tag{9.3}$$

Compared to the simple simulation method, we have now introduced a correction term. Here the difference between the real measurement signal y and the estimated measurement signal $\hat{y}$ affects the estimation.

By merging the two equations in Equation (9.3) the Kalman filter can be written in the form

$$\dot{\hat{x}} = (A - LC)\hat{x} + Bu + Ly \tag{9.4}$$

From this form it is obvious how the Kalman filter is driven by the two signals u and y.

The estimation error $\tilde{x}$ decreases according to

$$\begin{aligned} \dot{\tilde{x}} = \dot{x} - \dot{\hat{x}} &= Ax + Bu - A\hat{x} - Bu - LC(x - \hat{x}) \\ &= (A - LC)\tilde{x} \end{aligned}$$

The properties of the Kalman filter are no longer determined merely by the A matrix, but rather by the $A - LC$ matrix, in which the vector L is a free parameter. We can consequently choose a desired decrease rate of the estimation error $\tilde{x}$. The choice is a compromise between speed and sensitivity towards disturbances and modelling errors. A method used to determine L is illustrated by the following example.

Example 9.2—Kalman filtering of pendulum dynamics

In this example we shall balance an inverted pendulum. We leave the control part to the next lecture and put our focus on process state estimation. The pendulum to be controlled is shown in Figure 9.5.

We assume that we can measure the angle φ which consequently becomes our measurement signal, i.e. $y = \varphi$. The control objective is to balance the pendulum so that $y = \varphi = 0$. The setpoint will thus be $r = 0$. The control signal u is proportional to the force affecting the cart. For simplicity we assume that $u = \ddot{z}$.

There are two forces affecting the pendulum. The gravitational force strives to increase the angle φ and thus causing the pendulum to fall down. The control signal can give the pendulum an acceleration in the opposite direction and thus re-erect the pendulum.

A momentum equation with suitable choice of states, linearization and parameters yields the following process model of the pendulum:

$$\ddot{\varphi} = \varphi - u$$

A second-order system requires at least two states for its state-space representation. We choose the angle φ and its derivative $\dot{\varphi}$ to represent these states.

$$x_1 = \varphi$$

$$x_2 = \dot{\varphi}$$

This yields the following state-space description of the process:

$$\dot{x} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ -1 \end{pmatrix} u = Ax + Bu$$

$$y = \begin{pmatrix} 1 & 0 \end{pmatrix} x = Cx$$

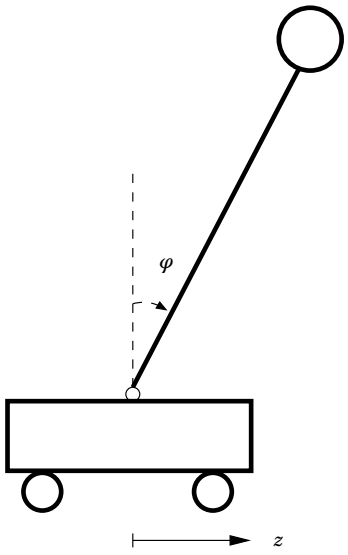

Figure 9.5 The pendulum in Example 9.2.

A Kalman filter estimating the state vector is given by Equation (9.4) and our task is to determine the vector L so that the matrix $A - LC$ obtains the desired eigenvalues. We have

$$A - LC = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} - \begin{pmatrix} l_1 \\ l_2 \end{pmatrix} \begin{pmatrix} 1 & 0 \end{pmatrix} = \begin{pmatrix} -l_1 & 1 \\ 1 - l_2 & 0 \end{pmatrix}$$

The eigenvalues are given by the characteristic polynomial

$$\det(sI - (A - LC)) = \begin{vmatrix} s + l_1 & -1 \\ -1 + l_2 & s \end{vmatrix} = s^2 + l_1 s - 1 + l_2$$

We can now place the two poles arbitrarily by choosing l_1 and l_2 adequately. E.g. assume that we desire to place the poles in $s = -4 \pm 4i$. This yields the desired characteristic polynomial

$$(s + 4 - 4i)(s + 4 + 4i) = s^2 + 8s + 32$$

By comparing the two polynomials, we arrive at the following Kalman filter parameters:

$$l_1 = 8 \qquad l_2 = 33$$

Figure 9.6 shows the result of a simulation experiment with the Kalman filter. The initial state is given by $\varphi(0) = -0.6$ and $\dot{\varphi}(0) = 0.4$ Since the process is unstable, it has to be controlled. This control is treated in the next lecture.

The solid curves in Figure 9.6 show the actual states. The angle is initially $\varphi(0) = -0.6$. After an initial overshoot to $\varphi \approx 0.4$ it stabilizes at $\varphi = 0$ after approximately 3 s.

The dashed curves show the estimates and are thus the ones of highest interest for us. They both start out in zero. After approximately 1.5 s they have converged well to the actual states. □

The example shows that the Kalman filter can be used to estimate the state vector. One thus realizes that it should be possible to use the estimated state for the

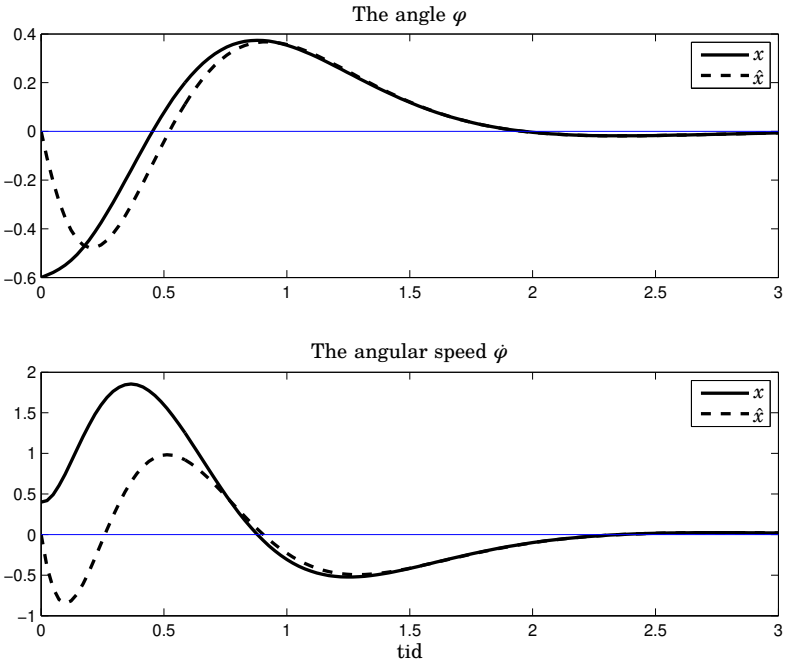

Figure 9.6 Actual and estimated angle and angular speed of the pendulum in Example 9.2.

feedback, rather than the actual ones. It is, however, important that the Kalman filter is sufficiently fast, since the transient behavior shown in Figure 9.6 will be repeated every time the process is disturbed, which would happen e.g. if someone pokes at the pendulum. The relationship between the Kalman filter and the state feedback will be further investigated in the next lecture.

Finally it is worthwhile noting that the Kalman filter is not merely interesting in the context of state feedback control. The method to estimate variables which are not directly measurable by exploiting the process dynamics and the availability of certain measurement signals is used in a large number of other applications. Examples include technical systems as well as biological, medical and economical ones.

Lecture 10

Output Feedback and Pole-Zero Cancellation

Two lectures ago we introduced the concept of output feedback and assumed that all process states were available for measurement. We chose the controller parameters with the placement of the closed-loop system poles in mind. In the previous lecture the Kalman filter was introduced. It turned out to be an attractive way of estimating the state vector and we chose the filter parameters so that a desired filter pole placement was obtained. In this lecture we will investigate what happens if we apply state feedback control to a Kalman filtered process.

The cancellation of poles and zeros in the transfer function leads to the loss of controllability and/or observability. This will be further explored in the second part of this lecture.

10.1 Output Feedback

We shall now merge the state feedback and the Kalman filter and close the loop from the estimated states, rather than the actual ones. We call this output feedback in order to emphasize that we do no longer close the loop from the states of the process. Rather, we use the setpoint r, measurement y and control signal u. The controller structure is shown in Figure 10.1, where the controller is confined within the dashed rectangle.

In order to simplify the analysis we have chosen to omit the integral term, which was introduced in the previous lecture.

We choose to describe the process in state space and assume that it lacks a direct

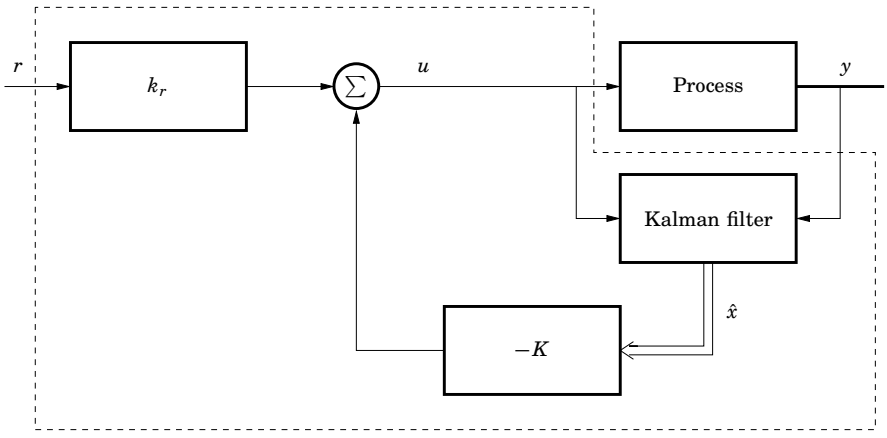

Figure 10.1 Output feedback. The controller is confined within the dashed rectangle.

term.

$$\dot{x} = Ax + Bu$$
$$y = Cx$$

The controller now becomes a combination of the Kalman filter and the state feedback

$$\dot{\hat{x}} = A\hat{x} + Bu + L(y - \hat{y})$$
$$\hat{y} = C\hat{x}$$
$$u = k_r r - K\hat{x}$$

We shall investigate the closed-loop system and begin by doing so in state space. A natural way to choose the state vector would have been to append the Kalman filter estimates $\hat{x}$ to the process states x. For reasons, which will become evident later in the lecture, we will rather append the process states x with the estimation error $\tilde{x} = x - \hat{x}$ in order to form the state vector.

$$x_e = \begin{pmatrix} x \\ \tilde{x} \end{pmatrix}$$

The state-space equations can now be written

$$\begin{aligned} \dot{x} &= Ax + Bu = Ax + Bk_r r - BK\hat{x} = Ax + Bk_r r - BK(x - \tilde{x}) \\ &= (A - BK)x + BK\tilde{x} + Bk_r r \\ \dot{\tilde{x}} &= \dot{x} - \dot{\hat{x}} = Ax + Bu - A\hat{x} - Bu - LC(x - \hat{x}) = (A - LC)\tilde{x} \end{aligned}$$

On block matrix form this becomes

$$\begin{aligned} \begin{pmatrix} \dot{x} \\ \dot{\tilde{x}} \end{pmatrix} &= \begin{pmatrix} A - BK & BK \\ 0 & A - LC \end{pmatrix} \begin{pmatrix} x \\ \tilde{x} \end{pmatrix} + \begin{pmatrix} Bk_r \\ 0 \end{pmatrix} r = A_e \begin{pmatrix} x \\ \tilde{x} \end{pmatrix} + B_e r \\ y &= \begin{pmatrix} C & 0 \end{pmatrix} \begin{pmatrix} x \\ \tilde{x} \end{pmatrix} = C_e \begin{pmatrix} x \\ \tilde{x} \end{pmatrix} \end{aligned} \tag{10.1}$$

Thanks to the introduction of x and $\tilde{x}$ as system states, the A_e, B_e and C_e matrices now contain a number of zeros. We will benefit from this substantially now that we will investigate the closed-loop system.

Due to the block triangularity of A_e, its characteristic polynomial is given by

$$\det(sI - A_e) = \det(sI - (A - BK)) \cdot \det(sI - (A - LC))$$

This is an appealing result. It shows that the characteristic polynomial is a product of the characteristic polynomial from the nominal state feedback and that of the Kalman filter. Consequently it is possible to separate the control problem into two parts, as we have already attempted in the previous lectures. The state feedback can first be determined and its poles placed as if all states were actually measurable. When later a Kalman filter is introduced in order to obtain state estimates and realize output feedback, this will not affect the pole placement of the feedback.

Let us compute the transfer function from r to y. It is given by

$$G_e(s) = C_e(sI - A_e)^{-1}B_e = \begin{pmatrix} C & 0 \end{pmatrix} \begin{pmatrix} E \\ F \end{pmatrix}$$

where

$$\begin{pmatrix} E \\ F \end{pmatrix} = \begin{pmatrix} sI - (A - BK) & -BK \\ 0 & sI - (A - LC) \end{pmatrix}^{-1} \begin{pmatrix} Bk_r \\ 0 \end{pmatrix}$$

By multiplying both sides with $(sI - A_e)$ we get

$$\begin{pmatrix} Bk_r \\ 0 \end{pmatrix} = \begin{pmatrix} sI - (A - BK) & -BK \\ 0 & sI - (A - LC) \end{pmatrix} \begin{pmatrix} E \\ F \end{pmatrix}$$
$$= \begin{pmatrix} (sI - (A - BK))E - BKF \\ (sI - (A - LC))F \end{pmatrix}$$

The latter part of the equation yields $F = 0$. Consequently, we arrive at

$$E = (sI - (A - BK))^{-1} Bk_r$$

which gives us the transfer function

$$G_e(s) = C(sI - (A - BK))^{-1} Bk_r$$

This is a remarkable result. The transfer function is identical to the one which we obtained when we closed the loop from the real states. The dynamics of the Kalman filter are hidden in the transfer function. The characteristic polynomial of the A_e matrix is of order $2n$, whereas the transfer function is merely of order n.

In the pendulum example from the previous lecture we saw how the estimated state converged to the actual state after a few seconds long transient. Following this transient there is inherently no difference between feedback from the estimated and actual states, as long as no disturbances forces the state estimate away from the actual state. This explains why the dynamics of the Kalman filter are not visible in the transfer function.

A transfer function of order n can always be described in state space using n states, which are all both controllable and observable. One could also introduce more than n states, but the additional states cannot be both observable and controllable. It is e.g. possible to introduce a state which has no significant relation to the process. This state will obviously be neither controllable nor observable.

Correspondingly a transfer function of degree n obtained from a state space model of degree $> n$ shows us that there are states which lack controllability or observability. In our case the Kalman filter state is not controllable. This can be seen by constructing the controllability matrix of the system in Equation (10.1).

$$W_s = \begin{pmatrix} B_e & A_eB_e & \cdots & A_e^{n-1}B_e \end{pmatrix} = \begin{pmatrix} Bk_r & (A - BK)Bk_r & \cdots \\ 0 & 0 & \cdots \end{pmatrix}$$

Since the last n elements in the columns are zero, there are n linearly independent columns and the non-controllable states correspond to those of the Kalman filter. The fact that the states of the Kalman filter are not controllable can be seen directly from the state space Equation (10.1).

$$\dot{\tilde{x}} = (A - LC)\tilde{x}$$

We can apparently not affect the estimates by means of the closed-loop system input, i.e. by the setpoint r.

Summary

We have now introduced a method to determine a controller—state feedback combined with Kalman filtering. The analysis has shown that the design process can be separated into two parts.

The first part was to estimate the state vector by means of the Kalman filter. The properties of the Kalman filter are determined by the gain vector L. The choice of this vector, i.e. the pole placement of the Kalman filter, is as always a balance between performance and robustness against model errors, process variations and disturbances.

The second part was to design the state feedback. Here the analysis has shown us that we can treat the problem as if working with the actual states. The feedback vector K determines the poles of the closed-loop system. Also here the choice is a compromise between performance and robustness. In order to keep the estimate close to the actual state, it is customary to choose K so that the poles of the state feedback are at least twice as slow as those of the Kalman filter. This is, however, not a requirement and there exist exceptions from this rule of thumb.

We will now round off by completing the pendulum example, which we started in the previous lecture.

Example 10.1—Inverted pendulum control
In Example 9.2 and Figure 9.5 the inverted pendulum was described and a Kalman filter for estimating its states

$$x_1 = \varphi$$
$$x_2 = \dot{\varphi}$$

was established for the following state space model of the process:

$$\dot{x} = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} x + \begin{pmatrix} 0 \\ -1 \end{pmatrix} u = Ax + Bu$$

$$y = \begin{pmatrix} 1 & 0 \end{pmatrix} x = Cx$$

We will now initially assume that the states are directly measurable and hence introduce the state feedback

$$u = -Kx$$

Since the setpoint is constantly $r = 0$ there is no reason to include the term $k_r r$ in the controller. Neither do we bother to introduce integral action. The closed loop system is given by

$$\dot{x} = (A - BK)x$$
$$y = Cx$$

where

$$A - BK = \begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} - \begin{pmatrix} 0 \\ -1 \end{pmatrix} \begin{pmatrix} k_1 & k_2 \end{pmatrix} = \begin{pmatrix} 0 & 1 \\ 1 + k_1 & k_2 \end{pmatrix}$$

The eigenvalues of the matrix are given by the characteristic polynomial

$$\det(sI - (A - BK)) = \begin{vmatrix} s & -1 \\ -1 - k_1 & s - k_2 \end{vmatrix} = s^2 - k_2 s - 1 - k_1$$

The two poles can now be placed arbitrarily by choosing k_1 and k_2 adequately. In Example 9.2 the poles of the Kalman filter were placed in $s = -4 \pm 4i$. It is reasonable to choose the poles for the state feedback closer to the origin. Here we will choose two poles, with half the distance to the origin, i.e. in $s = -2 \pm 2i$. This yields the desired characteristic polynomial

$$(s + 2 - 2i)(s + 2 + 2i) = s^2 + 4s + 8$$

By matching coefficients the following controller parameters are obtained.

$$k_1 = -9 \qquad k_2 = -4$$

Figure 10.2 shows the result of a simulation where feedback has been established from the actual states, using these controller parameters. The figure shows how the

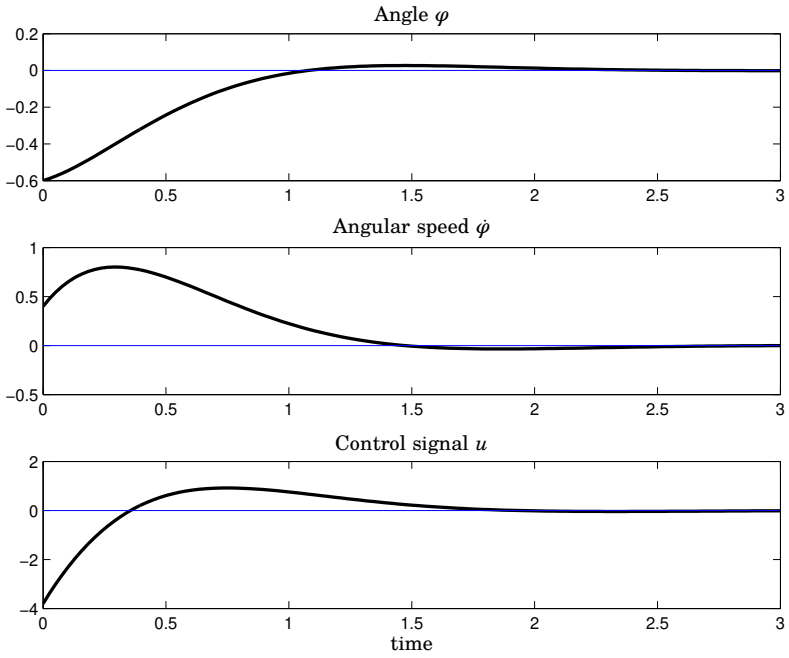

Figure 10.2 State feedback from actual states in Example 10.1.

