

# **Introduction to Feedback Control**

Christopher Nielsen  
Control Systems Group  
Department of Electrical & Computer Engineering



## Acknowledgments

These notes are for introductory undergraduate courses on control systems at the University of Waterloo. They are partially based on notes written by Prof. John Thistle, Prof. Daniel Miller, Prof. Daniel E. Davison at the University of Waterloo and Prof. Bruce Francis, Prof. Manfredi Maggiore, Prof. Raymond Kwong at the University of Toronto and Prof. Andrew Lewis at Queen's University. All errors, typographical or otherwise, are solely mine.

---

# Contents

<b>Notation</b>	<b>iv</b>
<b>Acronyms and Initialisms</b>	<b>v</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Familiar examples . . . . .	1
1.2 What is control engineering? . . . . .	2
1.3 Control engineering design cycle . . . . .	6
1.4 Feedback control in software and computer systems . . . . .	8
1.5 Very brief history . . . . .	9
1.6 Notation . . . . .	10
1.7 Summary . . . . .	10
<b>2 Mathematical models of systems</b>	<b>12</b>
2.1 General comments on modelling . . . . .	12
2.2 Block diagrams . . . . .	15
2.3 Modelling based on first principles . . . . .	17
2.4 State-space models . . . . .	28
2.5 Linearization . . . . .	34
2.6 Simulation . . . . .	39
2.7 The Laplace transform . . . . .	40
2.8 Transfer functions . . . . .	47
2.9 Block diagram manipulations . . . . .	51
2.10 Summary . . . . .	56
2.A Appendix: Linear functions . . . . .	56
2.B Appendix: The impulse and convolution . . . . .	57
<b>3 Linear system theory</b>	<b>60</b>
3.1 Initial-state response . . . . .	60
3.2 Input-response . . . . .	63
3.3 Total response . . . . .	65
3.4 Stability of state-space models . . . . .	66
3.5 Bounded-input bounded-output stability . . . . .	68
3.6 Steady-state gain . . . . .	73
3.7 Frequency response . . . . .	74
3.8 Graphical representations of the frequency response . . . . .	76
3.9 Summary . . . . .	90
3.A Appendix: Complex numbers . . . . .	90
3.B Appendix: Time domain, s-domain and frequency domain . . . . .	91

<b>4 First and second order systems</b>	<b>93</b>
4.1 Prototype first order system . . . . .	93
4.2 Prototype second order system . . . . .	97
4.3 General characteristics of a step response . . . . .	104
4.4 The effect of adding poles and zeros . . . . .	108
4.5 Dominant poles and zeros . . . . .	111
4.6 Summary . . . . .	113
<b>5 Feedback control theory</b>	<b>114</b>
5.1 Closing the loop . . . . .	114
5.2 Stability of feedback systems . . . . .	118
5.3 The Routh-Hurwitz criterion . . . . .	125
5.4 Steady-state performance . . . . .	129
5.5 Summary . . . . .	135
<b>6 Root-locus method</b>	<b>137</b>
6.1 Basic root-locus construction . . . . .	137
6.2 Examples . . . . .	141
6.3 Non-standard problems . . . . .	146
6.4 Summary . . . . .	149
<b>7 PID control</b>	<b>150</b>
7.1 Classical PID controller . . . . .	150
7.2 Empirical tuning . . . . .	157
7.3 Pole placement . . . . .	159
7.4 Integrator windup . . . . .	162
7.5 Summary . . . . .	164
<b>8 Frequency domain methods for stability analysis</b>	<b>166</b>
8.1 Cauchy's principle of the argument . . . . .	166
8.2 Nyquist stability criterion . . . . .	169
8.3 Examples . . . . .	172
8.4 Stability margins . . . . .	176
8.5 Summary . . . . .	186
<b>9 Introduction to control design in the frequency domain</b>	<b>187</b>
9.1 Loop shaping . . . . .	187
9.2 Performance specifications . . . . .	189
9.3 Lag compensation . . . . .	191
9.4 Lead compensation . . . . .	198
9.5 Lead-lag compensation . . . . .	205
9.6 Loop shaping theory . . . . .	208
9.7 Summary . . . . .	211
<b>Bibliography</b>	<b>213</b>
<b>Index</b>	<b>216</b>

---

# Notation

$\mathbb{R}[s]$	Ring of polynomials in the complex variable $s$ with real coefficients.	10
$\coloneqq$	Equal by definition.	10
$\delta(t)$	Impulse function.	10
$\mathbb{C}^+$	Closed right half complex plane.	10
$K_{\text{gm}}$	Gain margin.	181
$K_{\text{lgm}}$	Lower gain margin.	181
$K_{\text{ugm}}$	Upper gain margin.	180
$\text{Im}(z)$	Imaginary part of $z \in \mathbb{C}$ .	90
$\mathbb{N}$	The set of natural numbers, $\{1, 2, \dots\}$ .	
$\mathbb{N}_k$	The set $\{1, 2, \dots, k\}$ .	
$\%OS$	Percentage overshoot.	105
$\Phi_{\text{lpm}}$	Lower phase margin.	177
$\Phi_{\text{pm}}$	Phase margin.	177
$\Phi_{\text{upm}}$	Upper phase margin.	177
$\text{Re}(z)$	Real part of $z \in \mathbb{C}$ .	90
$\mathbb{R}$	The set of real numbers.	
$\mathbb{R}(s)$	The set of rational functions in $s \in \mathbb{C}$ with coefficients in $\mathbb{R}$ .	48
$T_p$	Time to peak.	107
$T_r$	Rise time.	108
$T_s$	Two percent settling time.	107
$\text{adj}(\cdot)$	Adjoint matrix.	71
$\det(\cdot)$	Determinant of a square matrix.	67
$\ \cdot\ _\infty$	Least upper bound of a signal.	68
$\mathbb{C}$	The set of complex numbers.	
$\mathbb{C}^+$	Open right half complex plane.	10
$\mathbb{C}^-$	Open left half complex plane.	10
$\mathbb{Z}$	The set of integers.	
$\mathbf{1}(t)$	Unit step function	10
$\mathcal{F}(x)$	Fourier transform of a signal $x(t)$ .	45
$\mathcal{L}(f)$	Laplace transform of a signal $f(t)$ .	40
$\omega_{\text{gc}}$	Gain crossover frequency.	176
$\omega_{\text{pc}}$	Phase crossover frequency.	180
$\bar{z}$	Complex conjugate of $z \in \mathbb{C}$ .	90
$y_{\text{ss}}$	Steady-state value of a signal $y(t)$ .	73
$g * u$	Convolution of signals $g(t)$ and $u(t)$ .	64
$j$	$\sqrt{-1}$ .	90
$j\mathbb{R}$	Imaginary axis.	10
$s_m$	Stability margin.	184