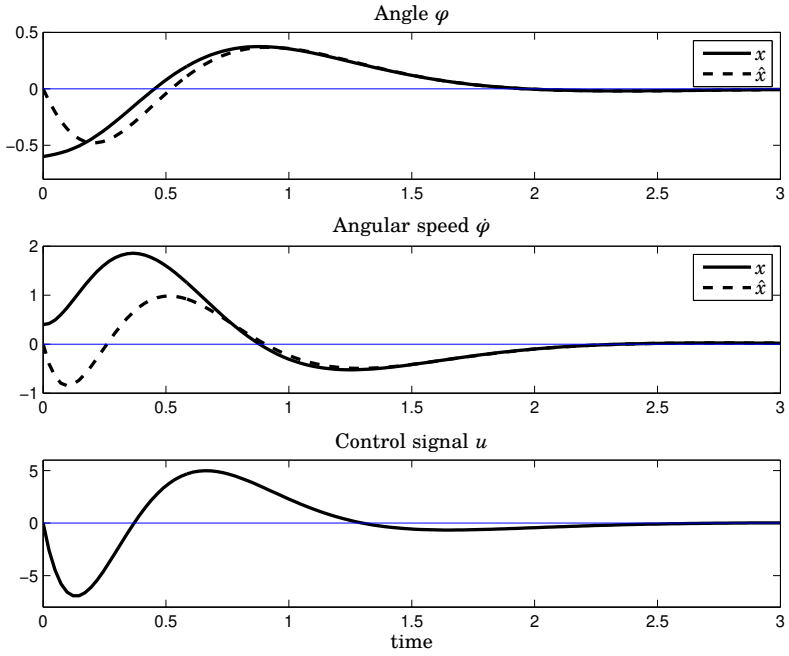

Figure 10.3 State feedback from estimated states in Example 10.1.

controller erects the pendulum from the initial state $\varphi = -0.6$, $\dot{\varphi} = 0.4$ to the desired state $\varphi = \dot{\varphi} = 0$.

Figure 10.3 shows the result when feedback was establish from the estimated states, which were obtained by the Kalman filter from Example 9.2. The figure clearly shows how control performance has decreased in the initial phase due to the poor initial estimates produced by the Kalman filter. □

10.2 Cancellation of Poles and Zeros

In connection to the state feedback and Kalman filtering we have discussed the concepts of controllability and observability. We have seen that the loss of controllability and observability is connected to the cancellation of poles and zeros in the transfer function, resulting in a lower order of the transfer function, than the dimension of the A matrix in the state-space representation.

We have previously pointed out that most processes are inherently designed to be both controllable and observable. However, the insight that controllability and observability can be lost through pole/-zero cancellation is important. It is indeed not uncommon that these properties are lost by introducing a controller which cancels part of the process dynamics. We shall illustrate the problem through a simple and very commonly occurring example.

Assume that the controlled process is described by the simple first-order model

$$G_P(s) = \frac{1}{1+sT}$$

and that it is controlled by a PI controller with transfer function

$$G_R(s) = K\left(1 + \frac{1}{sT_i}\right) = K\frac{1+sT_i}{sT_i}$$

where K is the controller gain and T_i its integral time. The system is shown in Figure 10.4.

Many well known tuning methods for PI control are based on choosing the integral time to correspond to the (slowest) time constant of the process, i.e.

$$T_i = T$$

Figure 10.4 shows that this results in the cancellation of the process pole and the controller zero. Consequently, a simplified description of the system is obtained, shown in Figure 10.5. The open loop transfer function is now given by

$$G_0(s) = G_R(s)G_P(s) = K\frac{1+sT}{sT} \cdot \frac{1}{1+sT} = \frac{K}{sT}$$

We have thus obtained a first order loop transfer function. The closed-loop transfer function from r to y becomes

$$G(s) = \frac{G_0(s)}{1+G_0(s)} = \frac{K}{K+sT}$$

This first order system can be made arbitrarily fast by choosing K so that the pole

$$s = -\frac{K}{T}$$

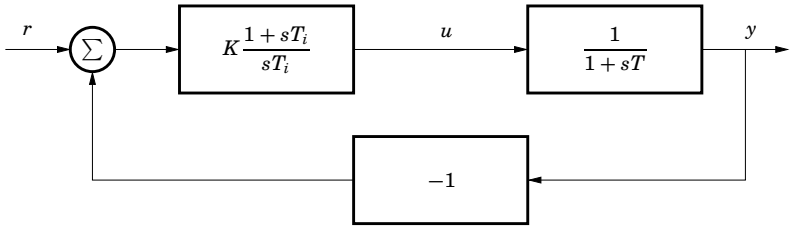

Figure 10.4 The simple feedback loop.

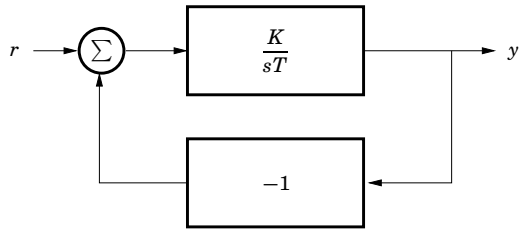

Figure 10.5 The simple feedback loop in figure 10.4 after the cancellation.

is adequately placed.

Figure 10.6 shows the result of a simulation where the time constant of the process was $T = 10$, resulting in $T_i = T = 10$. The controller gain was chosen to be $K = 10$, yielding the closed-loop time constant $T_s = 1$, i.e. 10 times faster than that of the open-loop system.

Thus far everything seems to be in order. However, the fact that poles and zeros have been cancelled should raise some suspicion. It means that there exists states which are not controllable or observable.

Let us now investigate what happens when load disturbances are introduced according to the block diagram in Figure 10.7. We now have two inputs—the setpoint r and the load disturbance l. This yields the following transfer functions:

$$\begin{aligned} Y(s) &= G_P(s)\left(L(s) + G_R(s)(R(s) - Y(s))\right) \\ &= \frac{G_P(s)G_R(s)}{1 + G_P(s)G_R(s)}R(s) + \frac{G_P(s)}{1 + G_P(s)G_R(s)}L(s) \\ &= \frac{K}{K + sT}R(s) + \frac{sT}{(1 + sT)(K + sT)}L(s) \end{aligned}$$

We observe that the transfer function from l to y is of second order and that the initial process pole is part of its dynamics. The state corresponding to this slow pole has been made uncontrollable from r by the pole/zero cancellation. However, the

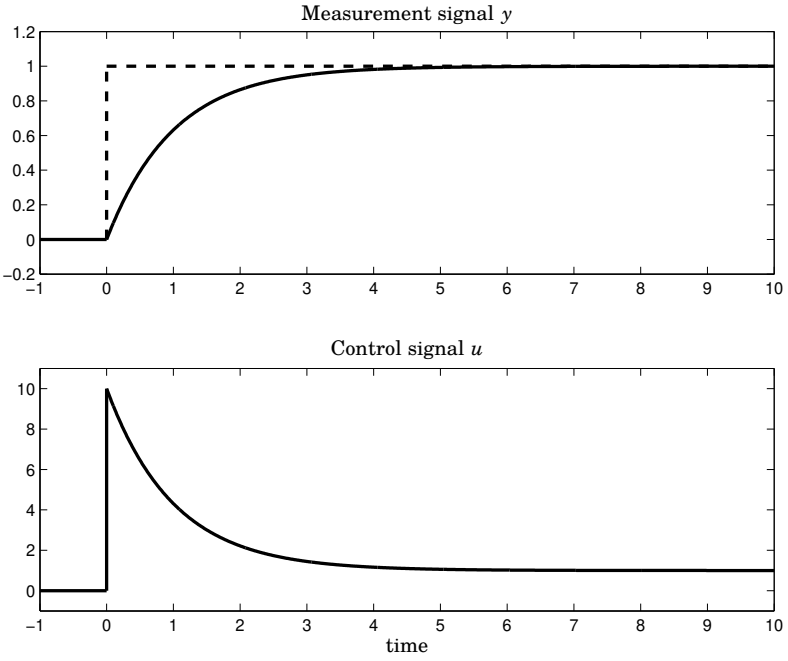

Figure 10.6 A setpoint step response. The time constant of the closed loop system is $T_s = 1$.

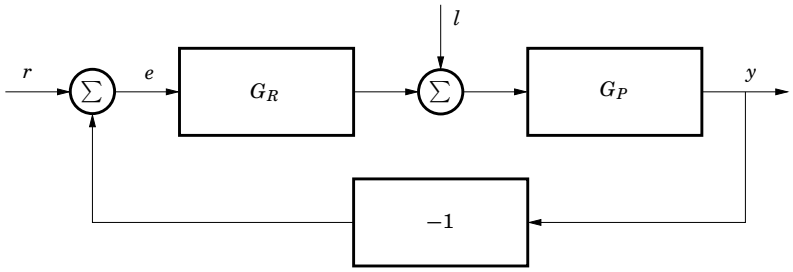

Figure 10.7 The simple feedback loop with both a setpoint r and load disturbances l.

state is observable which will be revealed upon load changes.

Figure 10.8 shows the response of a setpoint step followed by a load step. Note that the scales here are not the same as in Figure 10.6. The figure shows that the load response becomes significantly slower due to the presence of the open-loop pole, which has the time constant $T = 10$. Observe that the control signal is affected by the fast time constant but remains constant during the slow settling process.

This example shows the potential danger in cancelling poles and zeros. The cancellation can lead to simple calculations and good properties concerning variations in certain signals. However, the lack of controllability and/or observability caused by the cancellation can lead to unpleasant surprises when other signals are varied. This is especially true when slow dynamics are cancelled.

The conclusion is that it is safer to place transfer function poles adequately, so that they do not cancel any possible zeros of the system.

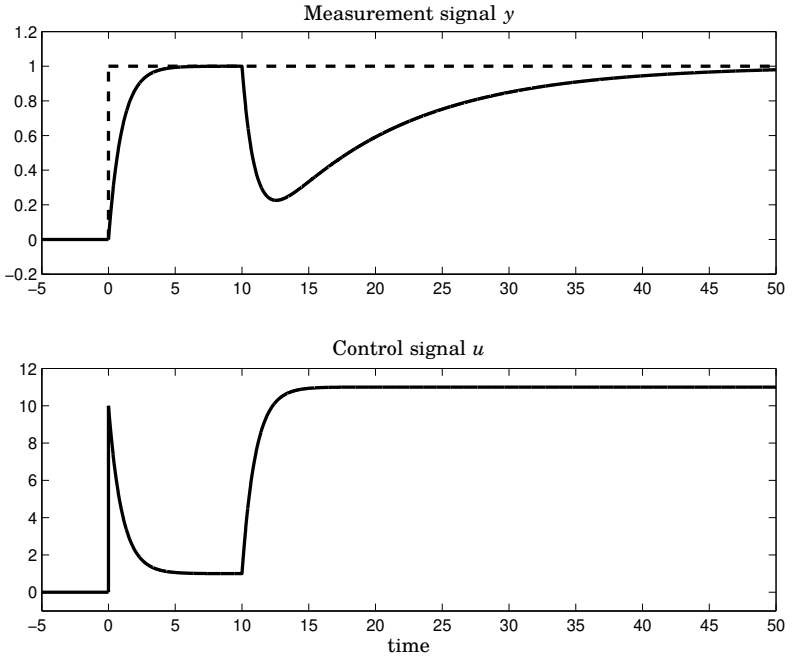

Figure 10.8 Response to a step in r at $t = 0$ followed by a step in l at $t = 10$.

Lecture 11

Lead–Lag Compensation

We shall now move on to determining controllers using frequency domain methods, aided by the Bode plot. First we will review how to determine the properties of the closed-loop system by imposing specifications on the Bode plot of the open-loop system. Subsequently, two types of compensation links will be introduced. One of them affects the low frequency properties and the stationary error, whereas the other affects the speed and robustness of the system.

11.1 Specifications in the Bode Plot

Figure 11.1 shows the Bode plot of a simple feedback loop with loop transfer function $G_0 = G_P G_R$, where G_P is the transfer function of the process and G_R that of the controller. The closed-loop transfer function is given by

$$G(s) = \frac{G_0(s)}{1 + G_0(s)}$$

Since one wants the measurement signal to track the setpoint, i.e. $y = r$, the ideal closed loop transfer function would be $G(s) = 1$. This is obviously not practically achievable. However, with a high open-loop gain $|G_0(i\omega)|$ at low frequencies it is possible to achieve $G(i\omega) \approx 1$ for low frequencies. Figure 11.2 shows a typical Bode plot of a closed-loop transfer function. We have $|G(i\omega)| \approx 1$ for low frequencies. The magnitude then increases for a frequency range and undergoes a fast decline for even higher frequencies.

The bandwidth ω_b is frequently used as a measure of system speed. It is defined as the frequency for which $|G(i\omega_b)| = 1/\sqrt{2}$, see Figure 11.2.

Figure 11.3 shows the Bode plot of the loop transfer function G_0. Its characteristics are typical for many loop transfer functions. Due to integral action, the gain is high for low frequencies. With increased frequency, the gain decreases while the phase shift increases.

Despite the fact that we are ultimately interested in the properties of the closed-loop system, specifications can be imposed on the corresponding open-loop system. It is e.g. the number of integrators in G_0 which determines how fast setpoint changes can be tracked without a persisting control error. It is also the low-frequency gain

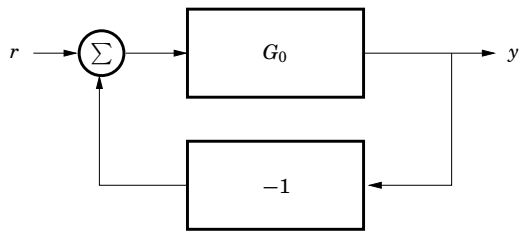

Figure 11.1 The simple feedback loop with loop transfer function $G_0(s)$.

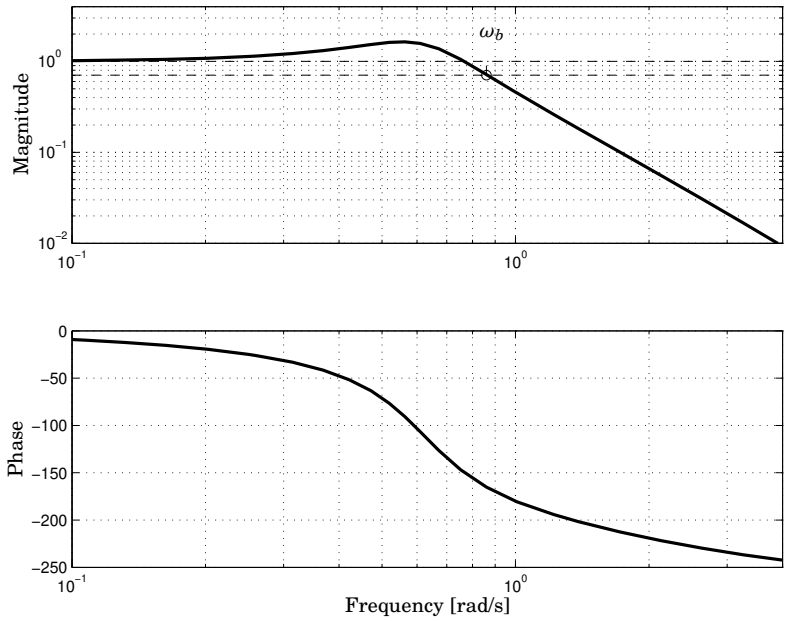

Figure 11.2 Bode plot of the closed-loop system where the bandwidth ω_b is used as a measure of system speed

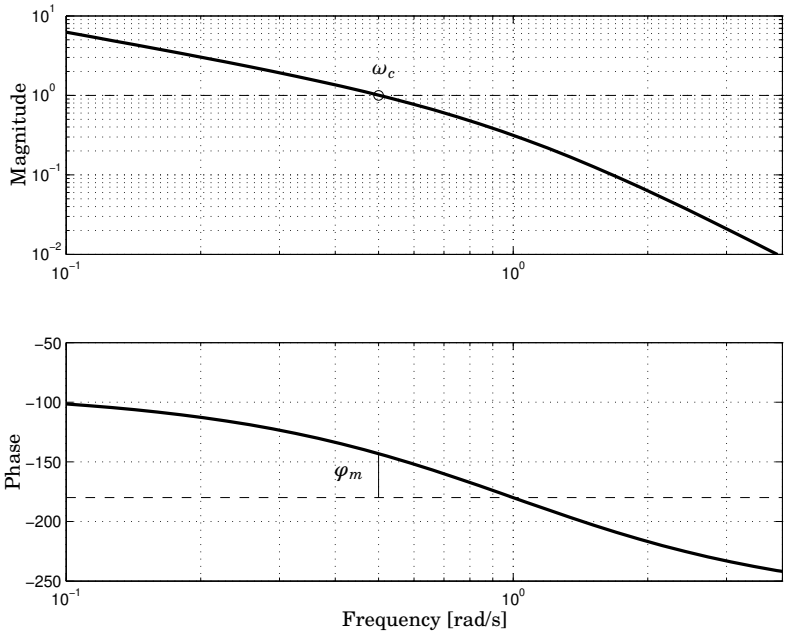

Figure 11.3 Bode plot of the open-loop system where the cross-over frequency ω_c has been used as a speed measure and the phase margin φ_m has been used as a robustness measure of the closed-loop system.

which determines how large the otherwise obtained stationary control errors become. These matters were handled in Lecture 7.

The speed of the closed-loop system is related to how high in frequency the open-loop system maintains a substantially high gain. Consequently, one could think of many ways to specify the speed of the system. Here we choose to use the cross-over frequency ω_c, i.e. the frequency for which the gain curve intersects the line $|G_0(i\omega)| = 1$.

We have previously introduced several measures on the open-loop system, which describe the robustness of the closed-loop system, e.g. the gain margin A_m, the phase

margin φ_m and the maximal value of the sensitivity function M_s. They all have in common that they somehow describe the distance to the critical point -1 in the Nyquist plot. Here we choose to use the phase margin as our robustness margin.

By reading off the low frequency gain, the cross-over frequency ω_c and the phase margin φ_m of the open-loop system G_0, we can predict the properties of the closed-loop system. If we are not satisfied by these properties, we can modify the Bode plot by introducing lead–lag compensation links.

11.2 Lead–Lag Compensation

Figure 11.4 shows the block diagram of the simple feedback loop, where the nominal loop transfer function $G_0(s)$ has been compensated using the link $G_K(s)$. The new open-loop transfer function thus becomes

$$G_0^{new}(s) = G_K(s)G_0(s)$$

This compensation link can be used to shape the new open-loop transfer function $G_0^{new}(s)$ so that it fulfills the specifications of the open-loop system.

Since

$$\begin{aligned}\log|G_0^{new}(i\omega)| &= \log|G_K(i\omega)| + \log|G_0(i\omega)| \\ \arg G_0^{new}(i\omega) &= \arg G_K(i\omega) + \arg G_0(i\omega)\end{aligned}$$

it is easy to determine the compensation link $G_K(s)$ as the difference between the desired open-loop transfer function $G_0^{new}(s)$ and the uncompensated transfer function $G_0(s)$.

Figure 11.5 shows the gain curves for two types of lead–lag compensation links. The two curves to the left show a case where the nominal open-loop transfer function G_0 has adequate high-frequency properties, but where the low-frequency gain is too low. The compensation link G_K therefore has a high gain for low frequencies, while its high frequency-gain is unity. The two figures to the right show a case where the nominal loop transfer function G_0 has adequate low-frequency properties. However, one wants to increase the speed of the closed-loop system by increasing the cross-over frequency. The compensation link G_K therefore has a gain which is unity for low frequencies, but high for higher frequencies.

We have now seen how compensation links can be obtained easily from the Bode plot by forming the difference between the desired and nominal open-loop transfer functions. In order to implement the compensation link, it must, however, be parametrized, i.e. a transfer function must be matched to the desired Bode plot. The following sections will be committed to demonstrating how this can be done.

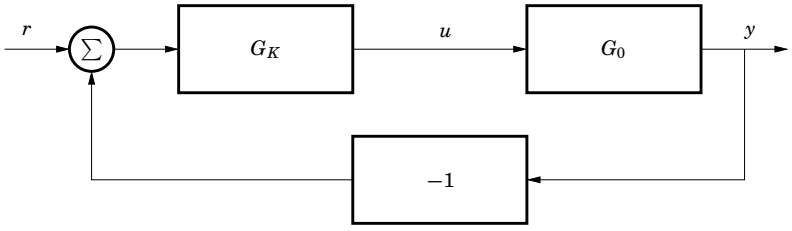

Figure 11.4 The simple feedback loop with compensation link $G_K(s)$.

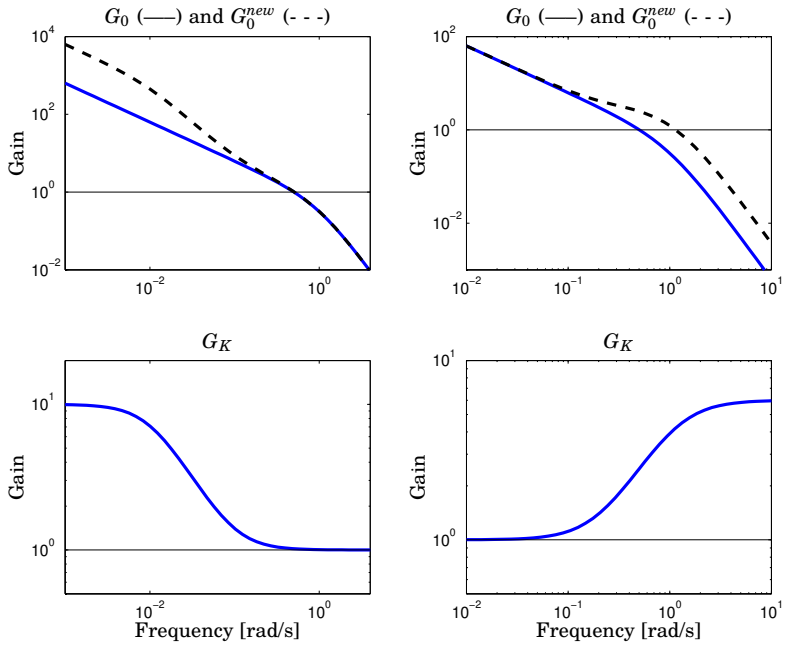

Figure 11.5 Two cases of lead–lag compensation. The plots to the left show a type of compensation which increases the low-frequency gain. The figures to the right shows another type of compensation which increases the high-frequency gain and thereby the cross-over frequency. The compensation links G_K is given by the difference between G_0^{new} and G_0.

11.3 Lag Compensation

We begin by studying the Bode plot of a compensation link with transfer function

$$G_K(s) = \frac{s+a}{s+a/M} = M\frac{1+s/a}{1+sM/a} \qquad M > 1$$

The Bode plot of G_K is shown in Figure 11.6. The low-frequency asymptote of the gain curve is $G_K(0) = M$, whereas its high-frequency asymptote is given by $G_K(s) \to 1,\ s \to \infty$. The compensation link has one pole, which makes the Bode plot break down at the frequency $\omega = a/M$ and one zero, which causes the Bode plot to break up again at $\omega = a$.

This lag compensation link has the same shape as the compensation link to the left in Figure 11.5. It increases the gain for low frequencies and is therefore used to reduce stationary errors. The term lag compensation comes from the fact that it contributes with a phase lag, which is obviously an unwanted property.

The lag link has two parameters, which need to be determined. The low-frequency gain M and the corner frequency a are determined in the following way:

M : The gain M is given by the specifications on how much we desire to decrease the stationary error. If the nominal loop transfer function G_0 contains at least one integrator, one can show that the stationary error is decreased by a factor M.

a : The corner frequency a determines at which frequency the gain of the compensation link shall break down from M to 1. The parameter a is commonly chosen so that the negative phase shift does not affect the phase margin too much. The choice can be simplified by a commonly used rule of thumb. If the parameter is chosen to be

$$a = 0.1\omega_c$$

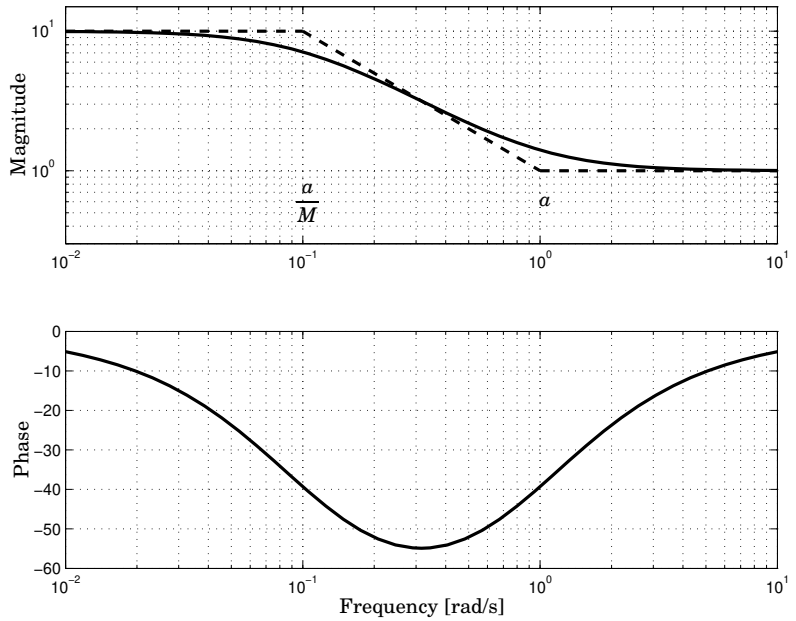

Figure 11.6 Bode plot of the lag compensation link. The parameters are given by $M = 10$ and $a = 1$.

it guarantees that the phase margin is decreased by at most 6°, assuming that ω_c is not changed. This can be seen by studying the argument of $G_K(i\omega_c)$:

$$\begin{aligned}\arg G_K(i\omega_c) &= \arctan\frac{\omega_c}{a} - \arctan\frac{M\omega_c}{a} = \arctan 10 - \arctan 10M \\ &> \arctan 10 - 90^\circ > -6^\circ\end{aligned}$$

Special Case Let us investigate what happens if $M \to \infty$. The resulting compensation link becomes

$$G_K(s) = \frac{s+a}{s+a/M} \to \frac{s+a}{s} = 1 + \frac{a}{s}, \qquad M \to \infty$$

This is the equation of a PI controller. Choosing $M \to \infty$ will thus eliminate the stationary error, since it introduces an additional integrator in the control loop.

Sometimes one chooses to introduce as many integrators as needed to eliminate persisting stationary errors, rather than determining a finite value of M in a lag compensation filter.

Example 11.1—Decreasing stationary errors
An electric motor can be described by the transfer function

$$G_P(s) = \frac{100}{s(s+10)}$$

The input is the motor current and the output is the shaft angle φ. Initially the motor is part of a negative feedback connection, as seen in Figure 11.1, i.e. $G_0 = G_P$. This yields satisfactory speed (cross-over frequency) and robustness (phase margin). Since G_0 contains an integrator it is possible to track setpoint steps without persisting control errors. However, setpoint ramps yield persistent control errors, which we wish to decrease by a factor 10.

The Bode plot of the motor is shown in Figure 11.7. The slope of the magnitude curve is -1 for low frequencies and its phase shift is $-90°$. The gain curve breaks down to slope -2 at the corner frequency $\omega = 10$, while the phase approaches $-180°$.

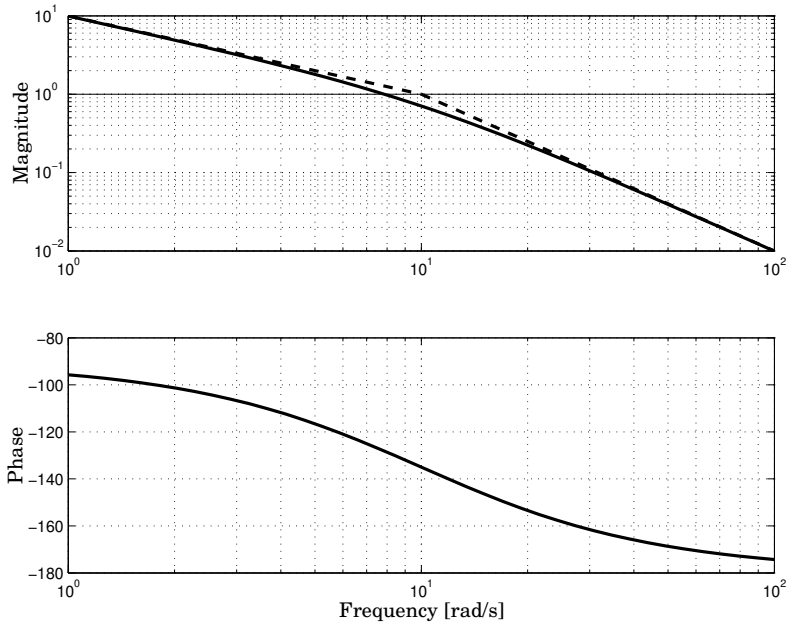

Figure 11.7 The Bode plot of the motor in Example 11.1.

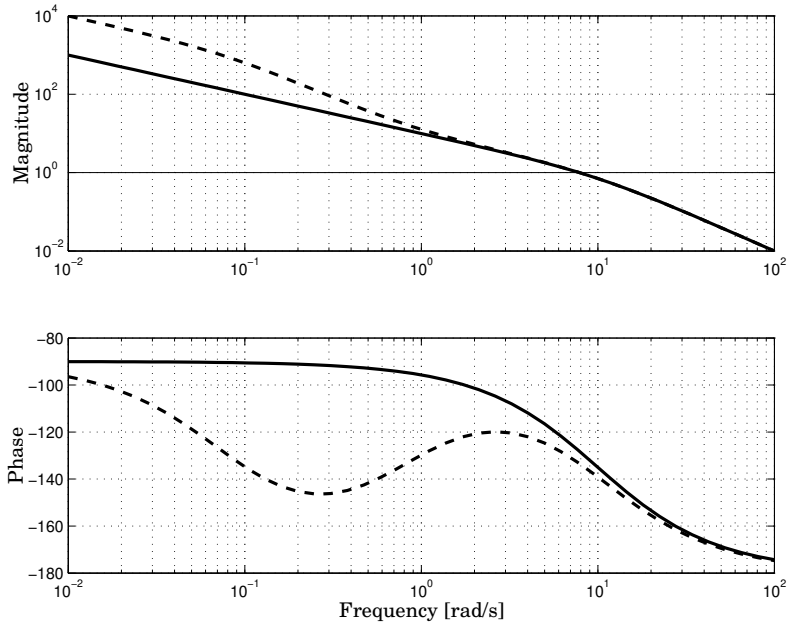

Figure 11.8 Bode plot of G_0 (——) and G_0^{new} (- - -) in Example 11.1.

We can also observe that the cross-over frequency is given by $\omega_c \approx 8$ and that the phase margin is $\varphi_m \approx 50°$.

Since we aim at decreasing the stationary error by a factor 10 and $G_0(s)$ has an integrator, we choose $M = 10$. In order not to affect the phase margin excessively we choose the corner frequency $a = 0.1\omega_c = 0.8$. This gives us the compensation link

$$G_K(s) = \frac{s+a}{s+a/M} = \frac{s+0.8}{s+0.08}$$

Figure 11.8 shows the Bode plot of the compensated system G_0^{new}. The low-frequency gain has increased by a factor 10, as desired. Meanwhile, there have been no significant changes to the properties of the system at or above the cross-over

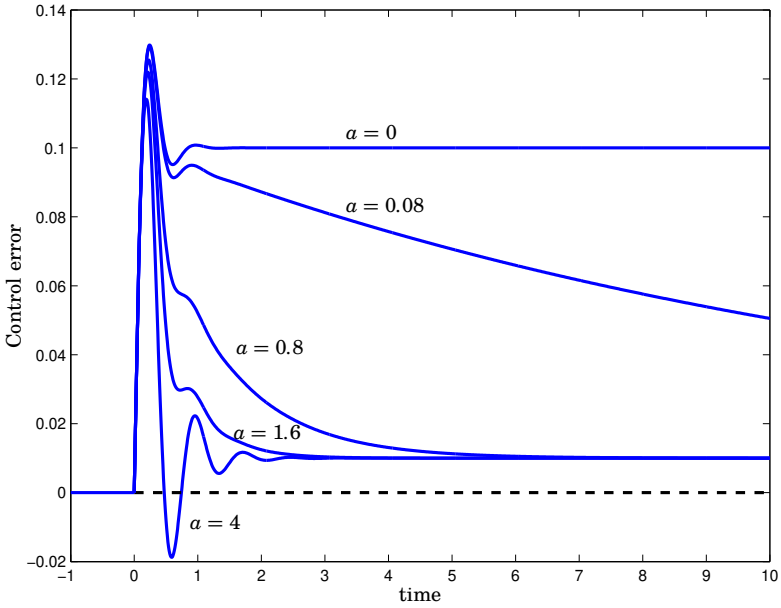

Figure 11.9 The control errors caused by a setpoint ramp for different choices of the parameter a in Example 11.1.

frequency.

Finally, Figure 11.9 shows the setpoint ramp responses of the compensated system. For clarity reasons the control error is shown, rather than the response itself. The figure shows the control error for some different choices of the corner frequency a. The case $a = 0$ corresponds to $G_K = 1$, i.e. compensation-free control. It is clear from the figure that the choice of a is a compromise between performance and robustness. If a is chosen small we obtain a robust but slow system, whereas a large a yields a fast settling but a significant overshoot. In the latter case, the phase margin has decreased significantly. The figure shows that the choice $a = 0.1\omega_c = 0.8$ is reasonable, but perhaps a little bit conservative. □

11.4 Lead Compensation

We will now move on to study the Bode plot of another type of compensation link. Its transfer function is given by

$$G_K(s) = K_K N \frac{s+b}{s+bN} = K_K \frac{1+s/b}{1+s/(bN)} \qquad N > 1$$

The Bode plot of $G_K(s)$ is shown in Figure 11.10. The low-frequency asymptote is $G_K(0) = K_K$ and the high-frequency asymptote is given by $G_K(s) \to K_K N,\ s \to \infty$. The transfer function contains a zero, which makes the Bode plot break up at the frequency $\omega = b$ and a pole which makes it break down again at $\omega = bN$.

The lead compensation link has the same appearance as the rightmost compensation link in Figure 11.5. It increases both the high-frequency gain and the phase shift of the loop transfer function. Consequently, this link can be used both to change the speed (cross-over frequency) and the robustness (phase margin).

The lead compensation link contains three parameters: K_K, N and b. In the lag compensation case we chose the parameters so that the negative phase shift should not affect the phase margin. Now we do the opposite and ensure that the peak of the positive phase curve appears at the cross-over frequency.

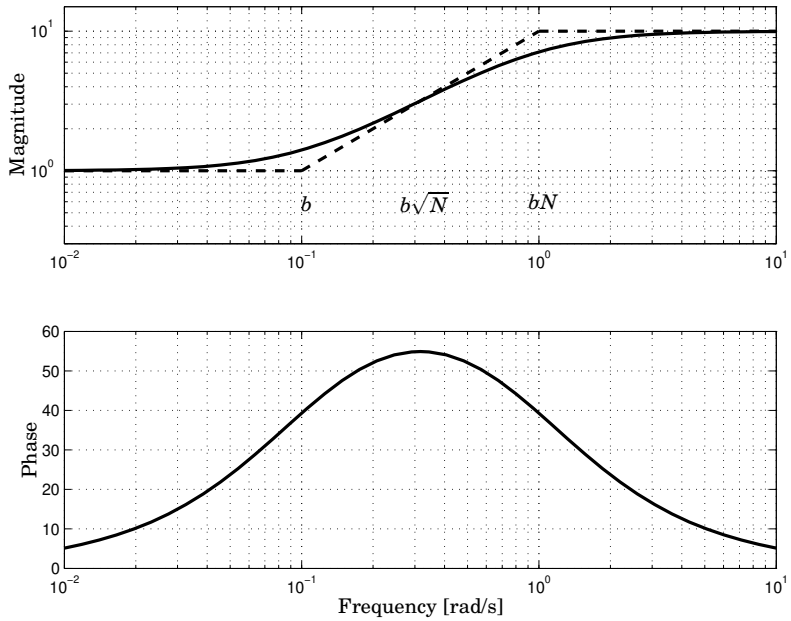

Figure 11.10 The Bode plot of a lead compensation link. The parameters are given by $K_K = 1$, $N = 10$ and $b = 0.1$.

The compensation link G_K has a phase peak at the frequency which is given as the geometric mean of the corner frequencies b and bN, i.e. $\omega = b\sqrt{N}$. For this frequency it holds that

$$|G_K(ib\sqrt{N})| = K_K\sqrt{N}$$
$$\arg G_K(ib\sqrt{N}) = \arctan(\sqrt{N}) - \arctan(1/\sqrt{N})$$

The compensation link parameters are determined in four steps:

1. The first step is to specify the desired cross-over frequency ω_c and phase margin φ_m.

2. The parameter N determines the magnitude of the phase curve peak. A larger N results in a larger phase lead. Figure 11.11 shows the relation between N and the phase lead $\Delta\varphi$. The desired phase lead at the cross-over frequency ω_c is
$$\Delta\varphi = \varphi_m - (180^\circ + \arg G_0(i\omega_c))$$

3. The next step is to ensure that the phase peak is attended for ω_c. Since the peak lies at the frequency $b\sqrt{N}$, we should choose b so that
$$b\sqrt{N} = \omega_c$$

4. The last step is to determine the parameter K_K so that the frequency ω_c really becomes the cross-over frequency, i.e.
$$|G_K(i\omega_c)| \cdot |G_0(i\omega_c)| = 1$$
Here, the following relation comes in handy:
$$|G_K(i\omega_c)| = |G_K(ib\sqrt{N})| = K_K\sqrt{N}$$

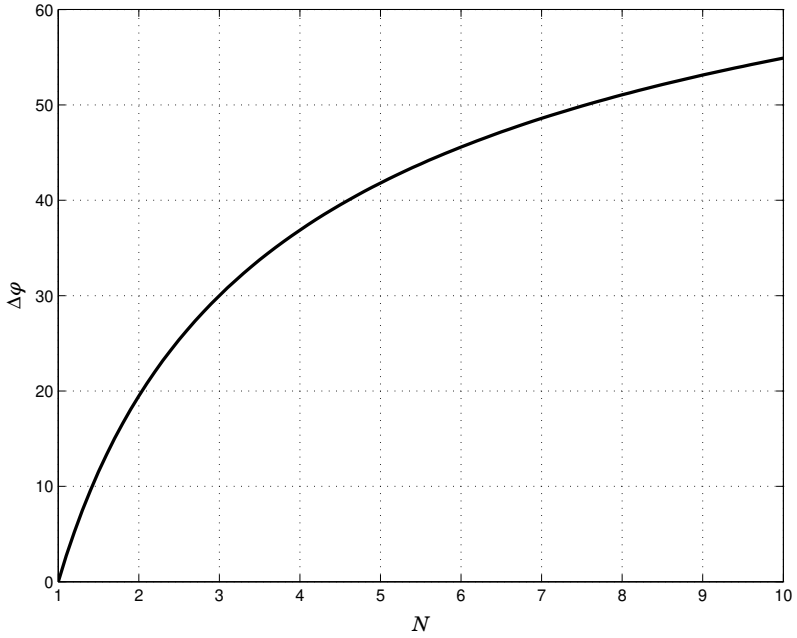

Figure 11.11 The relationship between the parameter N and the phase lead $\Delta\varphi$ in the lead compensation link.