---

# Acronyms and Initialisms

**B.F.L.** Buffer Fill Level. [9](#)

**B.I.B.O.** Bounded-Input Bounded-Output. [68](#)

**C.C.W.** counterclockwise. [167](#)

**C.R.H.P.** Closed Right Half Complex Plane. [10](#)

**C.W.** clockwise. [165](#)

**E.M.F.** Electromotive Force. [26](#)

**F.T.** Fourier Transform. [46](#)

**F.V.T.** Final-Value Theorem. [73](#)

**K.C.L.** Kirchhoff's Current Law. [23](#)

**K.V.L.** Kirchhoff's Voltage Law. [23](#)

**L.T.** Laplace Transform. [7](#)

**L.T.I.** Linear Time Invariant. [7](#)

**O.D.E.** Ordinary Differential Equation. [4](#)

**O.L.H.P.** Open Left Half Complex Plane. [10](#)

**O.R.H.P.** Open Right Half Complex Plane. [10](#)

**P.D.E.** Partial Differential Equation. [15](#)

**P.I.** Proportional-Integral. [9](#)

**P.I.D.** Proportional-Integral-Derivative. [149](#)

**R.E.D.** Random Early Detection. [9](#)

**R.O.C.** Region of Convergence. [40](#)

**S.I.S.O.** Single-Input Single-Output. [9](#)

**T.C.P.** Transmission Control Protocol. [8](#)

**T.F.** Transfer Function. [7](#)

# Chapter 1

---

## Introduction

This course is about the analysis and design of control systems. Control systems are ubiquitous, essential, and largely invisible to the general public. Without control systems there could be no manufacturing, no vehicles, no computers, no regulated environment — in short, no technology. Control systems are what make machines, in the broadest sense of the term, function as intended [Doyle et al., 1990].

## Contents

1.1	Familiar examples . . . . .	1
1.2	What is control engineering? . . . . .	2
1.3	Control engineering design cycle . . . . .	6
1.4	Feedback control in software and computer systems . . . . .	8
1.5	Very brief history . . . . .	9
1.6	Notation . . . . .	10
1.7	Summary . . . . .	10

### 1.1 Familiar examples

A familiar example of a control system is the one we all have that regulates our internal body temperature at 37° C. As we move from a warm room to the cold outdoors, our body temperature is maintained. Nature has many other interesting examples of control systems: blood sugar regulation in the human body; the population of a species, which varies over time, is a result of many dynamic interactions with predators and food supplies. And human organizations are controlled by regulations and regulatory agencies.

Our homes are full of control systems. There are thermostats to regulate the temperature of the house, the refrigerator, the dishwasher, the clothes washer and dryer, the oven, and the hot water heater. The level of water in the toilet tanks and the sump pump is also automatically controlled. Many household controllers are less obvious. These include the automatic controllers that regulate the flow of fuel to the furnace, those inside electronic devices in the house and those that ensure that our coffee maker and microwave oven work properly [Society, 2015].

The modern automobile is also full of automatic controllers. Pollution is reduced, fuel is saved, and comfort is enhanced by automatic controllers in the car. Stability and control augmentation systems make nearly every modern vehicle easier for the human to control and safer. Still, in 2012, over thirty thousand Americans died in car accidents [Association, 2015]. Some estimate that over 90% of fatalities are caused by human error [Smith, 2015]. Hence the control systems needed for autonomous driving have the potential to save many lives. Other examples of control systems:

- autofocus mechanism in cameras,
- noise cancelling head phones,

- cruise control system in cars,
- auto-pilot on a plane,
- flow control in data networks,
- bioengineering applications, e.g., prosthetics and artificial pancreas [Cobelli et al., 2009], [Bequette, 2005],
- economic modelling and policy development [Seierstad and Sydsæter, 1986],
- Cyber-physical systems, e.g., smart grid, traffic control systems [Kim and Kumar, 2012],
- operating systems [Hellerstein et al., 2004], [Pothukuchi et al., 2018]
- database privacy control [Ny and Pappas, 2014],
- opinion dynamics in social networks [Anderson and Ye, 2019] [Jia et al., 2015],
- middleware (application servers, database management systems, email servers) [Hellerstein et al., 2004].
- self-adaptive software systems [Filieri et al., 2015], [Patikirikorala et al., 2012].

## 1.2 What is control engineering?

A **system** maps an **input** to an **output**; for **dynamical systems** these are functions of time. If we use the symbol  $P$  to denote a system that maps an input signal  $u$  to an output signal  $y$ , then the relationship is denoted by  $y = P(u)$ . Figure 1.1 illustrates this relationship. The direction of the arrows indicate whether a signal is

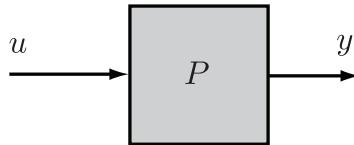


Figure 1.1: Basic block diagram of a system.

an input or an output of the system  $P$ . In control engineering we are given a system called the **plant**. We are interested in making the output of the plant behave in a desirable manner in the presence of model uncertainty and despite uncontrolled influences called **disturbances**. We change the behaviour of the plant by connecting it to another system called the **controller or compensator**<sup>1</sup>. The essential picture is given in Figure 1.2.

In Figure 1.2 the **reference signal**  $r(t)$  represents the desired output. The external disturbance is  $d(t)$ . The controller affects the behaviour of the plant through the input signal to the plant  $u(t)$  which is physically implemented by **actuators**. The basic feedback loop of (i) sensing, (ii) computation and (iii) actuation is the central concept in control.

**Example 1.2.1. (Helicopter)** In helicopter flight control, the plant is the helicopter itself plus its sensors and actuators. The controller is implemented as a program running in an on-board computer. The disturbance might be the force of the wind acting on aircraft. The design of a flight control system for a helicopter requires first the development of a mathematical model of the helicopter dynamics. This is a very advanced subject, well beyond the scope of this course. We must content ourselves with much simpler plants. ▲

---

<sup>1</sup>The term compensation is used for the process of designing a controller, reflecting the fact that one is “compensating” for deficiencies in the innate properties of the plant. The controller itself is sometimes called a compensator.

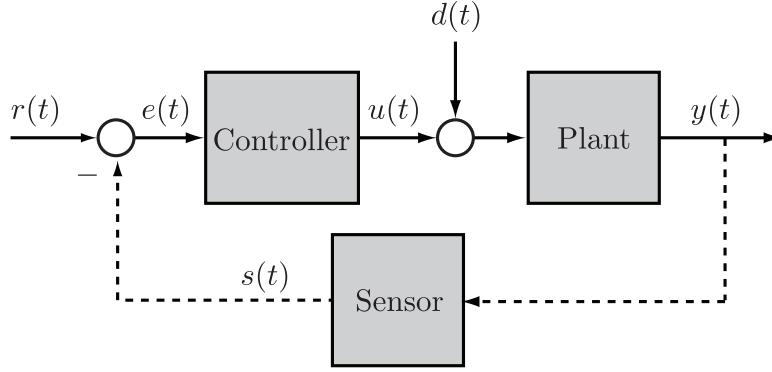


Figure 1.2: Basic block diagram of a control system.