A Special Case Let us investigate what happens if $N \to \infty$. As a result the compensation link becomes

$$G_K(s) = K_K \frac{1 + s/b}{1 + s/(bN)} \to K_K(1 + \frac{s}{b}), \qquad N \to \infty$$

This is the equation of a PD controller.

Example 11.2—Increased speed with maintained robustness
We consider the same motor as in the previous example, with transfer function

$$G_P(s) = \frac{100}{s(s+10)}$$

Initially the motor is part of the simple feedback loop shown in Figure 11.1, i.e. $G_0 = G_P$. This time we shall not affect the stationary properties, but rather make the motor control twice as fast with maintained robustness properties. The aim is thus to double the cross-over frequency without a decrease in phase margin.

Figure 11.7 shows that the nominal cross-over frequency is 8 while the corresponding phase margin is 50°. We now determine the parameters in the lead link according to the four steps given above.

1. Since the specification was to double the cross-over frequency while maintaining the phase margin, we call for $\omega_c = 2 \cdot 8 = 16$ and $\varphi_m = 50°$.
2. We see from the Bode plot that the argument of the process at the frequency $\omega_c = 16$ is given by $\arg G_0(i \cdot 16) \approx -150°$. This means that we need a phase lead of $\Delta\varphi = \varphi_m - (180° - 150°) = 20°$. Figure 11.11 shows that this corresponds to $N = 2$.
3. The next step is to choose b so that the phase peak is located at ω_c. From the equation $b\sqrt{N} = \omega_c = 16$ we obtain $b = 16/\sqrt{2} \approx 11$.
4. Finally we determine K_K so that $\omega_c = 16$ really becomes the cross-over frequency. In the Bode plot we observe that $|G_0(i\omega_c)| \approx 0.35$. The equation

$$|G_K(i\omega_c)| \cdot |G_0(i\omega_c)| \approx K_K\sqrt{N} \cdot 0.35 = 1$$

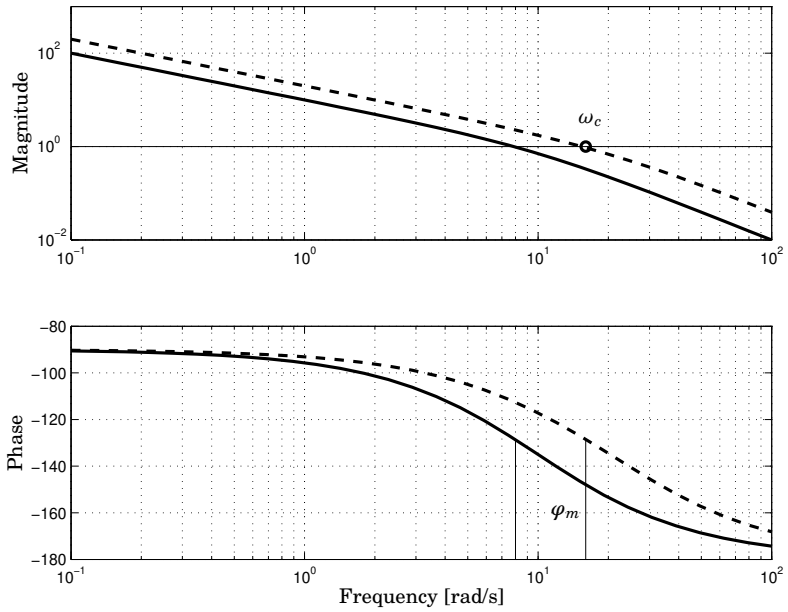

Figure 11.12 Bode plot of G_0 (——) and of G_0^{new} (- - -) in Example 11.2

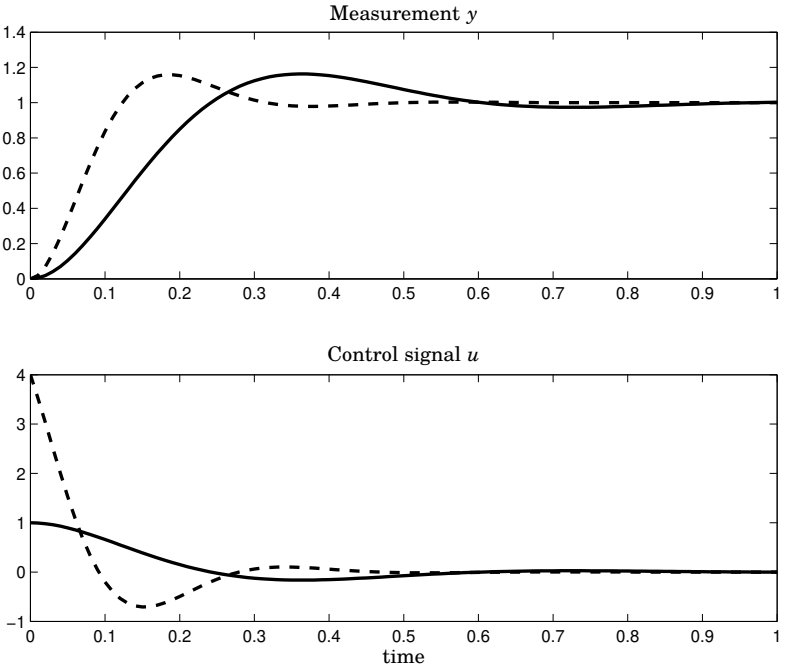

Figure 11.13 Setpoint step responses of the uncompensated system (——) and compensated system (- - -) in Example 11.2

now yields $K_K \approx 2$.

The lead compensator hence becomes

$$G_K(s) = K_K N \frac{s+b}{s+bN} = 4\frac{s+11}{s+22}$$

Figure 11.12 shows the Bode plot of the compensated system G_0^{new}. The cross-over frequency has moved to $\omega_c = 16$ with maintained phase margin, as desired.

Figure 11.13 finally shows the setpoint step responses. The figure shows that

we have succeeded in doubling the speed of the system without increasing its step response overshoot. □

Lecture 12

PID Control

In this lecture we shall first investigate the Bode plot of the PID controller and therefrom draw conclusions regarding the function and tuning of the controller. Subsequently, we review a few simple rules of thumb for PID tuning. The second part of the lecture is devoted to handling of setpoint values and transfer function zeros. Finally, a few augmentations of the PID controller, needed for practical implementations, are introduced.

12.1 The Bode Plot of the PID Controller

In the previous lecture we saw that one can use the Bode plot of the open-loop system in order to determine compensation links which give the closed-loop system desired properties.

We use the same methodology to investigate both the functionality and parameter interpretation of the PID controller. In order to do this we draw the Bode plot of the PID controller. For simplicity we choose to draw the Bode plot for a certain parametrization of the PID controller, namely the series form.

The Series Form of the PID Controller

Thus far we have assumed that the PID controller is described by the equation

$$u = K\left(e + \frac{1}{T_i}\int e(t)dt + T_d\frac{de}{dt}\right)$$

with corresponding transfer function

$$G_R(s) = K\left(1 + \frac{1}{sT_i} + sT_d\right)$$

This form is known as the parallel form, since the control error e is treated in parallel in the P, I and D parts. An equally common form in industrial applications is illustrated by the transfer function

$$G'_R(s) = K'\left(1 + \frac{1}{sT'_i}\right)(1 + sT'_d)$$

This is known as the series form, since it can be described as a series connection of a PI and a PD controller. The difference between these two forms is not as large as it might appear. If we multiply the factors in the series form we arrive at

$$G'_R(s) = K'\left(1 + \frac{1}{sT'_i}\right)(1 + sT'_d) = K'\left(1 + \frac{T'_d}{T'_i} + \frac{1}{sT'_i} + sT'_d\right)$$

The controller thus contains P, I and D parts. The only difference between the two forms is hence their parametrizations. The relations are given by the following

equations:

$$K = K' \frac{T_i' + T_d'}{T_i'} \qquad K' = \frac{K}{2}\left(1 + \sqrt{1 - \frac{4T_d}{T_i}}\right)$$

$$T_i = T_i' + T_d' \qquad T_i' = \frac{T_i}{2}\left(1 + \sqrt{1 - \frac{4T_d}{T_i}}\right)$$

$$T_d = \frac{T_i' T_d'}{T_i' + T_d'} \qquad T_d' = \frac{T_i}{2}\left(1 - \sqrt{1 - \frac{4T_d}{T_i}}\right)$$

If one switches from one controller form to the other and simultaneously translates the controller parameters appropriately, the functionality of the controller remains unchanged.

Two interesting observations can be made concerning the structures. The first is that the two representations are identical when the controllers are used as either P, PI or PD controllers. It is only when all three terms are used that we have a difference in parametrization between the series and parallel forms.

The second observations is that the parallel form is more general than its series counterpart. We can always translate a controller from series to parallel form, but the contrary is true only for the special case

$$T_i \geq 4T_d$$

This can be seen by comparing the transfer functions. The PID controller has a pole in the origin and two zeros. In the parallel form the two zeros can be complex, while the series form only allows for real zeros.

When the PID controller was implemented with pneumatic technology back in the 30- and 40-ies, it was done in series form, for practical reasons. The explanation to why there still exist so many controllers in series form is that many manufacturers have kept the controller structure, though the technology used to implement the controllers have changed.

The Bode Plot of the PID Controller

We shall now study the Bode plot of the PID controller and choose to do so for its series form. This form is easier to draw due to its real zeros. Also, the interpretation of the controller parameters is more straightforward for the series form. The drawn conclusions will, however, be valid also for the parallel form.

The transfer function G_R' can be written

$$G_R'(s) = K'\left(1 + \frac{1}{sT_i'}\right)(1 + sT_d') = \frac{K'}{sT_i'}(1 + sT_i')(1 + sT_d')$$

The low-frequency asymptote of the Bode plot is $K'/(sT_i')$ while its high-frequency asymptote is given by $K'T_d's$. The Bode plot shows two corner frequencies. Under the assumption that $T_i' > T_d'$ the first corner frequency will appear at $\omega = 1/T_i'$ while the second lies at $\omega = 1/T_d'$. Both the corner frequencies correspond to zeros, causing the Bode plot to break upwards.

The Bode plot of the PID controller is shown in Figure 12.1. The Bode plot clearly shows the role of the three parameters. The gain K' determines the level of the magnitude curve whereas the integral and derivative times T_i' and T_d' determine its two corner frequencies. We shall exploit the Bode plot in order to investigate the meanings of the PID parameters for the closed-loop system.

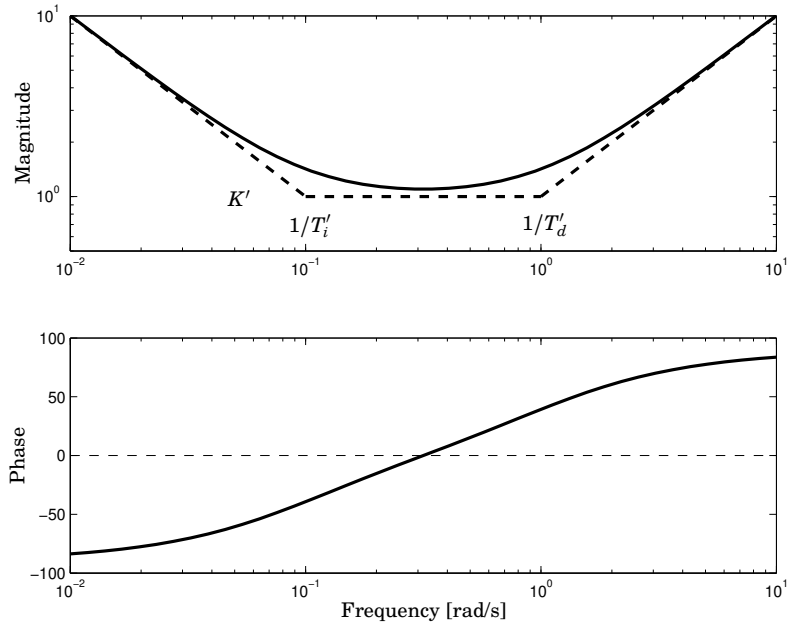

Figure 12.1 Bode plot of the PID controller with parameters $K' = 1$, $T_i' = 10$ and $T_d' = 1$.

The PID Parameters

Figure 12.2 shows how the Bode plot is affected when the three controller parameters are varied. We shall now investigate how these variations affect the properties of the closed-loop system. As we do this we assume that we have a fairly well tuned controller. This implies that the cross-over frequency ω_c lies close to both corner frequencies of the controller, i.e. close to $1/T_i'$ and $1/T_d'$. Further we assume that the loop transfer function exhibits the typical structure shown in Figure 11.3, with declining gain and phase.

Increased K' An increase of the gain K leads to an equal raise of the gain curve for all frequencies while the phase curve remains unaffected. The increased low-frequency gain causes a possible decrease of the stationary error.

The increased gain will increase the cross-over frequency ω_c of the compensated system, which leads to increased speed of the closed-loop system. Since the phase margin is now obtained at a higher frequency, the increased gain will also lead to decreased robustness resulting from the smaller phase margin.

Decreased T_i' A decrease of T_i' means an increase of the integral action, since T_i' occurs in the denominator of the I part. A decrease of T_i' leads to an increased low frequency gain, while the phase is decreased. The higher low frequencies gain causes a decrease of any stationary errors.

The increased gain leads to an increase in the speed of the closed-loop system. Since the phase margin is now to be evaluated at a higher frequency, while the phase has decreased, the robustness of the closed loop system has decreased.

Increased T_d' An increase of T_d' causes an increased gain for high frequencies and an increased phase. The cross-over frequency ω_c increases, due to the increased gain.

The increased cross-over frequency results in a faster system. The fact that the phase margin is now evaluated for a higher frequency would generally results in reduced robustness. However, we cannot draw this conclusion since the phase has also been increased. There is actually a possibility of increasing both speed and robustness by increasing T_d'. This is, however, only true within certain bounds. Further increase of T_d' yields a decrease of the robustness.

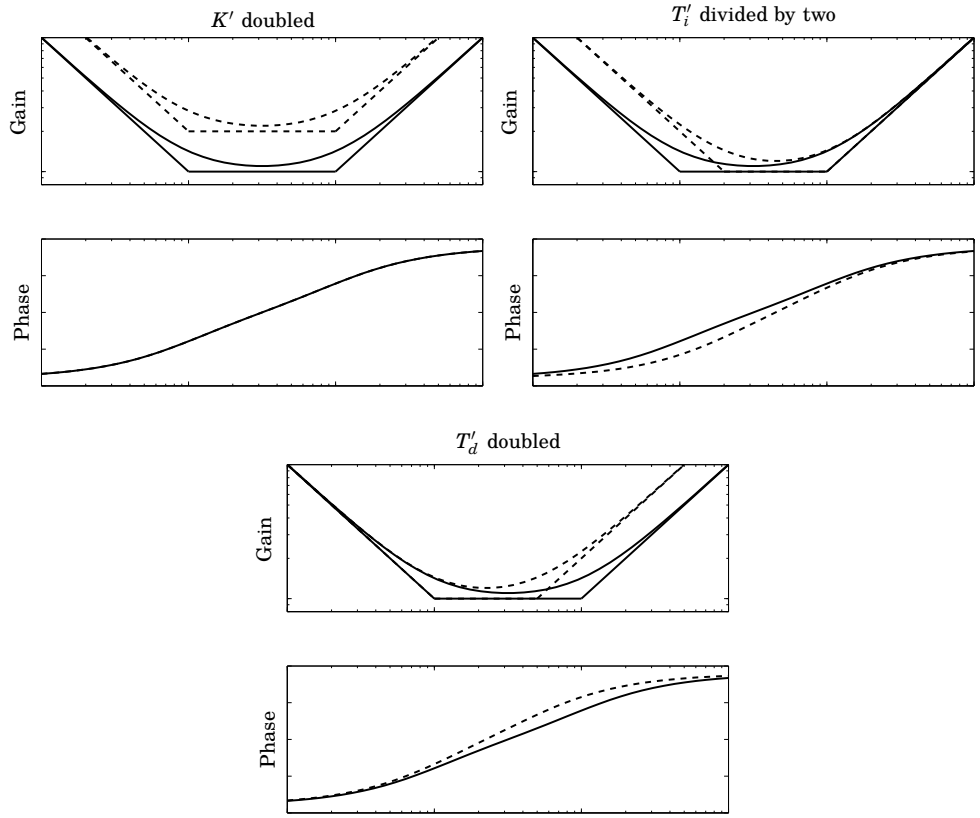

Figure 12.2 Bode plot of the PID controller. The dashed curves show how the Bode plot is affected when the three parameters K', T_i' and T_d' are varied.

12.2 Simple tuning rules

The PID controller is an elementary controller, often used in simple applications where one lacks the time and knowledge needed to further analyze the control problem. Consequently, there is a demand for easily applied rules of thumb, which can be used to tune the controller parameters to an acceptable performance level. The tuning should preferably be based on simple experiments conducted on the process.

Many tuning methods have been suggested since the PID controller was first introduced. The by far most well known are, however, Ziegler–Nichols' methods. They are not the best methods, but among the easiest to apply. It is worth noting that these methods are only to be considered as rules of thumb. They yield often acceptable controller parameters. If one has higher demands of controller performance, more elaborate methods may be needed.

The minimal amount of information needed to control a process is a process gain in order to determine K and a process time for the determination of T_i and T_d. Ziegler and Nichols presented two methods in the 40-ies, which can be used to obtain these parameters, a step response method and a frequency method.

The most popular method in industry today is the Lambda method. It was derived in the sixties, and it provides acceptable control performance for a large class of processes.

Ziegler–Nichols' Step Response Method

Ziegler–Nichols' step response method is based on manual control of the process, i.e. that the control signal u is manually governed and no controller is present.

When the process has reached an equilibrium a step is issued in the control signal. We assume the step size to be 1. Steps of different magnitude will require

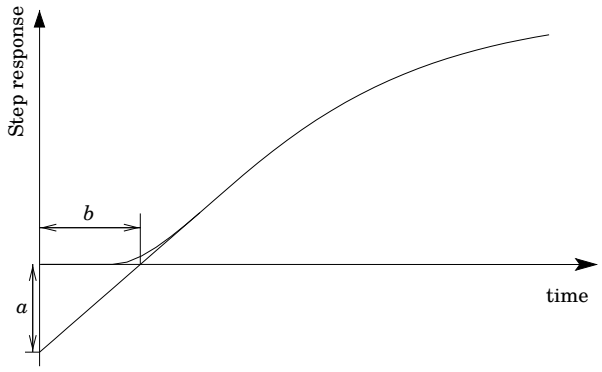

Figure 12.3 Evaluation of the gain a and time b from a process step response.