**Example 1.2.2. (Open-Loop Control)** Suppose we want the plant output to **track** the reference signal, i.e., we want the difference  $r(t) - y(t)$  to approach zero as time passes. One way to do this is to use an **open-loop** architecture. In this case, one omits the sensors (saving money) and hence removes the part of the diagram in Figure 1.2 with dashed lines. The controller reads the reference signal  $r(t)$  and uses this information to decide on the input  $u(t)$  to the plant.

The open-loop control design has inherent problems. First, if  $d(t)$  is unknown then the disturbance may cause the plant output to deviate from the reference signal. Another problem is that the plant model used to decide on the input signal can never be a completely accurate representation of a real physical system. Since there is no feedback, the controller is oblivious to these issues and will not take any corrective action when the plant output inevitably deviates from the reference signal. ▲

A powerful way to solve these problems is to use **feedback**. In this architecture the plant output is read by sensors which may themselves be considered a system. The sensor output is then compared to the reference signal and an **error** signal  $e(t)$  is formed. The error is sent to the controller<sup>2</sup> which decides on the corrective action to be taken.

**Example 1.2.3. (Cruise Control)** A classic example is the cruise control system in a car. The plant is the car itself. The output of the plant is the car's speed. The speed is read by a speedometer whose values are sent to an embedded processor in the car. The control algorithm running on the processor compares the speedometer reading to the desired speed set by the driver and then decides whether to increase or decrease the throttle. In this example disturbances may include the pitch of the road, ice and wind. ▲

**Example 1.2.4. (Automated Shopping Cart)** Consider an automated shopping cart that follows a person around the grocery store while they shop. The cart must maintain a predefined safe distance from the shopper while not colliding with shelves. In this example the plant is the shopping cart. The actuators are motors attached to the wheels. The cart is equipped with distance sensors, e.g., a camera, that measure how far the cart is from the shelves and the shopper. The sensors send the distance information to an embedded computer. The control algorithm running on the computer compares the distance measurements to the desired distances and sends appropriate command signals to the motors. The disturbances in this example include other shoppers and carts. ▲

### 1.2.1 Detailed introductory example

A common actuator used in control systems to provide rotary motion is the **permanent magnet DC motor**. DC motors are found in robotic manipulators, ground and aerial vehicles and many other applications. Suppose

<sup>2</sup>Of course the controller may have access to the signals  $r(t)$  and  $s(t)$  separately. We'll generally stick to the simpler case where the controller reads  $e(t)$ .

we have a DC motor whose output is the angular velocity  $\omega(t)$  of its shaft, the input is a voltage  $u(t)$ , and there is a disturbance torque  $T(t)$  resulting from, for example, an unknown external load being applied to the shaft of the motor. This external torque is something we cannot alter. In Chapter 2 we'll derive the governing differential equation for this system to be

$$\frac{d\omega}{dt} = -\frac{1}{\tau}\omega(t) + k_E u(t) + k_T T(t).$$

Here  $\tau$ ,  $k_E$  and  $k_T$  are positive physical constants. This system can be represented by the block diagram in Figure 1.3a.

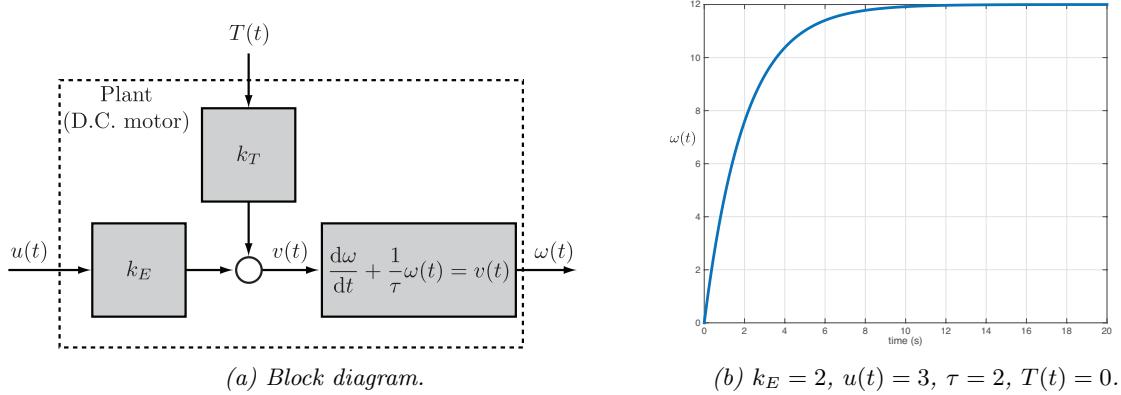


Figure 1.3: Block diagram of DC motor speed model and open-loop step response.

Suppose that we want the motor shaft to spin at a constant desired speed of  $\omega_0$  radians per second. This constant represents our reference signal, i.e.,  $\omega_{\text{ref}}(t) = \omega_0$ . Our first naïve attempt to control this system is to use an open-loop scheme. We start applying constant voltages to observe the resulting speed of the motor's shaft. From this we'll generate a rule for our control signal  $u(t)$ . If  $u(t) = u_0$  (constant) and there is no disturbance torque  $T(t) = 0$  then the system model becomes

$$\frac{d\omega}{dt} = -\frac{1}{\tau}\omega(t) + k_E u_0, \quad t \geq 0.$$

If we assume, as is reasonable, that the motor isn't moving when we apply the voltage then  $\omega(0) = 0$ . The solution to the Ordinary Differential Equation (O.D.E.) is then

$$\omega(t) = k_E u_0 \tau \left(1 - e^{-\frac{t}{\tau}}\right), \quad t \geq 0.$$

Figure 1.3b gives a plot of this signal for  $k_E = 2$ ,  $u_0 = 3$ ,  $\tau = 2$ . From either the above expression for  $\omega(t)$  or by running more experiments (with different values of  $u_0$ ) on the motor to obtain enough data like Figure 1.3b, we can figure out that if we want the motor to run at a speed of  $\omega_0$  radians per second, then we should set the voltage to

$$u(t) = \frac{\omega_0}{k_E \tau}. \quad (1.1)$$

The block diagram for this open-loop strategy is depicted in Figure 1.4. As discussed in Example 1.2.2, this strategy has problems. First, suppose that the disturbance torque is a non-zero constant  $T(t) = -T_0$ . The ODE becomes

$$\frac{d\omega}{dt} = -\frac{1}{\tau}\omega(t) + k_E u_0 - k_T T_0, \quad t \geq 0.$$

and its solution, again assuming  $\omega(0) = 0$ , is

$$\omega(t) = \tau (k_E u_0 - k_T T_0) \left(1 - e^{-\frac{t}{\tau}}\right), \quad t \geq 0.$$

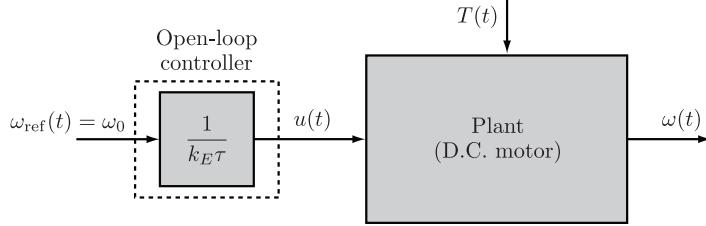


Figure 1.4: Open-loop control of a DC motor.