Controller	K	T_i	T_d
P	$1/a$		
PI	$0.9/a$	$3b$	
PID	$1.2/a$	$2b$	$0.5b$

Table 12.1 Recommended controller tuning according to Ziegler–Nichols' step response method.

normalization of the response in the measurement signal.

Figure 12.3 shows a typical step response. A tangent is drawn through the point where the incline of the step response attains a maximum. The gain a and time b are then obtained from the intersection between this tangent and the coordinate axis. Ziegler and Nichols suggested that the obtained parameters should yield the PID parameters according to table 12.1.

From the table we see that the controller gain K is chosen to be inversely proportional to the process gain a and that the controller times T_i and T_d are chosen proportional to the process time b.

Ziegler–Nichols' Frequency Method

Ziegler and Nichols frequency method is based on incorporating a P controller in the loop and thereafter conducting the following steps:

1. Successively adjust K until the process oscillates with a constant amplitude. The corresponding gain is denoted K_0.
2. Measure the period T_0 of the oscillation.
3. Choose the controller parameters from table 12.2.

In Ziegler–Nichols' frequency method we identify the point $G_P(i\omega_0)$, being the point where the phase shift is $-180°$. If a P controller is installed in the loop and its gain is gradually increased, one will eventually reach the gain K_0 for which $|K_0 G_P(i\omega_0)| = 1$. This corresponds to the limit of stability. The gain K_0 thus holds adequate information to calculate $G_P(i\omega_0)$ while the frequency ω_0 is given by the period T_0, $\omega_0 = 2\pi/T_0$.

Controller	K	T_i	T_d
P	$0.5K_0$		
PI	$0.45K_0$	$T_0/1.2$	
PID	$0.6K_0$	$T_0/2$	$T_0/8$

Table 12.2 Recommended controller tuning according to Ziegler–Nichols' frequency method.

Now that we have realized how Ziegler–Nichols' frequency method works, we can observe that in the case of P control it suggests a controller with gain margin $A_m = 2$.

The Lambda method

The Lambda method is based on a step response experiment, where static process gain K_p, time delay L, and time constant T are determined. The experiment is shown in Figure 12.4. The point where the process output has the largest derivative is first determined, and the tangent to the curve at this point is drawn. The intersection between this tangent and the line representing the level of the process output before the step change is then determined. The time from the step change to this intersection point gives an estimate of time delay L. Time constant T is then determined from the time it takes to reach 63 % of the final value. Static gain K_p is finally determined by dividing the change in process output with the magnitude of the control signal step:

$$K_p = \frac{\Delta y}{\Delta u}$$

The Lambda method has one parameter that can be set by the user, namely the desired time constant of the closed-loop system. This time constant is called lambda λ.

The original Lambda method only considered PI control. The tuning rule is

$$\begin{aligned} K &= \frac{1}{K_p}\frac{T}{L+\lambda} \\ T_i &= T \end{aligned} \tag{12.1}$$

The integral time is chosen equal to the process time constant T. The gain, however, is dependent on the choice of λ. A common choice is $\lambda = T$, which means that it is desired to give the closed-loop system the same time constant as the open-loop system.

The controller parameters are derived in the following way. The process and the controller transfer functions are

$$G_P(s) = \frac{K_p e^{-sL}}{1+sT} \qquad G_R(s) = K\frac{1+sT_i}{sT_i}$$

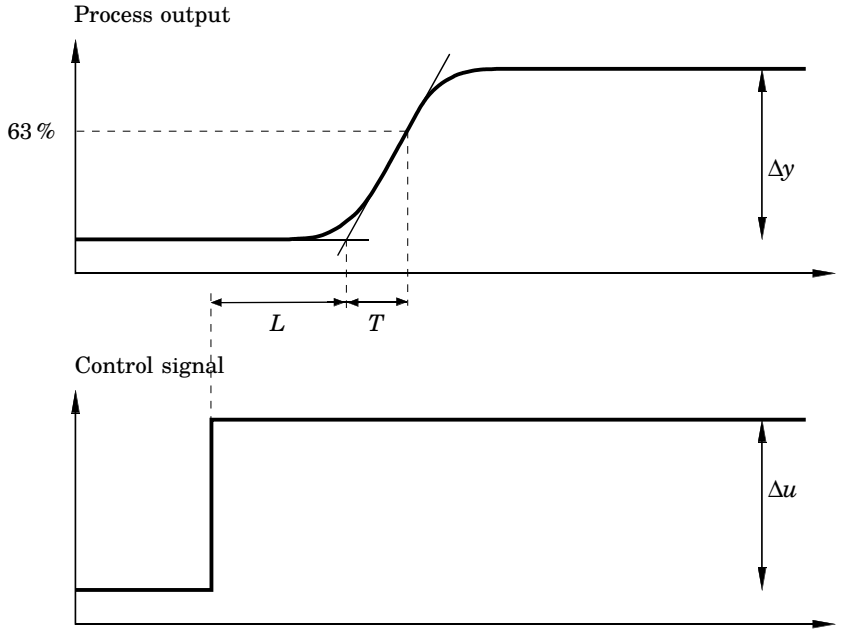

Figure 12.4 Determination of K_p, L, and T from a step response experiment.

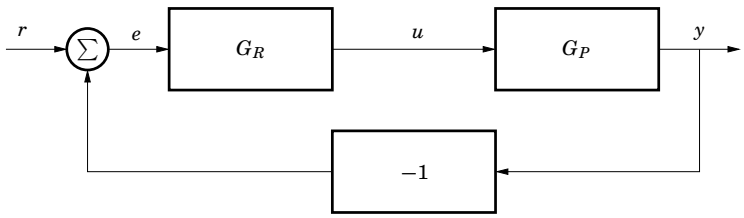

Figure 12.5 Standard structure of the simple feedback loop.

Since $T_i = T$, the loop transfer function is

$$G_0(s) = \frac{K_P K e^{-sL}}{sT}$$

The closed-loop transfer function between the setpoint and the process output becomes

$$G(s) = \frac{G_0(s)}{1 + G_0(s)} = \frac{K_P K e^{-sL}}{sT + K_p K e^{-sL}} \approx \frac{K_P K e^{-sL}}{sT + K_p K(1 - sL)} = \frac{e^{-sL}}{1 + s(T/(K_p K) - L)}$$

where the approximation is that the time delay in the denominator is replaced by the first terms in its Taylor series expansion. Therefore, the closed-loop transfer function is a first-order system with the same time delay as for the process, and with the time constant $\lambda = T/(K_p K) - L$. By specifying λ, K is given by Equation (12.1).

It is possible to derive tuning rules for PID controllers with the same approach as for the PI controller. In this case, the time delay in the denominator of the closed-loop transfer function is not approximated by a Taylor series expansion, but with $e^{-sL} \approx (1 - sL/2)/(1 + sL/2)$. The ingegral and derivative times are chosen so that poles in the loop transfer function are cancelled, in the same way as for the PI controller. The rules for the series and parallell form are:

$$K' = \frac{1}{K_p}\frac{T}{L/2 + \lambda} \qquad K = \frac{1}{K_p}\frac{L/2 + T}{L/2 + \lambda}$$

$$T_i' = T \qquad T_i = T + L/2$$

$$T_d' = \frac{L}{2} \qquad T_d = \frac{TL}{L + 2T}$$

12.3 Setpoint Handling

A controller has normally two inputs, setpoint r and measurement signal y, and one output, control signal u. As we have analyzed and synthesized controllers we have generally assumed that the controller structure is that shown in Figure 12.5.

This works fine when analyzing the poles and stability margins of the closed-loop system. The structure can also be used when calculating properties of the closed-loop system subject to load disturbances. However, the setpoint value is commonly not handled as shown in Figure 12.5. In Figure 12.5 the control error $e = r - y$ is formed and constitutes the controller input so that

$$U = G_R E = G_R(R - Y)$$

This means that the setpoint and measurement signals are treated identically by the controller. However, one often wants to treat the two signals differently. We shall give examples of such situations later in this lecture.

A more general controller structure is shown in Figure 12.6. The control signal is now given by

$$U = G_{R1} R - G_{R2} Y$$

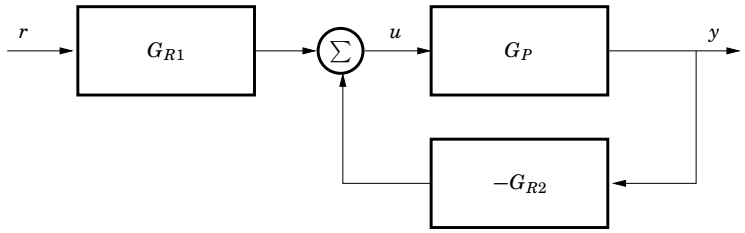

Figure 12.6 Structure with two degrees of freedom.

One usually says that the controller structure shown in Figure 12.6 has two degrees of freedom, since it provides us with the freedom to treat the two inputs separately. This freedom is typically used to improve control performance with respect to setpoint changes.

A PID controller, tuned to perform well under load disturbances, often yields large overshoots at setpoint steps if the standard structure is used. Hence it is common to modify the PID controller in process control applications, as discussed in the next section. In addition it is customary to let steps in the setpoint pass through a ramp function or a low pass filter in order to further depress the overshoot.

In servo applications, setpoint tracking is the main control objective. Consequently a large part of the control calculations are committed to obtaining satisfactory setpoint tracking.

Zeros

Previously in the course we have observed that the poles of the transfer function have a critical role for the properties of the closed-loop system. We have, however, not discussed the role of its zeros. We shall now show that the zeros play an important role when it comes to setpoint handling.

We begin by studying a control loop with transfer function $G(s)$, measurement signal y and setpoint input r, as shown in Figure 12.7. Figure 12.8 shows the output y and its derivative $\dot{y}$ resulting from a step in the setpoint r.

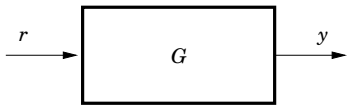

Figure 12.7 The nominal transfer function G where the output is given by the measurement signal y while the setpoint r constitutes the system input.

Assume that the transfer function is extended by the zero $s = z$, so that the transfer function from setpoint to measurement is given by

$$G_1(s) = G(s)\left(1 - \frac{s}{z}\right)$$

We denote the new measurement y_1. The new control loop is described by the block diagram shown in Figure 12.9. It is evident from the figure that the measurement y_1 is a weighted sum of the old measurement y and its derivative $\dot{y}$, since

$$y_1 = y - \frac{1}{z}\dot{y}$$

The position of the zero z determines the weighting. The zero has most influence when it lies close to the origin, i.e. when $|z|$ is small.

Figure 12.10 shows the step response for some different values of the zero z. The step response is given by a weighting of the two signals shown in Figure 12.8.

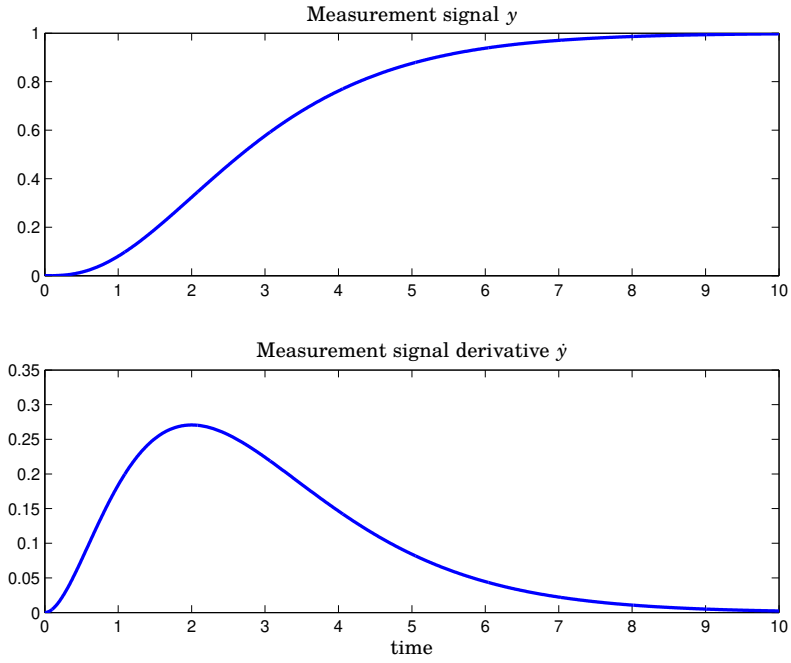

Figure 12.8 The step response and its derivative for the system $G(s) = (s+1)^{-3}$.

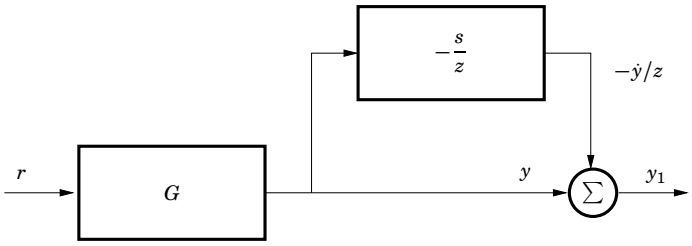

Figure 12.9 Block diagram of the system $G_1 = G(1 - s/z)$.

For positive values of z, i.e. for right-half plane zeros, the weighting of the derivative $\dot{y}$ will be negative. This results in a step response which starts in the 'wrong' direction.

For negative z, i.e. left-half plane zeros, the weight of the derivative $\dot{y}$ will be positive. This results in a faster response and large overshoots can be obtained if the zero lies close to the origin.

Zeros are further discussed in the next section, in connection to setpoint weighting in the PID controller.

12.4 Practical Modifications of the PID Controller

The elementary form of the PID controller is given by the equation

$$u = K\left(e + \frac{1}{T_i}\int e(t)dt + T_d\frac{de}{dt}\right)$$

In order to obtain a practically useful controller, this form is usually modified in several ways. We shall here bring up the most common modifications and do so by treating the three terms separately.

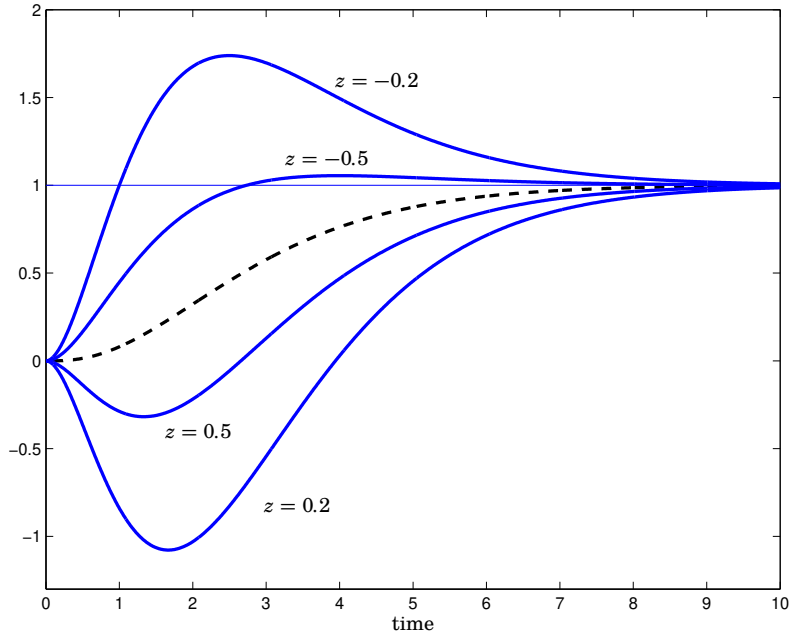

Figure 12.10 Step response of the system $G_1(s) = (s+1)^{-3}(1-s/z)$ for different values of the zero z. The dashed curve shows the nominal case with no zero.

Modification of the P Part

The proportional term of the PID controller is given by

$$u_P = Ke$$

A common modification is to exchange the nominal P part for

$$u_P = K(br - y)$$

where b is a weighting factor normally chosen so that $0 \leq b \leq 1$. This means that one exploits the two degrees of freedom and consequently treat r and y differently.

The reason that one sometimes wants to reduce the magnitude of the setpoint value in the P part is partly to reduce the overshoot at steps in r and partly to avoid that rapid setpoint changes result in wear of the actuators.

For a PI controller with setpoint weighting the control signal is given by

$$U = K\left(bR - Y + \frac{1}{sT_i}(R-Y)\right) = K\frac{1+sbT_i}{sT_i}R - K\frac{1+sT_i}{sT_i}Y$$

The setpoint weighting thus affects the zeros of the transfer function form r to u. For the standard structure where $b = 1$, the zero lies in $s = -1/T_i$. The setpoint weighting b moves the zero to $s = -1/(bT_i)$. If $0 \leq b < 1$ the zero will be moved further into the left-half plane and thus decreasing the overshoot, as we have just seen.

Figure 12.11 shows PI control for which the weighting factor b is being varied. The figure shows a setpoint change followed by a load disturbance. It reveals that the response to the load disturbance is unaffected by b, unlike the setpoint response. Smaller b result in decreased overshoots and control signal steps at the setpoint change.

Modification of the I Part

The integral term of the PID controller is given by

$$u_I = \frac{K}{T_i}\int e(t)dt$$

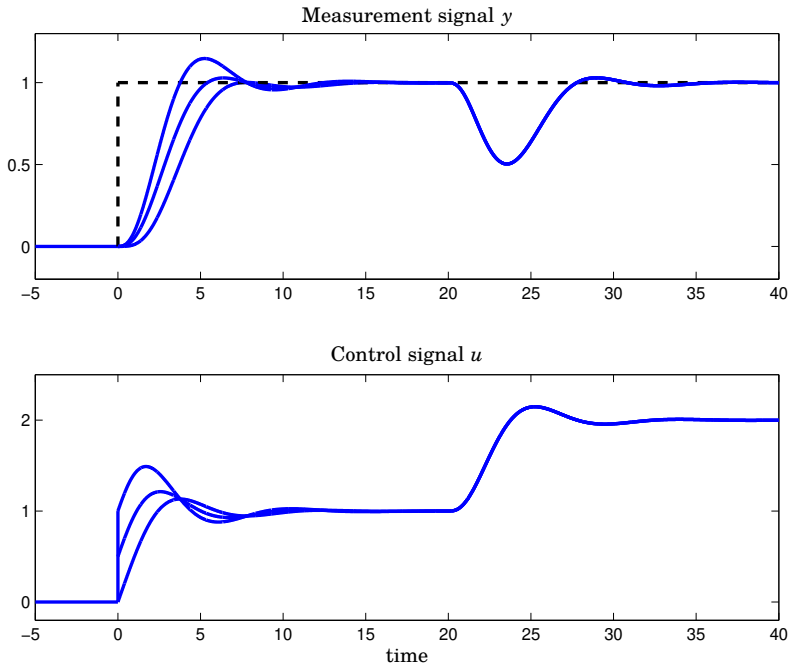

Figure 12.11 PI control of the process $G_P(s) = (s+1)^{-3}$. The setpoint weighting b is given by $b = 0$ (the slowest response), $b = 0.5$ and $b = 1$ (the fastest response). The figure shows the response of a setpoint step at $t = 0$ followed by a load step at $t = 20$.

Its purpose is to eliminate stationary errors. It is hence important that it is the actual controller error, which we integrate. Consequently, we do not apply setpoint weighting in the integral part.

An inherently unstable process requires stabilizing feedback. Likewise, the addition of an integrator renders the controller unstable, requiring stabilizing feedback. This can lead to problems if the signals in the control loop are confined within certain bounds, since the feedback no longer works when the control signal becomes saturated. We illustrate what could happen in Figure 12.12.

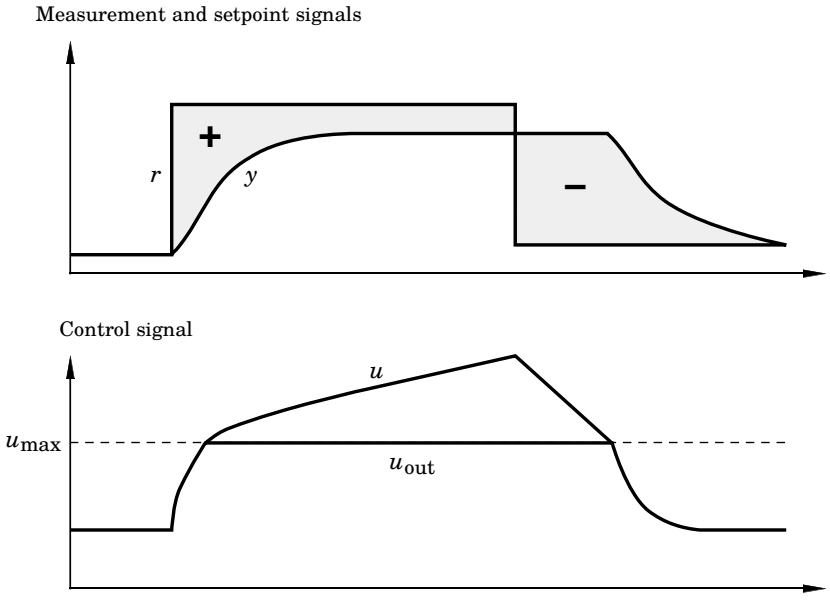

Figure 12.12 Integrator windup caused by a limited control signal.

Figure 12.12 shows the control signal, the measurement signal and the setpoint for a case where the control signal is limited.

After the first setpoint change, the control signal grows to its upper bound $u_{\max}$. The saturated control signal is not sufficient to eliminate the control error. The error integral and thereby the integral term u_I will thus grow linearly. Consequently, the desired control signal grows. We hence obtain a difference between desired control signal u and the applied control signal u_{out}.

Figure 12.12 shows what happens when the setpoint is eventually decreased to a level for which the controller can eliminate the error. The setpoint change results in a change in the control error which causes a decrease of the integral part and ultimately also in the control signal u. Since the integral part has been allowed to grow while the control signal was saturated, it will take some time before the applied control signal u_{out} starts to decrease.

This phenomenon is known as integrator windup. The easiest way to eliminate this problem is to stop updating the integral part when the control signal is saturated. This obviously requires the saturation to be known. Most industrial PID controllers are equipped with methods to avoid integrator windup. They are called anti-windup.

Modification of the D Part

The derivative term of the PID controller is given by

$$u_D = KT_d \frac{de}{dt} = KT_d \left(\frac{dr}{dt} - \frac{dy}{dt} \right)$$

If the control error e is derived accordingly, the resulting changes in u_D may become very large if e varies rapidly. We shall introduce two modifications of the D part which address the problem of rapid changes in r and y, respectively.

It is very common to exclude the setpoint r from the derivative part, resulting in

$$u_D = -KT_d \frac{dy}{dt}$$

In many cases, especially within the process industry, the setpoint is constant during long periods of time. When eventually there is a change in the setpoint, it is often rapid. In these contexts it is reasonable not to include the setpoint in the derivative part, since the derivative of the setpoint is zero during long periods of time. However, at setpoint changes, the derivative grows very large resulting in large steps in the control signal, leading to unnecessary wear on the actuators.

In servo problems and other cases which do not involve fast variations of the setpoint, it is of course still practical to include the setpoint in the derivative part.

Fast, high-frequency variations in the measurement signal y are seldom caused by variations in the actual process, but rather by measurement noise. It is therefore not desirable to derive these high frequent variations. In the Bode plot of a PID controller shown in Figure 12.1, one can clearly see that the gain increases with the frequency. A certain gain increase in the controller is desirable. However, there is no reason to increase the gain also for higher frequencies. This can be prevented by augmenting the derivative part with a low-pass filter, changing the transfer function of the derivative term according to

$$KT_d s \quad \longrightarrow \quad \frac{KT_d s}{1 + sT_f}$$

The time constant T_f of the low-pass filter is often chosen to be a factor N times the derivative time, i.e.

$$T_f = \frac{T_d}{N}$$

where N normally lies in $[5, 10]$.

The low-pass filter adds a pole to the transfer function of the PID controller. This pole changes the high frequency gain of the PID controller into $K(1 + N)$, i.e. constant rather than increasing as previously. It is often desirable to decrease the disturbance sensitivity even further by introducing an additional filter, which results in a decreasing - rather than constant - high frequency gain.

Lecture 13

Controller Structures and Implementation

Thus far we have studied control problems based on the structure of the simple feedback loop with *one* measurement signal and *one* control signal. In practise it is not unusual to have several available signals in order to solve various control problems. There exists several widely used controller configurations, where several signals are used. We shall here review three of the most common controller structures: cascaded controllers, feed forward and delay compensation. Finally we address some questions connected to the fact that most current controllers are implemented in computers rather than analog circuits.

13.1 Cascaded Controllers

Cascade control is a strategy where two controllers are combined so that the output of the first controller forms the setpoint of the other. This is illustrated by the following example.

Example 13.1—Control of a Heat Exchanger
We shall study control of a heat exchanger where steam on the primary side is used to heat water on the secondary side. We want to control the temperature on the secondary side of a heat exchanger by controlling the steam valve on its primary side. This can be achieved by letting the temperature controller actuate the steam valve directly as shown in Figure 13.1. What actually affects the temperature is not the position of the valve, but rather the steam flow. If the valve is linear and the steam flow does not vary, there is a constant relation between the valve position and the steam flow. Usually, however, valves exhibit some form of nonlinearity and the steam pressure varies over time.

E.g. assume that the steam pressure on the primary side suddenly starts to decrease. As a consequence the steam flow will decrease, leading to slower heating of the water on the secondary side. The temperature controller will issue a control

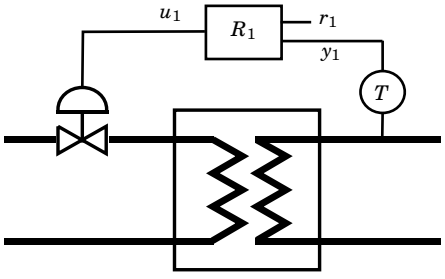

Figure 13.1 Temperature control of a heat exchanger.

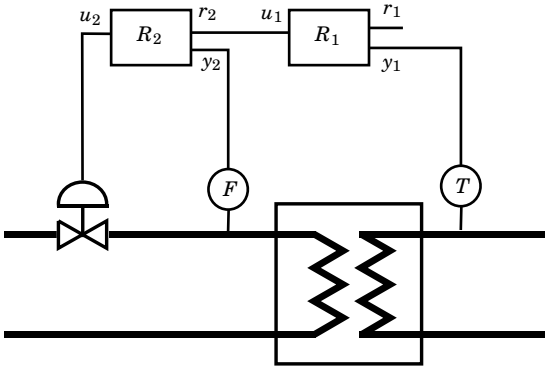

Figure 13.2 Cascade control of a heat exchanger.

signal corresponding to a more open valve position and after a while the steam flow will anew stabilize at a correct level. Consequently this strategy works, but the price is rather large disturbances in the temperature.

If one can measure the steam flow it is possible to incorporate a flow controller according to Figure 13.2. We form an inner control loop, which controls the steam flow. The setpoint of the flow controller R_2 is given by the control signal of the temperature controller R_1. This is an example of a cascade control.

Cascading the controllers leaves the master controller R_1 with a simpler task. Rather than letting R_1 bear the entire control burden, part of the assignment is reassigned to the controller R_2. The controller R_1 now only needs to produce a flow setpoint. Subsequently, it is up to the flow controller to maintain a flow close to this setpoint. A pressure variation will efficiently be eliminated by the flow controller, leading to a decreased disturbances in the temperature, as compared to the case with only one PID controller. □

The general principle of cascaded control is shown in Figure 13.3. The primary goal is to control the signal y_1 by means of the controller R_1. This could be achieved by using merely the controller R_1 and letting its control signal enter the process directly. During cascade control one exploits the availability of an additional measurement signal, y_2. By locally establishing a feedback connection from y_2 by means of the controller R_2 one can achieve more efficient control, than would be possible with only one controller. The controller R_1 is often referred to as the primary or master controller, whereas R_2 is known as the secondary or slave controller.

Cascade control is a typical example of how one can achieve more advanced control solutions, despite the simple structure of the PID controller, by combining several controllers. The foremost reason to use cascade control is to handle disturbances which enter the process at P_2, before they give rise to disturbances in the primary measurement signal y_1. An example of this are the pressure variations in the above example. A prerequisite is obviously that the inner control loop is significantly faster than the outer one. Another advantage with cascade control is that the dynamics which the primary controller is to control can be simplified. Without cascade control, the controller R_1 works on a process consisting of the two section P_1 and P_2. When cascading is implemented the process sections are changed to a

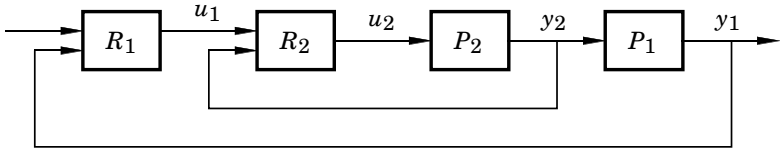

Figure 13.3 The principle of cascade control.

combination of P_1 and P_2 in a feedback connection with R_2.

13.2 Feed Forward

Feedback is an efficient method to solve control problems. One measures the signal to be controlled, compares it to a setpoint and subsequently computes a control signal based on this comparison. The feedback strategy, however, has a weakness in that the controller does not react to disturbances before they contribute to the control error. In many cases it is possible to measure a disturbance before it affects the control error. By compensating for disturbances at this earlier stage in the loop it is possible to dramatically improve control performance. The strategy used to do this is known as feed forward.

A well-known example application of feed forward is temperature control in apartments. Most apartments are equipped with a thermometer. It measures the outdoor temperature, being the most significant disturbance when controlling the indoor temperature.

By exploiting the information about the outdoor temperature one can compensate for variations in the outdoor temperature before they affect the indoor temperature. If, for instance, the outdoor temperature decreases, the feed forward suggests an increased radiator water temperature, despite that there has not yet been a decrease in the indoor temperature. By means of this feed forward, the variations in the indoor temperature are considerably reduced.

It is central to the efficiency of the feed forward how early in the loop the disturbance can be measured. If this can be done long before the disturbance is seen in the measurement signal, the feed forward can be made more efficient. This is especially true if the process contains a long delay. If we can only measure the disturbance when it has already affected the process output, there is generally no benefit in implementing a feed forward connection. Here one might just as well let the controller handle the disturbance by means of feedback.

We illustrate the principle of feed forward and how to calculate a feed forward link by means of an example.

Example 13.2—Feed forward of disturbance flows in the tank process
Figure 13.4 shows a tank process consisting of two tanks. Water is pumped into the upper tank and the objective is to control the level in the lower tank. In this example we shall study two types of disturbances. In the first case we assume that

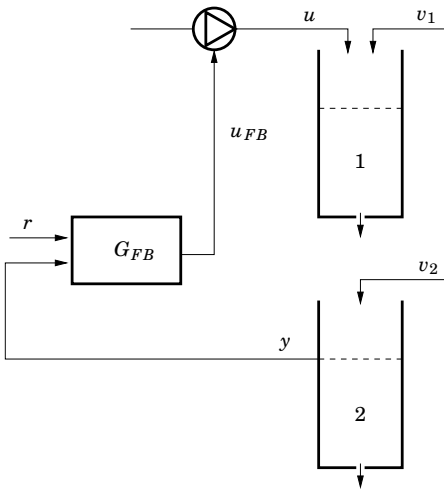

Figure 13.4 Feed forward in Example 13.2.

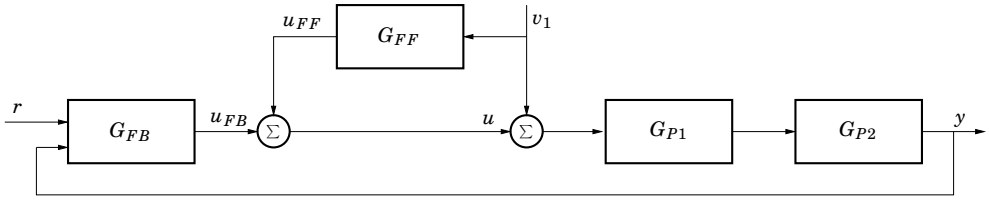

Figure 13.5 Feed forward in Example 13.2 when the flow disturbance v_1 enters the upper tank.

a disturbance flow v_1 enters the upper tank and in the second case we have a disturbance flow v_2 entering the lower tank.

Obviously, one does not need to treat these disturbances explicitly, since they are indirectly counteracted by the feedback control. If the flows are measurable we can, however, improve the control performance significantly by introducing feed forward.

We begin by studying the first case, when the flow disturbance enters the upper tank. The control problem is described by the block diagram shown in Figure 13.5.

The transfer function G_{P1} describes the relation between inflow and outflow in the upper tank, while G_{P2} describes the relation between the inflow and the level in the lower tank. We have here disregarded the pump dynamics and assumed that the control signal u is given by the controlled flow. The usual feedback part of the controller is given by the transfer function G_{FB}, which consequently determines the feedback term u_{FB} of the control signal based on the setpoint r and the measurement y.

Finally, we have the disturbance flow v_1. Since it enters the process at the same point as the controlled flow, it is added to the process input. The disturbance signal v_1 is sent to the feedforward part of the controller G_{FF}, resulting in the feedforward term u_{FF}. The control signal is thus given by

$$u = u_{FB} + u_{FF}$$

In this example it is obvious how the feed forward part of the controller can be determined. If we choose $G_{FF} = -1$ so that

$$U_{FF} = G_{FF}V_1 = -V_1$$

the disturbance flow will never give rise to a change in the level y. It is also obvious from figure 13.4 that a change in the disturbance flow v_1 gives rise to an equally large change in the controlled flow u, but with the opposite sign.

We will now move on to study the problem when the disturbance flow v_2 enters the lower tank. The block diagram of the problem is shown in Figure 13.6.

The optimal feed forward is one which makes disturbance variations invisible in the measurement signal. We can thus compute the optimal feedforward transfer function G_{FF} by computing the transfer function from v_2 to y and choosing G_{FF} so that this transfer function becomes zero. However, we do not need to do all this. It is sufficient to consider the transfer function from v_2 to the signal x in Figure 13.6. The signal x corresponds to the inflow of the lower tank.

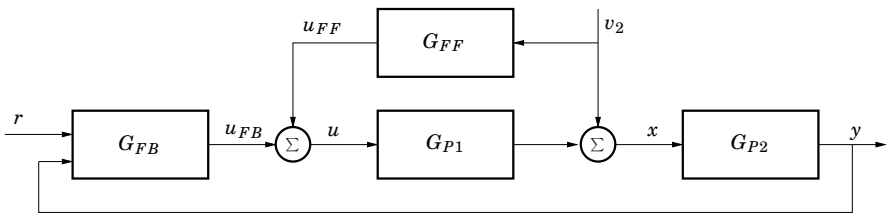

Figure 13.6 The feed forward in Example 13.2, where the disturbance flow enters the lower tank.

The transfer function from v_2 to x is obtained from

$$X = V_2 + G_{P1}(U_{FB} + U_{FF}) = G_{P1}U_{FB} + (1 + G_{P1}G_{FF})V_2$$

This yields the optimal feed forward

$$G_{FF} = -\frac{1}{G_{P1}}$$

which results in the feed forward term

$$U_{FF} = G_{FF}V_2 = -\frac{1}{G_{P1}}V_2$$

Most processes have low-frequency characteristics, which means that their gains decrease for high frequencies. It is thus common for process transfer functions to have more poles than zeros. A previously used model of a tank is e.g. given by

$$G_{P1}(s) = \frac{1}{s + a}$$

When computing the optimal feedforward transfer function G_{FF}, one always obtains an expression, which involves inverting the part of the process dynamics situated between the control signal u and the place where the disturbance enters the process. In the case of the disturbance v_1 this was no problem, since there existed no such dynamics. This was the case since the disturbance entered the process in the same place as the control signal. In the actual case, however, the following feed forward terms is obtained:

$$U_{FF} = -\frac{1}{G_{P1}}V_2 = -(s + a)V_2$$

It means that we have to derive the disturbance flow v_2. This is commonly avoided, since it gives rise to disturbance sensitivity and large variations in the control signal. Either one can extend the feed forward with a low pass filter or, which is more common, one can skip the derived terms and suffice with the static feed forward

$$u_{FF} = -av_2$$

□

13.3 Delay Compensation

The derivative part of the PID controller is used to predict future values of the measurement signal. This is done by studying its derivative. Evidently this method works poorly if the process involves a long delay. In this case one could of course use a PI controller, with the loss of the prediction provided by the D part. This is, however, a significant disadvantage, since the prediction of future control errors is especially useful when we have long process delays. As a consequence special controllers have been developed, which can predict future control errors also in processes with long delays. These predictions are, however, not based on deriving the measurement signal.

The principle of delay compensation controllers is to construct the control error prediction based on the control signal, rather than the measurement signal. By devising a model of the process to be controlled and letting the control signal drive both the real process and the model, one can obtain near-future values of the measurement signal of the process by studying the measurement signal of its simulation. The most common delay compensation strategy is implemented in the Smith

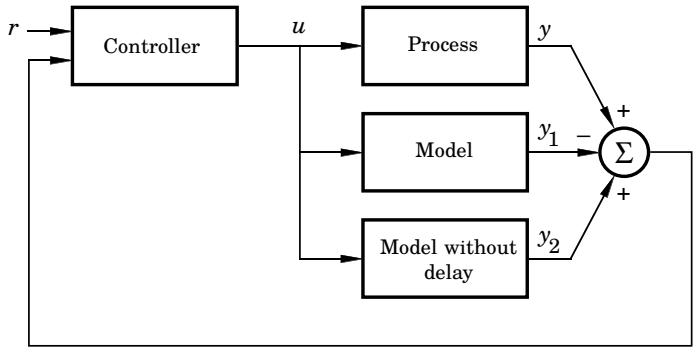

Figure 13.7 The working principle of the Smith predictor.

predictor. A schematic sketch, illustrating the working principle of this strategy is shown in Figure 13.7.

Apart from the usual controller parameters, the Smith predictor needs a model of the process. Especially, it needs to know the length of the process delay. The figure shows how the control signal enters the process as well as models of the process, one with and one without the estimated process delay.

Let us assume that the model is an accurate deception of the process. The two signals y and y_1 will then be identical and therefore cancel each other. The remaining signal entering the controller is y_2, i.e. the signal we would have obtained if there was no delay in the process. This way the controller works against a simulated process, identical to the real process, with its delay removed. The control performance becomes as good as it would have been without the delay, except the fact that the measurement signal is obviously still delayed.