With the open-loop control strategy (1.1) the ultimate speed of the motor is  $\omega_0 - \tau k_T T_0$ . Figure 1.5a shows the response using the open-loop control when  $\omega_0 = 5$ ,  $k_T = 1$  and  $T_0 = 2$ . We can see that the final speed ( $\approx 1$  rad/sec) is way below the desired speed of 5. The other problem is that, due to modelling errors, we are unlikely to have exact numerical values for  $\tau$  and  $k_E$ . Figure 1.5b shows the response of the open-loop system when the actual value of  $\tau$  is 1.6 instead of our estimated value of 2. Again the motor fails to reach the desired speed of 5.

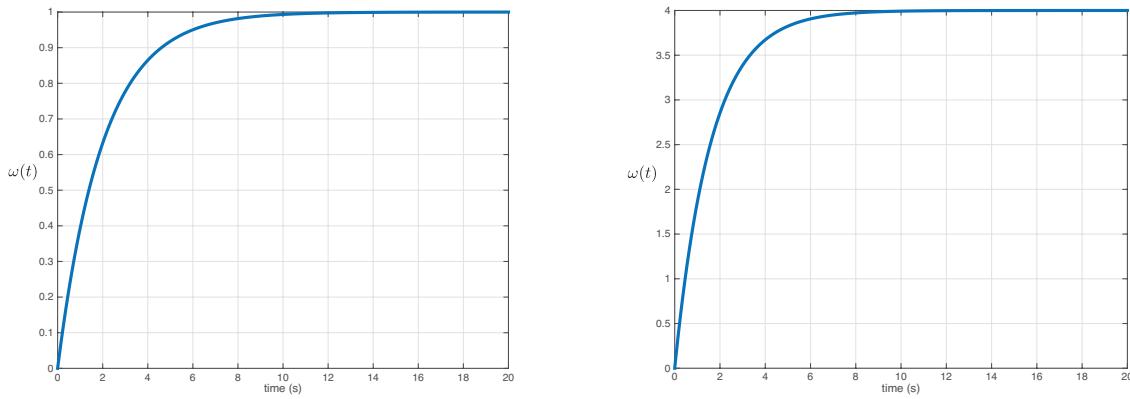


Figure 1.5: Open-loop control of a DC motor in the presence of disturbances and modelling error.

As mentioned in our introductory discussion, a powerful way to solve the problems we've just observed is to use feedback in our decision making. To that end, let's suppose instead that the motor has a tachometer which measures its shaft speed. The tachometer takes an angular velocity and returns a voltage. We'll model the tachometer as a scaling factor  $k_S$ . The voltage from the tachometer is available for feedback. Our controller converts this voltage back to the physically meaningful unit of radians per second  $\omega_{\text{meas}}(t)$  and then compares the measured speed to the reference<sup>3</sup>. This is illustrated in Figure 1.6.

In Figure 1.6 the controller computes the tracking error, i.e., the difference between the desired speed  $\omega_{\text{ref}}(t)$  and the measured speed  $\omega_{\text{meas}}(t)$ . Then it multiplies the error by a constant  $K_p$  which is the **gain** of the controller. The differential equation governing this system is

$$\frac{d\omega}{dt} = -\frac{1}{\tau}\omega(t) + k_E K_p (\omega_{\text{ref}}(t) - \bar{k}_S k_S \omega(t)) + k_T T(t).$$

The only independent signals in the closed-loop system are the reference signal and the disturbance torque. Figure 1.7a shows the response of the closed-loop system, for various values of the gain  $K_p$ , when  $\omega_{\text{ref}}(t) = 5$ ,  $\bar{k}_S k_S = 1$ , there is no disturbance torque  $T(t) = 0$  and we assume perfect knowledge of the physical constants.

Compared to the open-loop response (cf. Figure 1.3b) we see that the closed-loop system doesn't achieve the desired speed but that the response is much faster. We can try to increase the gain of the controller to see

<sup>3</sup>Ideally in Figure 1.6 we have  $\bar{k}_S k_S = 1$  so that  $\omega_{\text{meas}}(t) = \omega(t)$ . For this reason the sensor blocks in the feedback path are often omitted in which case we have a **unity feedback system**.

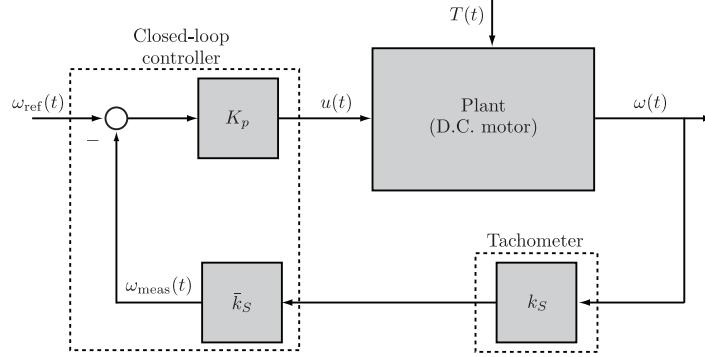


Figure 1.6: Closed-loop control of a DC motor.

if this helps. Figure 1.7a shows the response for  $K_p \in \{5, 10, 15, 20, 25\}$ . As the gain  $K_p$  increases the error decreases and the response becomes faster. High gain is a very important concept in control but we have to be careful using it. Figure 1.7b shows the applied voltage  $u(t)$  corresponding to  $K_p \in \{5, 10, 15, 20, 25\}$ . As the

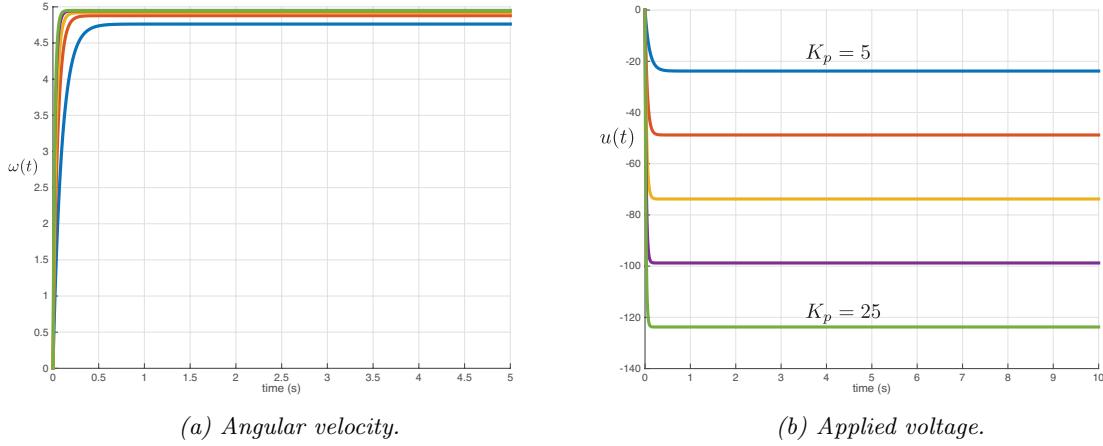


Figure 1.7: Effect of increasing gain on closed-loop response of DC motor.

gain increases so does the applied voltage. If the applied voltage is too large we will damage the motor so we need to be judicious when using high gain<sup>4</sup>.