In reality the model is, however, not a perfect description of the process. Consequently, the signal going back into the controller is not identical to y_2. This usually calls for a more conservative tuning of the controller than one would have chosen if the delay was not present.

We shall now give an example of control using the Smith predictor.

Example 13.3—The Smith predictor

Figure 13.8 shows the control of a first order process with a delay. The figure shows the control performance of both the PI controller and the Smith counterpart.

The PI controller is tuned to provide fast control action at step changes in the setpoint, without overshoots. At the setpoint change the control signal steadily starts to integrate the control error. The integration is slow enough not to cause any overshoot.

The delay compensation controller is also tuned not to cause any overshoot. It responds to setpoint changes and load disturbances significantly faster than the PI controller. We can especially notice a crucial improvement at setpoint changes, as compared to the PI case. This is because the Smith predictor issues a control signal change before the setpoint change is visible in the measurement signal.

At load disturbances none of the controllers can react before the disturbance is visible in the measurement signal. The only remedy here would be to introduce a feedforward link from the disturbance. □

We shall now conduct a closer study of the Smith predictor. The controller described by Figure 13.7 is equivalent to the block diagram shown in Figure 13.9.

The transfer function of the process is given by

$$G_P(s) = G_{P0}(s)e^{-sL}$$

The estimated process is correspondingly described by the transfer function

$$\hat{G}_P(s) = \hat{G}_{P0}(s)e^{-s\hat{L}}$$

The controller G_{R0}, which is part of the Smith controller, is often a PI controller.

From the block diagram in Figure 13.9 the control signal of the Smith controller can be computed as

$$\begin{aligned} U &= G_{R0}(R - Y + Y_1 - Y_2) \\ &= G_{R0}(R - Y + \hat{G}_{P0}e^{-s\hat{L}}U - \hat{G}_{P0}U) \\ &= G_{R0}(R - Y) + G_{R0}\hat{G}_{P0}(e^{-s\hat{L}} - 1)U \end{aligned}$$

If G_{R0} is a PI controller, we see that the Smith controller can be considered as a PI controller with an added term, which is driven by the control signal u. This is the term responsible for the prediction. In other words we have replaced the measurement signal derivative based prediction in the PID controller with a prediction based on the process model $\hat{G}_P$ and the control signal.

If the model is identical to the real process, i.e. if $\hat{G}_P = G_P$ and if we do not have any process disturbances, the signals y and y_1 will be identical. The only signal re-entering the controller G_{R0} will then be y_2. This yields the control signal

$$U = G_{R0}(R - Y_2) = G_{R0}(R - G_{P0}U) = \frac{G_{R0}}{1 + G_{R0}G_{P0}}R$$

Subsequently, the relation between the setpoint and measurement signal is given by

$$\frac{Y}{R} = \frac{G_P G_{R0}}{1 + G_{P0}G_{R0}} = \frac{G_{P0}G_{R0}}{1 + G_{P0}G_{R0}}e^{-sL}$$

This means that we have a transfer function identical to the case of a process without delay, $G_P = G_{P0}$, except from the fact that the measurement signal is delayed by a time L. Ideally this means that the controller G_{R0} can be determined as if there was no process delay. In reality, however, modelling errors and robustness margins force us to tune the controller more conservatively. In Lecture 6 we motivated that the cross-over frequency should not be chosen faster than $\omega_c L < 0.2$, where L is the length of the process delay.

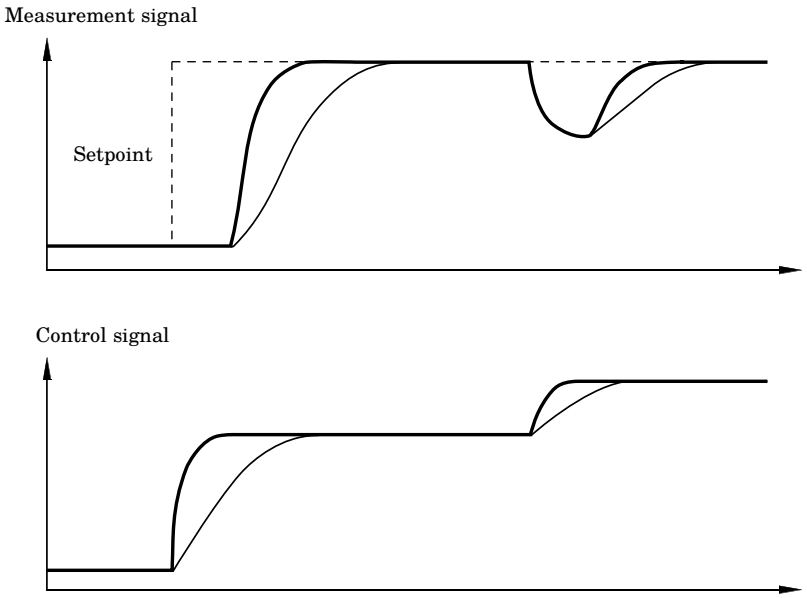

Figure 13.8 Comparison between the Smith controller (thick lines) and a PI controller (thin lines). The figure shows the control performance when a setpoint change is followed by a load disturbance.

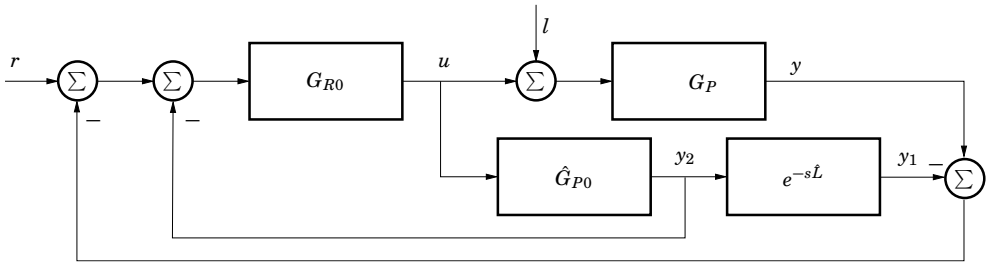

Figure 13.9 The Smith predictor.

13.4 Sampling

In the 30-ies and 40-ies, when controller manufacturers noticed an upswing due to the ongoing industrialization, most controllers were implemented using pneumatic technology. It did not take long before analog electronic controllers were starting to replace the pneumatic ones. In the middle of the 70-ies another paradigm shift took place. The analog technology was replaced with digital circuitry. Today pneumatic controllers are still used in environments where it is not possible or desirable to introduce electricity. Other than that, mostly all controllers are digital and computer based.

The fact that almost all controllers are implemented in computers introduces a whole set of new problems related to how signals are treated. The analog inputs are discretized by an A/D converter and the digital outputs are converted to analog signals using a D/A converter.

Figure 13.10 shows an analog measurement signal and its sampled realization. By sampling a signal, information is lost. The controller obtains no information regarding the measurement signal between the sampling events. It is therefore important that the sampling frequency is adequately high.

If the sampling frequency is not adequate, a phenomenon known as frequency folding or aliasing can occur. The problem is illustrated by Figure 13.11.

The figure shows the signal s with a period of just above a second, which has been sampled with a period of one second. Due to the relatively low sampling frequency, the controller will not obtain the signal s, but rather the signal s_a, which has a significantly lower frequency. The signal s_a is known as an alias of the signal s. (It is this effect which causes wheels with spokes and airplane propellers to appear as if they stand still or move backwards in movies.)

The sampling frequency introduces an upper bound on how high frequencies are visible to the controller. In Figure 13.11 we see that this frequency, known as the

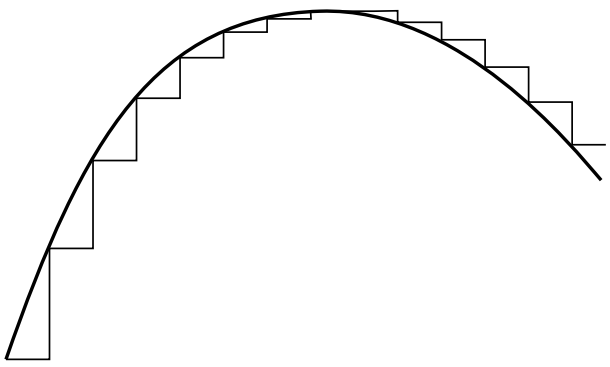

Figure 13.10 An analog signal and its sampled realization.

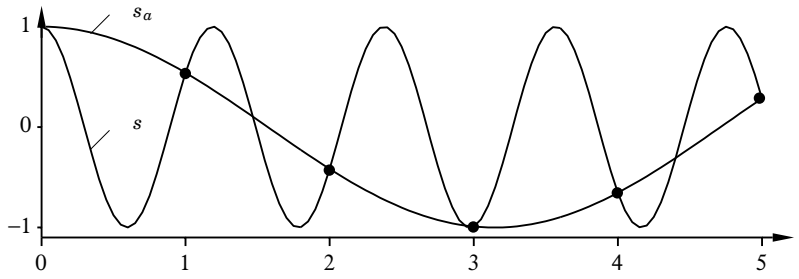

Figure 13.11 Illustration of the folding phenomenon. The figure shows the actual signal s and its alias s_a.

Nyquist frequency, is given by

$$\omega_N = \frac{\omega_s}{2}$$

where ω_s is the sampling frequency. In other words we must keep the sampling frequency at least twice as high as the highest frequency relevant to the controller.

Even if we choose the sampling frequency adequately high to represent all frequencies of interest in the system, the problem of high-frequent noise remains. It is therefore important that the analog signals are low-pass filtered *before* they enter the A/D converter, so that frequency components above the Nyquist frequency are eliminated. If this is not done, there exists a risk that high-frequent disturbances show up as low-frequent ones in the controller due to aliasing.

13.5 Discretization of the PID Controller

If one wants to implement a continuous controller, e.g. a PID controller, in a computer, one somehow has to approximate the derivations and integrations. This can be readily done by approximating the derivatives by differences and integrations by summations. We shall now see how this discretization is carried out.

We begin with the analog PID controller, given by the Laplace transformation

$$U = K\left((bR - Y) + \frac{1}{sT_i}E - \frac{sT_d}{1 + sT_d/N}Y\right)$$

This is the version we arrived at in the previous lecture, after the practical modifications of the standard form.

The discrete control signal is given by

$$u(kh) = P(kh) + I(kh) + D(kh)$$

i.e. a sum of the proportional, integral and derivative parts. The sampling interval is denoted h and k is an integer. The discrete control signal $u(kh)$ can now be described as a function of the controller inputs r and y at the sampling instants $t = kh,\ (k-1)h,\ (k-2)h,\ \ldots$. We do this for one term at a time.

Discretization of the P Part

The continuous P part is given by

$$u_P(t) = K(br(t) - y(t))$$

Since the P part does not contain any dynamics, the discretized version simply becomes

$$P(kh) = K(br(kh) - y(kh))$$

Discretization of the I Part

The continuous integral term is given by

$$u_I(t) = \frac{K}{T_i} \int_0^t e(t)dt$$

Here we have to approximate the integral and do so by replacing it with a sum.

$$I(kh) = \frac{K}{T_i} \cdot h \sum_{i=1}^{k} e(ih)$$

A more useful form of the integral term is the recursive form

$$I(kh) = I(kh-h) + \frac{Kh}{T_i} e(kh)$$

Discretization of the D Part

From the transfer function of the PID controller we get the following equation for the derivative term:

$$\left(1 + \frac{sT_d}{N}\right) U_D = -KT_d sY$$

Inverse transformation provides the following expression for the derivative term:

$$u_D(t) + \frac{T_d}{N} \frac{du_D(t)}{dt} = -KT_d \frac{dy(t)}{dt}$$

Here we have to approximate the derivatives by replacing them by differences.

$$D(kh) + \frac{T_d}{N} \frac{D(kh) - D(kh-h)}{h} = -KT_d \frac{y(kh) - y(kh-h)}{h}$$

The derivative term thus becomes

$$D(kh) = \frac{T_d}{T_d + Nh} D(kh-h) - \frac{KT_d N}{T_d + Nh} (y(kh) - y(kh-h))$$

The obtained discrete PID controller will work just as its continuous counterpart given that the sampling period is short in comparison to the dominant time constant of the closed-loop system. A rule of thumb is to choose the sampling period short enough to allow ten samples during one time constant.

Lecture 14

Example: The Ball on the Beam

This lecture is dedicated to solving a concrete control problem. The objective is to illustrate a typical approach used in solving control problems. We thus begin with a modelling part, in which we obtain a mathematical description of the processes dynamics. Subsequently we analyze the control problem and discuss different possible solutions. Finally we choose a strategy and conduct the calculations leading to a controller.

14.1 Model Building

The first task is to understand the process and describe its dynamical properties mathematically. We begin by describing the process and its functionality.

Process Description

The process is a laboratory process—"The ball on the beam". It is shown in Figure 14.1.

The process can be divided into three parts; a ball, a beam with a track in which the ball can roll, and an electric motor, turning the beam. The aim of the control is to steer the inclination of the beam via the motor so that the ball tracks a position setpoint.

The process has one control signal, being the voltage over the motor. There exist two measurement signals; the angle of the beam with respect to the horizontal plane and the position of the ball on the beam. The position of the ball is measured by means of a resistive wire, running along the edge of the beam.

Figure 14.1 The ball on the beam.

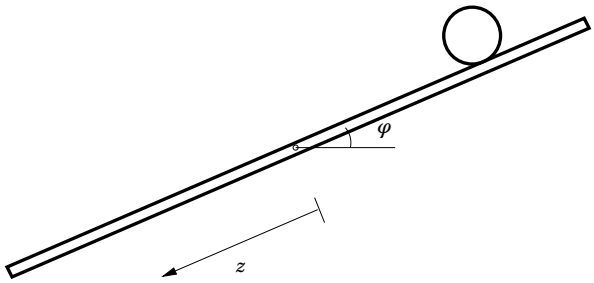

Figure 14.2 Schematic description of the beam process with measurement signals φ and z.

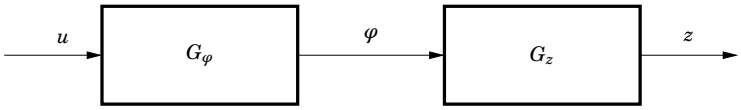

Figure 14.3 Block diagram of the ball on the beam process.

A schematic description of the process is shown in Figure 14.2, where the measurement signals—beam angle φ and ball position z—are both marked. Since there exist two measurement signals, the transfer function of the process can be divided into two parts as shown in Figure 14.3. The first partial transfer function G_φ describes the dynamics between control signal u and angle φ. The second partial transfer function G_z describes the relation between beam angle φ and ball position z. In the modelling part we will consider these two parts independently.

The Dynamics between Control Signal and Beam Angle

The motor contains an internal feedback connection, making the control signal proportional to the angular speed of the beam, i.e.

$$\dot{\varphi} \approx k_\varphi u$$

where k_φ is a constant. It is easy to convince oneself of this by conducting step response experiments on the process. Since it is an integrating process, there exist only one equilibrium for the control signal, as the beam is still. If a change is made in the control signal from this equilibrium, the beam will rotate with a constant angular speed. Gain k_φ can be experimentally determined by e.g. studying the step response or by frequency analysis.

Figure 14.4 shows the result of a frequency analysis conducted on the beam.

The Bode plot shows that the low-frequency process dynamics can be well approximated by an integrator, i.e. that the transfer function

$$G_\varphi(s) = \frac{k_\varphi}{s}$$

is a good model of the process. One can also see that $|G_\varphi(i\omega)| = 1$ for $\omega \approx 4.5$, which yields the gain $k_\varphi \approx 4.5$. The transfer function G_φ thus becomes

$$G_\varphi(s) = \frac{4.5}{s}$$

The Bode plot also shows that this simple model is only valid for low frequencies. For higher frequencies other dynamics starts to influence the system. For frequencies up to $\omega \approx 10$ rad/s the gain curve matches the model well while the phase curve has decreased from the model value of $-90°$ to approximately $-100°$. It is very important to keep the validity range of the model in mind. If we for some reason want to work in the higher frequency range further on, we must obtain a new model describing the process also for these frequencies.

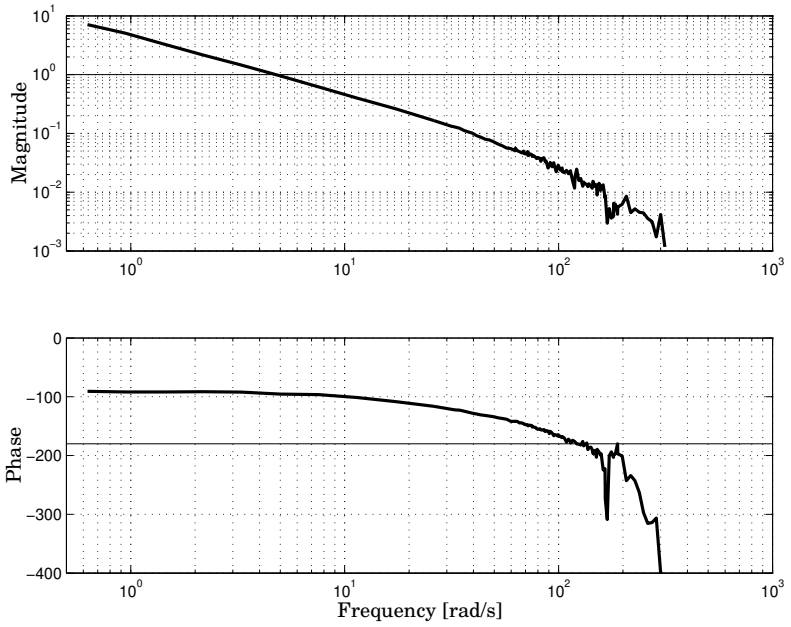

Figure 14.4 Bode plot of the angle process.

The Dynamics between the Angle and the Position of the Ball

The relation between the ball position z and the beam angle φ can be calculated using mechanical laws. When the beam is inclined, a momentum causes the ball to roll. This is illustrated in Figure 14.5. There is also a centrifugal force acting on the ball, but this force is neglected. The momentum of the ball is given by

$$M = mgr \sin \varphi$$

where m is the mass of the ball and r is the ball radius. The momentum of inertia of the ball with respect to the contact surface is obtained from a table and given by

$$J = \frac{7mr^2}{5}$$

From the momentum equation

$$M = J\dot{\omega}$$

we can now compute the angular acceleration of the ball:

$$\dot{\omega} = \frac{M}{J} = \frac{5g \sin \varphi}{7r}$$

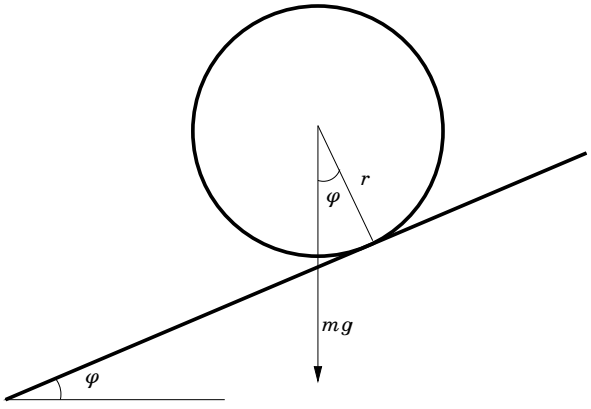

Figure 14.5 The ball on the beam.

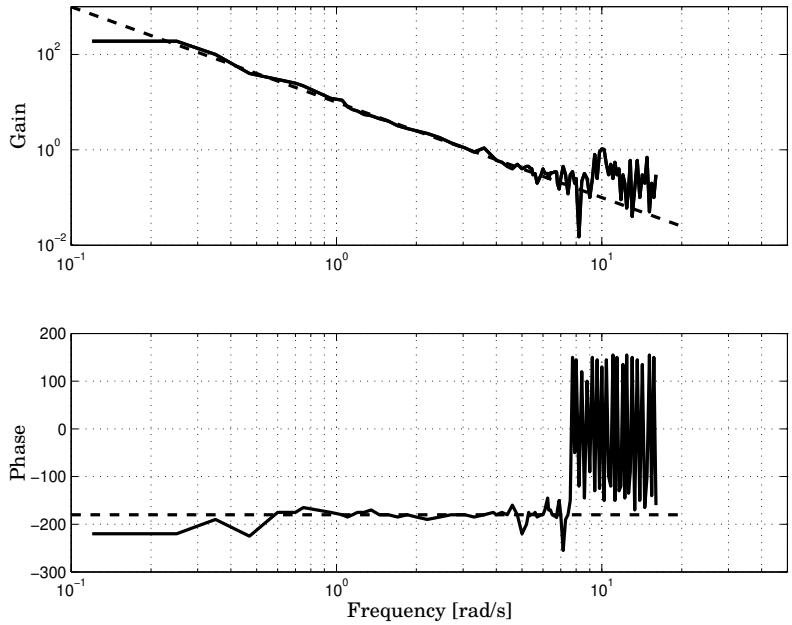

Figure 14.6 Bode plot of the position process. The dashed curves correspond to the model $G_z = 10/s^2$.

This equation yields the relation between φ and ω. However, we are interested in the relation between φ and z. Since

$$\dot{z} = r\omega$$

this relation is given by

$$\ddot{z} = r\dot{\omega} = \frac{5g\sin\varphi}{7} \approx 7\sin\varphi \approx 7\varphi$$

The last approximation assumes small angular variations.

Using the laws of mechanics we have thus arrived at the transfer function

$$G_z = \frac{7}{s^2}$$

i.e. a double integrator with gain 7. This transfer function describes the relation between the angle φ, measured in radians, and the position z, measured in meters. What we still lack in the transfer function is the conversion factors to volts of the sensors. Figure 14.6 shows the result of a frequency analysis conducted on the beam. The Bode plot confirms that the double integrator is an adequate model of the process. The gain can be determined by e.g. noting that the gain is approximately one at the frequency $\omega = 3.2$. This gives

$$|G_z(3.2i)| = \frac{k_z}{3.2^2} \approx \frac{k_z}{10} = 1$$

which means that the gain is approximately 10. The transfer function thus becomes

$$G_z = \frac{10}{s^2}$$

The figure shows that the model is accurate for frequencies up to $\omega \approx 5$ rad/s. Above this frequency the model is no longer a good representation of the model.

A problem, which is not covered by the model, is that the ball jumps at too fast angular changes, especially if it lies far away from the pivot point of the beam. The model does not take the centrifugal force on the ball into account either.

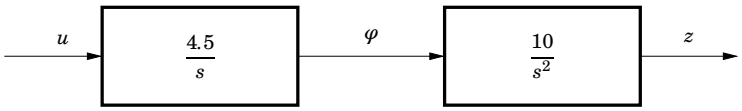

Figure 14.7 Block diagram of the process.

Summary

The obtained model of the process is shown in Figure 14.7. The first transfer function, G_φ, is valid for frequencies up to $\omega \approx 10$ rad/s while the second transfer function, G_z, is only valid up to $\omega \approx 5$ rad/s.

14.2 Analysis

The process to be controlled has now been modelled as shown in Figure 14.7. The transfer function from control signal u to ball position z is given by

$$G_P = G_z G_\varphi = \frac{45}{s^3}$$

The process is i.e. well described by a triple integrator and thus unstable. Its Bode plot has a gain curve with slope -3 and a phase curve showing a constant $-270°$ phase for low frequencies.

We have introduced several control strategies throughout the course and we shall now discuss some alternative ways to control the ball on the beam.

Can the process be controlled using a PID controller? The answer is no. Since the process has a phase shift of $-270°$, a controller with a phase lead of more than $90°$ is required to obtain a positive phase margin and thus stabilize the process. The PID controller can theoretically achieve a maximal phase lead of $90°$ using the D part. However, with a filter on the D part, one will never reach $90°$.

One possible solution is to combine the PID controller with lead compensation and in this way obtain a phase advance larger than $90°$. Since we have access to two measurement signals, better solutions can be obtained by using both of them. One approach is to use two PID controllers cascaded according to Figure 14.8. The inner PID controller, PID2, can be used to move the integrator of G_φ into the left half plane. This results in a phase increase for low frequencies from $-90°$ to $0°$.

The outer PID controller, PID1, does no longer control the entire process G_P, but rather a process consisting of two integrators yielding a phase shift of $-180°$ for low frequencies. As opposed to G_P, this process allows for PID control.

An other approach would be to conduct the controller calculations in the frequency domain, utilizing lead-/lag compensation links. The lead link, which we studied previously in the course, cannot stabilize G_P on its own. However, one can replace the controller PID1, in the cascade solution above, with a lead link.

A third approach would be state feedback. Since the process is of order three, this requires at least three states to close the loop over. We only have two available states, but through Kalman filtering, a third state can be estimated.

We choose to solve the control problem using two cascaded PID controllers according to Figure 14.8. When we are done we will study the solution and show that the obtained controller could just as well have been the result of other synthesis methods.

14.3 Control

We have now arrived at the actual controller synthesis part. We shall divide it into two parts. First we determine the secondary controller, PID2, in the cascade

configuration shown in Figure 14.8. This is done in a way yielding good control performance of the beam angle. Subsequently we determine the primary controller PID1 so that good control performance of the ball position with respect to the beam is obtained.

Angle Control

The transfer function of the angle control part of the process is given by

$$G_\varphi = \frac{4.5}{s}$$

We shall control this part of the process using a PID controller. Since the process dynamics are uncomplicated, it will in fact suffice to use the P controller

$$G_{R2} = K_2$$

The closed-loop transfer function becomes

$$G_2 = \frac{4.5K_2}{s + 4.5K_2} = \frac{1}{1 + sT_2}$$

where $T_2 = 1/(4.5K_2)$ is the time constant of the closed-loop system. Thanks to the integrator of the process, the stationary gain is given by $G_2(0) = 1$. This means that setpoint steps do not result in persisting control errors. Since this part of the process does not involve any load disturbances, we do not need to introduce integral action in the controller.

Theoretically the time constant T_2 of the closed-loop system can be chosen arbitrarily short without causing stability issues. The Bode plot in Figure 14.4, however, shows that the model is uncertain for frequencies above 10 rad/s. A cross-over frequency of 10 rad/s means that

$$|G_0(i\omega_c)| = |G_0(i \cdot 10)| = K_2 \frac{4.5}{10} = 1$$

i.e.

$$G_{R2} = K_2 = \frac{10}{4.5} \approx 2.2$$

This yields the closed-loop time constant $T_2 = 1/(2.2 \cdot 4.5) \approx 0.10$ s. The resulting speed is adequate and does not manifest any major limitations when we move on to the part of the control problem involving the position of the ball with respect to the beam.

Ball Position Control

We now move on to the synthesis of the primary controller. Thanks to the cascading and the secondary controller, the process G_P involving at least three integrators has been simplified into

$$G_{P1} = \frac{1}{1 + 0.1s} \cdot \frac{10}{s^2}$$

This transfer function has the low-frequency phase $-180°$ and can be stabilized by means of the D part of the PID controller.

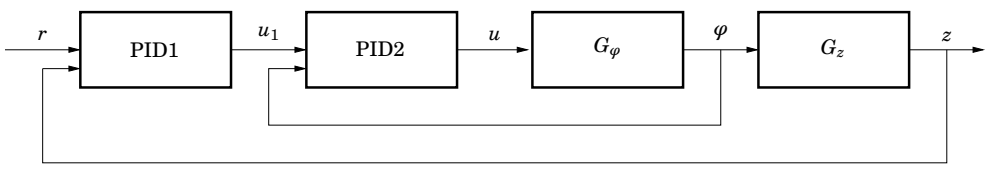

Figure 14.8 Cascade control of the beam process.

There are many ways to determine the controller parameters. Since the process has third-order dynamics, the poles cannot be arbitrarily placed. We therefore choose the controller parameters aided by the Bode plot. The design is simplified if we assume that the controller is given in the series form

$$G'_{R1} = K'\left(1 + \frac{1}{sT'_i}\right)\frac{1 + sT'_d}{1 + sT'_d/10}$$

From the equations it is evident that the controller contains a filter with time constant $T'_d/10$ on its derivative part.

If we compare the equation of G'_{R1} to the lead–lag links studied in Lecture 11, we see that the controller is a product of a lag link and a lead link. The lag link is given by

$$1 + \frac{1}{sT'_i} = \frac{sT'_i + 1}{sT'_i} = \frac{s + 1/T'_i}{s} = \frac{s + a}{s + a/M}$$

where $a = 1/T'_i$ and $M \to \infty$. The lead link is given by

$$K'\frac{1 + sT'_d}{1 + sT'_d/10} = K'\frac{1 + s/b}{1 + s/(bN)}$$

where $b = 1/T'_d$ and $N = 10$.

We can now determine the controller parameters by means of the methods from Lecture 11. By determining the parameters of the lead link, we determine the derivative time T'_d and the gain K'. Thereafter we determine the integral time T'_i from the parameter a of the lag link.

Lead Compensation The first step in the lead compensation is to give specifications on the cross-over frequency and the phase margin. In this case, it is not easy to find reasonable specifications, since the process has such a large phase lag. The phase of G_{P1} goes from $-180°$ at low frequencies to $-270°$ at high frequencies.

Suppose that we would like to have a phase margin $\varphi_m = 40°$. It means that the compensator must provide a phase advance of more than $40°$. If we choose $N = 10$, Figure 11.11 shows that the lead compensator gives a phase advance of $55°$ at the cross-over frequency. With the rule of thumb $a = 0.1\omega_c$, the lag compensator gives a phase decrease of $-6°$ at the cross-over frequency. This means that the combination of the compensators gives a phase advance of $49°$ and we can choose a cross-over frequency where the process has the phase $-189°$, i.e.

$$\arg G_{P1}(i\omega_c) = -180° - \arctan(0.10\omega_c) = -189°$$

This gives the cross-over frequency $\omega_c \approx 2$ rad/s.

After choosing ω_c we can now determine T'_d and K' according to the method described in Lecture 11. The derivative time T'_d is determined first, so that the maximal phase lead occurs at the cross-over frequency.

$$b\sqrt{N} = \omega_c \Rightarrow T'_d = \frac{1}{b} = \frac{\sqrt{N}}{\omega_c} = \frac{\sqrt{10}}{2} = 1.6$$

Subsequently K' is chosen so that ω_c really becomes the cross-over frequency, i.e. so that the magnitude of the open loop transfer function is unity at ω_c. This yields the equation

$$|G'_{R1}(i\omega_c)G_{P1}(i\omega_c)| = K'\sqrt{N}\cdot\frac{1}{\sqrt{1 + \omega_c^2 T_2^2}}\cdot\frac{10}{\omega_c^2} = 1$$

where we have exploited that the magnitude of the compensation link is $K'\sqrt{N}$ at the cross-over frequency, cf. Lecture 11. We have also neglected the lag component, since it has a gain close to one at the cross-over frequency. The equation now gives us the gain

$$K' = \frac{\omega_c^2\sqrt{1 + \omega_c^2 T_2^2}}{10\sqrt{10}} = \frac{2^2\sqrt{1 + 2^2 0.10^2}}{10\sqrt{10}} = 0.13$$

Lag Compensation Finally we determine the integral time T_i', using the same rule of thumb as we learnt in Lecture 11. The integral time is chosen so that the phase at the cross-over frequency is not decreased by more than 6°. This gives us the equation

$$a = 0.1\omega_c \Rightarrow T_i' = \frac{1}{a} = \frac{1}{0.1\omega_c} = \frac{1}{0.1 \cdot 2} = 5.0$$

Result We have now determined the parameter sets for the two cascaded controllers. The obtained parameters are

$$\begin{aligned} G_{R1}: &\quad K' = 0.13 \quad T_i' = 5.0 \quad T_d' = 1.6 \\ G_{R2}: &\quad K_2 = 2.2 \end{aligned}$$

Figure 14.9 shows a Bode plot of the loop transfer function $G_{R1}G_{P1}$. It shows that we have obtained the desired cross-over frequency $\omega_c = 2$ rad/s and phase margin $\varphi_m = 38°$.

Figure 14.10 shows the result of a simulation of the beam process, utilizing the computed controllers. It first shows the response to a setpoint step. At $t = 20$ a load disturbance is introduced. The load disturbance corresponds to a suddenly applied pressure on the ball.

The Bode plot and the simulation shows that the obtained controller is reasonable. However, the real process involved nonlinearities and unmodeled dynamics and the implementation of the controller leads to delays and necessitates filters, which have all been omitted in the above calculations. Hence we should expect deviations from the simulation. Nevertheless the experiment shows us that these deviations are small and that the computed controller works well on the real process.

Other Approaches

We have now solved the control problem and chose to do so using two cascaded PID controllers. As mentioned earlier, we could have used other approaches. We shall now round off by investigating the obtained controller and showing that it could well have been obtained through other methods. For example, it could have been the result of a state feedback from Kalman filtered state estimates. This is illustrated in Figure 14.11. The block diagram shows a state feedback with integration where the

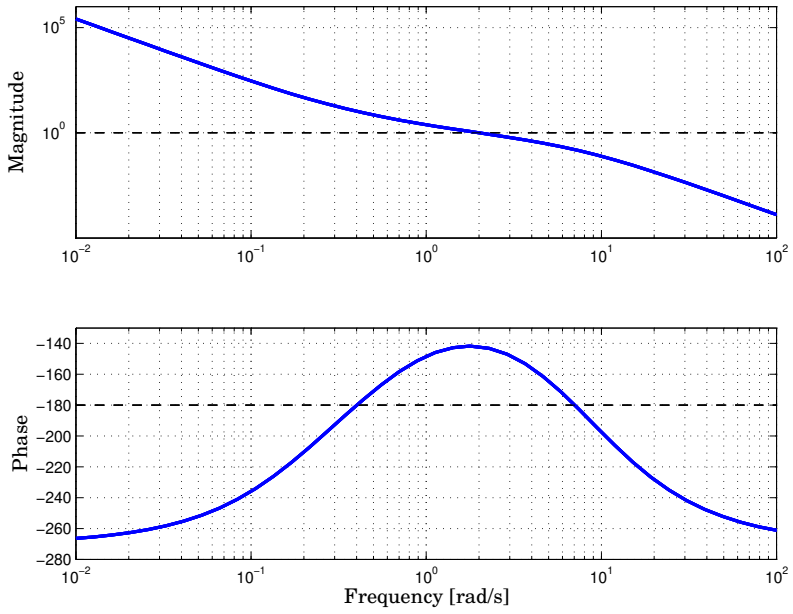

Figure 14.9 Bode plot of the open-loop transfer function $G_{R1}G_{P1}$.

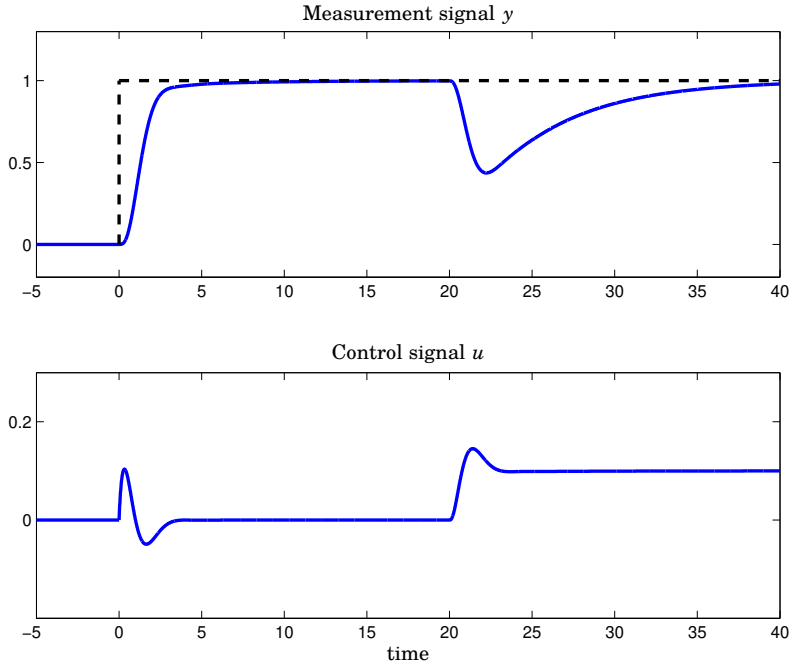

Figure 14.10 The result of a simulation of the beam process. The upper curve shows the ball position at setpoint and load steps. The lower curve shows the corresponding control signal.

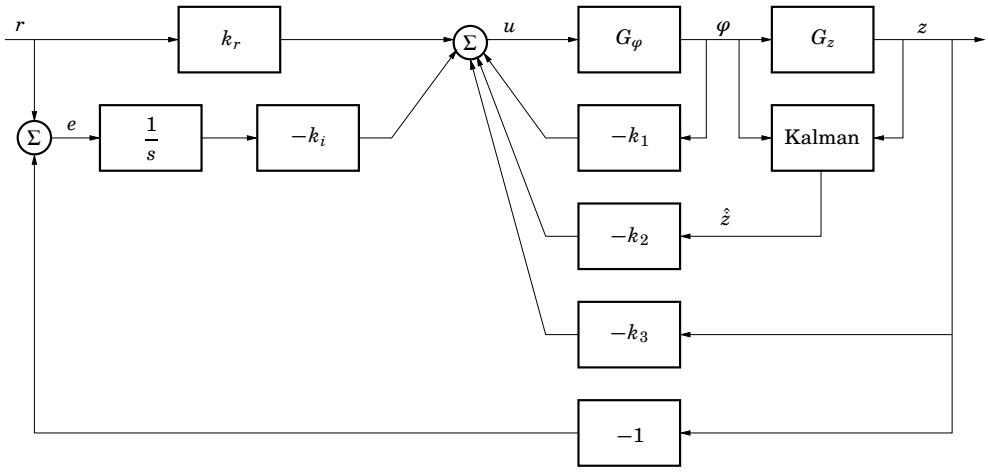

Figure 14.11 Cascade control of the beam process, interpreted as state feedback from Kalman filtered state estimates.

speed with respect to the beam, $\dot{z}$, is estimated by the Kalman filter. If we disregard the filter on the derivative part of the PID controller, it is possible to show that the control in Figure 14.11 is identical to the two cascaded PID controllers, given the following choice of parameters

$$k_1 = K_2 \qquad k_2 = KK_2T_d \qquad k_3 = KK_2$$
$$k_r = KK_2b \qquad k_i = -\frac{KK_2}{T_i}$$

Here, G_{R1} is given in parallel form. The only difference is that the position of the ball is estimated by the derivative of the measurement signal in the PID case, while it is estimated by means of a Kalman filter in Figure 14.11.

It would have been possible to obtain the PID parameters through pole placement, even though we would not have been able to place the outer loop poles arbitrarily.

We could also have utilized other PID tuning methods.

By this final section we have concluded that the different methods of controller synthesis, which we have studied, might very well result in the same controller. The difference between the methods thus lies in their computational methodology, rather than the resulting controller.

Index