A major benefit of the closed-loop architecture is in handling disturbances and modelling error. Figure 1.8 shows the response of the closed-loop system (cf. Figure 1.5) with  $T(t) = -2$  (Figure 1.8a) and  $\tau = 1.6$  (Figure 1.8b). In these scenarios the closed-loop system gets closer to the desired speed of  $\omega_{\text{ref}}(t) = 5$ . It's actually possible to design a controller so that we exactly track the desired speed. We'll learn how to do this as the course progresses. In summary, it is hoped that this simple introductory example helps convince you that feedback is a good thing.

## 1.3 Control engineering design cycle

The typical components of a modern computer controlled system are shown in Figure 1.9. The typical control engineering design cycle involves the following steps:

1. Study the system to be controlled and decide what types of sensors and actuators will be used and where they will be placed.

<sup>4</sup>Another problem with high gain is that it can lead to instability. We'll see this in detail as the course progresses.

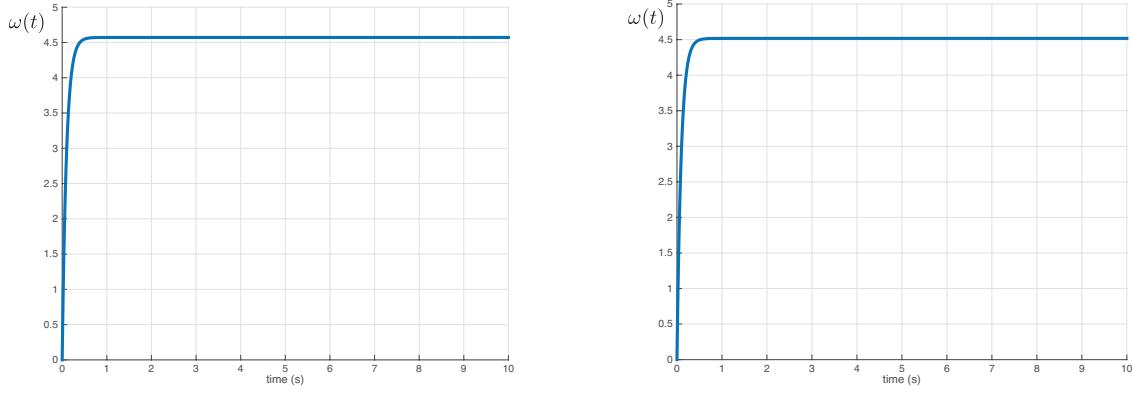


Figure 1.8: Closed-loop control of a DC motor in the presence of disturbances and modelling error.

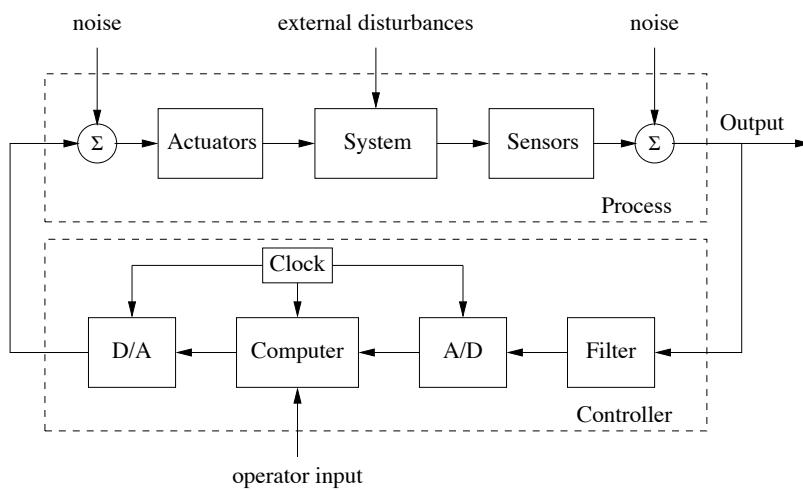


Figure 1.9: Components of a computer controlled system [Åström and Murray, 2019]. The upper dashed box represents the process dynamics, which include the sensors and actuators in addition to the dynamical system being controlled. Noise and external disturbances can perturb the dynamics of the process. The controller is shown in the lower dashed box. It consists of a filter and analog-to-digital (A/D) and digital-to-analog (D/A) converters, as well as a computer that implements the control algorithm. A system clock controls the operation of the controller, synchronizing the A/D, D/A and computing processes. The operator input is also fed to the computer as an external input.

2. Model the resulting system to be controlled.

- By model we mean a mathematical model of the plant. For a continuous-time dynamic system a model is typically one or more differential equations. Experiments may be needed to obtain numerical values of some parameters. The processes of deducing a model from experimental data is called **system identification**.

3. Simplify the model if necessary so that it is tractable.

- Classical control, the subject of this course, uses a **transfer function** to model the plant. **Linear Time Invariant (L.T.I.)** systems, and only LTI systems, have transfer functions. The **Transfer Function (T.F.)** of an LTI system is defined to be the ratio of the **Laplace Transform (L.T.)** of the output to the Laplace transform of the input where the LTs are taken with zero initial conditions<sup>5</sup>.

4. Analyze the resulting model; determine its properties.

---

<sup>5</sup>If the plant is nonlinear, which virtually every system is, then we first need to linearize its model to obtain a transfer function.

5. Decide on performance specifications.
6. Decide on the type of controller to be used.
7. Design a controller to meet the specs, if possible; if not, modify the specs or generalize the type of controller sought.
  - In this course the controller itself is a transfer function. It corresponds to an ODE that relates the controller output (input signal to plant) to the controller input (usually tracking error).
8. Simulate the resulting controlled system, either on a computer or in a pilot plant.
9. Repeat from step 1 if necessary.
10. Choose hardware and software and implement the controller.
11. Tune the controller on-line if necessary.

## 1.4 Feedback control in software and computer systems

The tools of this course find applications in networking [Hellerstein et al., 2004, Keshav, 2012, Åström and Murray, 2019]<sup>6</sup> and even algorithm design [Lessard et al., 2016] as the next few examples illustrate.

### Load control of a web server

The following example is borrowed from [Keshav, 2012, Example 8.2]. Consider a web server that responds to GET and POST queries from web browsers. The server uses a buffer to hold pending requests before they are served so that requests arriving to a busy server are not lost. Requests arrive at different time instants and are served after spending some time in the buffer.

The difference between arrivals and departures at any moment in time represents the number of queries in the buffer (queue length) at that time. Denote this quantity by  $y(t)$ . The control objectives are

- (i) Don't let queue length get too big thereby exceeding the buffer capacity causing requests to be lost.
- (ii) Don't let queue length get to zero – causing a buffer underflow and idling the web server.

Taken together, these specifications suggest that we would like to keep the number of queries in the buffer  $y(t)$  equal to some reference value  $r(t)$  that is enough to not greatly impact the server response time but large enough that if the web server were to suddenly obtain additional capacity, the buffer would not underflow.

The problem here is that the rate at which the server handles requests is unknown – it depends on various things including the number of clients. We can model the uncertain service rate as a disturbance  $d(t)$ . The controller's goal is therefore to decide on the request rate  $u(t)$  so that the number of bits in the buffer  $y(t)$  remains close to  $r(t)$  despite the disturbance  $d(t)$ . The only information available to the controller is  $r(t)$  and  $y(t)$ .

### Detecting router overloads

The following example is borrowed from [Hellerstein et al., 2004, §1.6.4]. A central element of the Internet is **Transmission Control Protocol (T.C.P.)** which provides end-to-end communication across network nodes. The designers of TCP were concerned about regulating traffic flows in the presence of network congestion. One way in which this regulation occurs is at routers that direct packets between endpoints. In particular, routers

---

<sup>6</sup>The references [Keshav, 2012, Hellerstein et al., 2004] contain various other computing examples.

have finite-size buffers. Thus, to prevent buffer overflows during congestion, routers may discard packets (which results in their later re-transmission as part of the TCP protocol).

Unfortunately, by the time that buffer overflows occur, it may be that the network is already congested. The idea behind **Random Early Detection (R.E.D.)** is to take action before congestion becomes severe. RED measures how much of critical buffers are consumed, a metric that is referred to as the **Buffer Fill Level (B.F.L.)**. As depicted in Figure 1.10, RED introduces the capability of randomly dropping packets even if buffer capacity is not exceeded. If BFL is small, no packets are dropped. However, as BFL grows larger, progressively more packets are dropped.

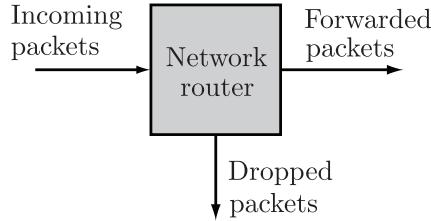


Figure 1.10: Router operation under random early detection. Incoming packets are dropped based on an adjustable drop probability.

RED has been shown to reduce network congestion and to improve network throughput. However, one challenge with using RED in practice is specifying its configuration parameters, especially the BFL at which packets start being dropped and the maximum drop probability. One approach to tuning these parameters has been to view RED as a feedback control system. Figure 1.11 depicts this perspective. Control theory is used to study the impact on stability and other properties for different settings of RED configuration parameters. **Proportional-Integral (P.I.)** control has been used for management of the accept queues of network routers.

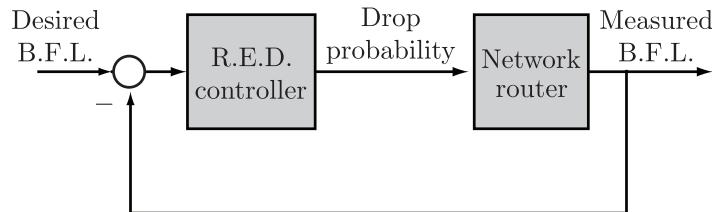


Figure 1.11: Feedback control of buffer fill level in a router. The reference input is the desired BFL, and the control input is the package drop probability.

## 1.5 Very brief history

1. Ancient control (pre-1940): ad hoc control, almost no analysis tools. Examples: water clock (2nd century BCE). More recently, in 1769 a feedback control system was invented by James Watt: the flyball governor for speed control of a steam engine. Stability of this control system was studied by James Clerk Maxwell.
2. Classical control (1940's-1950's): this constitutes the main topics covered in this course. Deals mostly with **Single-Input Single-Output (S.I.S.O.)** linear systems. The pioneering work of Bode, Nyquist, Nichols, Evans and others appeared at this time. This resulted in simple graphical tools for analyzing single-input single-output feedback control problems which are now generally known by the generic term **Classical Control Theory**. Specifications are based on closed-loop gain, bandwidth, and stability margin. Design is done using Bode plots. Controllers are typically tuned by hand. Examples: anti-aircraft guns (WWII), chemical plants, nuclear reactors, cruise control in cars, motor control.

3. Modern control (1960's - 1970's): The approach to control analysis and design, covered in ECE488, is in the time domain and uses state-space models instead of transfer functions. This period followed the work of Wiener, Kalman (and others) on optimal estimation and control. Specifications may be based on closed-loop eigenvalues, that is, closed-loop poles. This approach is known as the "state-space approach."
4. Robust and adaptive control (1980's): frequency domain tools and thinking but state-space computational methods; robust control including the effect of model error on the performance of feedback controllers; optimal control (where the controller is designed by minimizing a mathematical function). In this context classical control extends to  $\mathbf{H}_\infty$  optimization and state-space control extends to Linear-quadratic Gaussian (LQG) control. Discrete-event systems (DES). Examples: advanced aircrafts, large power systems, distributed control.
5. "Post modern" control (1990's - present): Striking developments in the control of nonlinear systems; hybrid systems; distributed robotics; sensor networks. Examples: unmanned vehicles, social networks and dynamics, cooperative control, advanced robotics including humanoids, medical applications, control over networks.

## 1.6 Notation

Generally, signals are written lower case: e.g.,  $x(t)$ . Their transforms are capitalized:  $X(s)$  or  $X(j\omega)$ . The impulse is  $\delta(t)$ . In signals and systems the unit step is denoted  $u(t)$ , but in control  $u(t)$  denotes a plant input so we'll denote the unit step by  $\mathbf{1}(t)$ . We will use the following notation for parts of the complex plane

$$\begin{aligned} \mathbb{C}^- &:= \{s \in \mathbb{C} : \operatorname{Re}(s) < 0\} && (\text{Open Left Half Complex Plane (O.L.H.P.)}) \\ \mathbb{C}^+ &:= \{s \in \mathbb{C} : \operatorname{Re}(s) > 0\} && (\text{Open Right Half Complex Plane (O.R.H.P.)}) \\ j\mathbb{R} &:= \{s \in \mathbb{C} : \operatorname{Re}(s) = 0\} && (\text{Imaginary axis}) \\ \overline{\mathbb{C}}^+ &:= \{s \in \mathbb{C} : \operatorname{Re}(s) \geq 0\} && (\text{Closed Right Half Complex Plane (C.R.H.P.)}). \end{aligned}$$

We denote by  $\mathbb{R}[s]$  the ring of polynomials in the (complex) variable  $s$  with coefficients in  $\mathbb{R}$ . The symbol  $:=$  means equal by definition. It is convenient to write vectors sometimes as column vectors and sometimes as  $n$ -tuples, i.e., ordered lists. For example

$$x = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, \quad x = (x_1, x_2).$$

We'll use both.

## 1.7 Summary

In this chapter we introduced the essential idea of feedback control. The key points that you should understand are the following.

1. Feedback control is used in all sorts of engineered systems that we use everyday and is found in many natural phenomena.
2. Control engineering is about making the output of a system (the plant) behave in a desirable manner despite disturbances acting on the system and despite modelling uncertainty. The essential picture is given in Figure 1.2 and you should understand what all the signals and systems in the figure represent.
3. In Section 1.2.1 we studied the problem of controlling the speed of a DC motor. We used this example to introduce the deficiencies of open-loop control and to motivate the use of feedback.

## 11 CHAPTER 1. INTRODUCTION

4. The “big picture” of the control engineering design cycle was presented at a high level in Section 1.3. The remainder of this course is spent filling in the details of this process. The components of a modern control system were shown in Figure 1.9.
5. Examples of feedback control applied to problems in computer hardware and software were presented. These examples demonstrate the power of the ideas covered in this course and the breadth of their application areas.
6. A brief history of the subject and notation used in this course were presented.

# Chapter 2

---

## Mathematical models of systems

Before doing any controller design, we must first find a good mathematical model of the system we want to control. A good model is simple but accurate. The system model serves as the basis on which a controller is designed. Modelling is an important part of the control engineering design process discussed in Chapter 1.

In principle, there are two different ways in which mathematical models can be obtained: from prior knowledge — e.g., in terms of physical laws — or by experimentation on a process. When attempting to obtain a specific model, it is often necessary and beneficial to combine both approaches. In this chapter we focus on the first approach to modelling, i.e., modelling based on physical principles.

## Contents

2.1	General comments on modelling . . . . .	12
2.2	Block diagrams . . . . .	15
2.3	Modelling based on first principles . . . . .	17
2.4	State-space models . . . . .	28
2.5	Linearization . . . . .	34
2.6	Simulation . . . . .	39
2.7	The Laplace transform . . . . .	40
2.8	Transfer functions . . . . .	47
2.9	Block diagram manipulations . . . . .	51
2.10	Summary . . . . .	56
2.A	Appendix: Linear functions . . . . .	56
2.B	Appendix: The impulse and convolution . . . . .	57

### 2.1 General comments on modelling

For some problems, it is both possible and feasible to find precise controller settings by simple trial and error. However, many problems preclude this approach due to complexity, efficiency, cost or even danger. Also, a trial and error approach cannot answer, before trial, questions such as:

- Given a plant and control objective, what controller can achieve the objective? Can it be achieved at all?
- Given plant and controller, how will they perform in closed-loop?
- Why is a particular loop behaving the way it is? Can it be done better? If so, by which controller?
- How would the loop performance change if the system parameters were to change, or disturbances were larger, or a sensor were to fail?

To answer these questions systematically, we need a means to capture the behaviour of the system in such a way that it can be manipulated outside the constraints of physical reality. An appropriate means to achieve this goal is to use a mathematical model [Goodwin et al., 2001]. A mathematical model is a set of equations that describe how a system transforms input signals into output signals.

There is a trade-off between model complexity and model accuracy. Typically, increasing the accuracy of a model also increases its complexity. The goal is to develop a model that is adequate for the problem at hand without being overly complex. Models are used for both control design and simulation. The models used for control design are usually simpler than simulation models. A model, no matter how detailed, is *never* a completely accurate representation of a real physical system.

One may observe that a general method of representing a system is by using a table of all possible inputs to the system and the possible responses (outputs) for these inputs. However, only very rarely will such a crude technique be useful. One major reason for paying so much attention to linear systems is that for such systems the table of input-output pairs can be drastically abbreviated [Kailath, 1980].

### 2.1.1 Examples of mathematical models

**Example 2.1.1. (Mass-Spring-Damper)** A common class of mathematical models for dynamical systems<sup>1</sup> is ordinary differential equations. In mechanics, one of the simplest differential equations is that of a mass-spring system with damping

$$M\ddot{q} + c(\dot{q}) + Kq = u. \quad (2.1)$$

Figure 2.1 illustrates this system. The variable  $q(t) \in \mathbb{R}$  represents the position of a box with mass  $M$  with

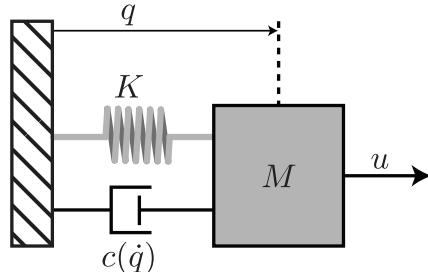


Figure 2.1: Mass-spring-damper system. The position of the mass  $M$  is  $q$  with  $q = 0$  corresponding to the rest position of the spring. The forces acting on  $M$  are due to a linear spring with constant  $k$ , a damper whose force depends on velocity  $\dot{q}$  and an applied force  $u$ .

$q = 0$  corresponding to the position of the mass when the spring is at rest. The notation  $\dot{q}$  represents the time derivative of  $q$ , i.e., the velocity of the mass. The symbol  $\ddot{q}$  represents the second derivative with respect to time, i.e., the acceleration. The spring is assumed to be linear, i.e., it satisfies Hooke's law, which says that the force exerted by the spring is proportional to the displacement of the mass. The friction element, called a damper, is taken as a possibly nonlinear function of the velocity  $\dot{q}$ . It can model effects such as viscous drag or static friction. The force  $u$  is an externally applied force which we treat as the input. This system is **second order** because the highest derivative appearing in the differential equation (2.1) is the second derivative of  $q$ .

If the damper's force is a nonlinear function of  $\dot{q}$ , then the differential equation (2.1) is **nonlinear**. If the damper function is a linear function of  $\dot{q}$ , i.e., it has the form  $c(\dot{q}) = b\dot{q}$  for some real constant  $b$ , then the differential equation is **linear**. ▲

**Example 2.1.2. (Simplified Atmospheric Model)** Consider a rectangular slice of air heated from below and cooled from above with its sides kept at constant temperatures. This is our atmosphere in its simplest

<sup>1</sup>Roughly speaking a dynamical system is one in which the effects of actions do not occur immediately. For example the speed of a car does not change instantaneously when the gas pedal is pushed.

description. The bottom is heated by the earth and the top is cooled by the void of outer space. Within this slice, warm air rises and cool air sinks. In the model, as in the atmosphere, convection cells develop, transferring heat from bottom to top. The state of the atmosphere in this model can be completely described by three variables, namely the convective flow  $x$ , the horizontal temperature distribution  $y$ , and the vertical temperature distribution  $z$ ; by three parameters, namely the ratio of viscosity to thermal conductivity  $\sigma$ , the temperature difference between the top and bottom of the slice  $\rho$ , and the width to height ratio of the slice  $\beta$ , and by three differential equations describing the appropriate laws of fluid dynamics

$$\dot{x} = \sigma(y - x), \quad \dot{y} = \rho x - y - xz, \quad \dot{z} = xy - \beta z.$$

These equations were introduced by E.N. Lorenz [Lorenz, 1963], to model the *strange* behaviour of the atmosphere and to justify why weather forecast can be erroneous. The Lorenz equations are still at the basis of modern weather forecast algorithms. In these equations there is no external influence, i.e., no input and the resulting system is said to be **autonomous**. It is also nonlinear because of the terms  $xz$  and  $xy$  in, respectively, the  $\dot{y}$  and  $\dot{z}$  equations.  $\blacktriangle$

**Example 2.1.3. (Google Page Rank)** The initial success of Google can be attributed in large part to the efficiency of its search algorithm which, linked with a good hardware architecture, creates an excellent search engine. Originally, a key part of the search engine was PageRank™, a system for ranking web pages developed by Google's founders Larry Page and Sergey Brin at Stanford University.

The main idea of the algorithm is the following. The web can be represented as an oriented (and sparse) graph in which the nodes are the web pages and the oriented paths between nodes are the hyperlinks. The basic idea of PageRank is to walk randomly on the graph assigning to each node a *vote* proportional to the frequency of return to the node. If  $x_i[k]$  denotes the vote of website  $i$  at time  $k$ , the updated vote is given by

$$x_i[k+1] = \sum_{j \in \mathcal{N}(i)} \frac{x_j[k]}{n_j}.$$

Here  $n_j$  is the number of nodes (websites) connected to node (website)  $j$  and  $\mathcal{N}(i)$  is the index set of all the nodes that link to node  $i$ . If we let  $N \approx 4 \times 10^{10}$  denote the total number of webpages on the internet, then  $i \in \{1, \dots, N\}$ . The graph, representing the internet, is not strongly connected, therefore to improve the algorithm for computing the vote, one considers a random jump, with probability  $p$  (typically 0.15) to another node (another page). As a result

$$x_i[k+1] = (1-p) \sum_{j \in \mathcal{N}(i)} \frac{x_j[k]}{n_j} + p \sum_{j=1}^N \frac{x_j[k]}{N}.$$

If we collect the variables  $x_i$  as a vector  $x \in \mathbb{R}^N$  we can express the equations for the vote in the form

$$x[k+1] = Ax[k]$$

for an  $N \times N$  matrix  $A$  of real constants. This is a linear, time-invariant **difference equation**. One can show, using the tools of ECE481, that

$$\lim_{k \rightarrow \infty} x[k] = \bar{x} \quad (\text{a constant}).$$

where all the elements  $\bar{x}_i$  of the constant vector  $\bar{x}$  are non-negative and bounded. The vector  $\bar{x}$  (after a certain normalization) is the Google Page Rank and  $\bar{x}_i$  is the rank of page  $i$ .  $\blacktriangle$

**Example 2.1.4. (Heated Rod)** Consider a thin rod of length  $L$  made of a homogeneous material, see Figure 2.2. Assume its sides are thermally insulated, while its ends are kept at constant temperatures  $T_1$  and  $T_2$ . Let  $q_0 : [0, L] \rightarrow \mathbb{R}$  be a differentiable function representing the temperature distribution in the rod at time 0,

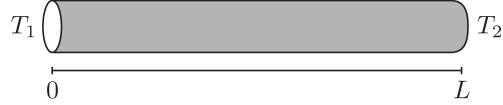


Figure 2.2: Heated Rod. The rod has length  $L$ , is insulated except for its ends which are kept at constant temperatures  $T_1$  and  $T_2$ .

i.e.,  $q_0(x)$  is the temperature of the rod at the point  $x \in [0, L]$  and at time  $t = 0$ . Then, the evolution of the temperature as time progresses is described by the **Partial Differential Equation (P.D.E.)**

$$\frac{\partial q}{\partial t} = k \frac{\partial^2 q}{\partial x^2}, \quad k > 0.$$

This is the one-dimensional **heat equation**. The unique solution to this PDE is the differentiable function  $q(t, x) : \mathbb{R} \times [0, L] \rightarrow \mathbb{R}$  such that, for all  $x \in [0, L]$  and all  $t \in \mathbb{R}$ ,

$$q(0, x) = q_0(x), \quad q(t, 0) = T_1, \quad q(t, L) = T_2.$$

If the temperature  $T_1$  at one end of the rod is allowed to be a variable input that can be freely assigned, then the heat propagation process turns into a control system. ▲

Although these examples are quite different, they share important fundamental properties. They all describe a dynamical system, i.e., a process that evolves with time. They illustrate the power of mathematical models to help us understand simple systems (like the mass-spring-damper) to very complex systems (like the atmosphere).

In obtaining the mathematical model for a given physical system, we make use of a storehouse of known physical cause-and-effect relations for individual elements—a storehouse built up through innumerable physical experiments, performed (sometimes at great expense) by many research teams. It is unrealistic for an engineer to be an expert on all classes of system models, this requires domain specific knowledge. Instead, the aim of this chapter is to give an introduction to basic modelling and linearization.

## 2.2 Block diagrams

The importance of block diagrams in control engineering can't be overemphasized. One could easily argue that you don't understand your system until you have a block diagram of it. We shall take the point of view that a block diagram is a picture of a function. We can draw a picture of the function  $y = f(x)$  like in Figure 2.3. Thus a box represents a function and the arrows represent variables; the input is the independent variable, the

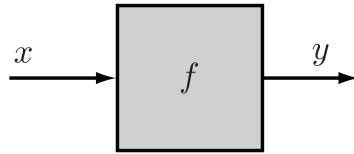


Figure 2.3: A block diagram is a picture of a function.

output the dependent variable.

**Example 2.2.1. (Simple Cart)** The simplest vehicle to control is a cart on wheels illustrated in Figure 2.4. Figure 2.4a is a **schematic diagram**, not a block diagram, because it doesn't say which of  $u$ ,  $y$  causes the other.

Assume the cart has mass  $M$  and can only move in a straight line on a flat surface. Assume a force  $u$  is applied to the cart and let  $y$  denote the position of the cart measured from a stationary reference position. Then  $u$  and  $y$  are functions of time  $t$  and we could indicate this by  $u(t)$  and  $y(t)$ . We regard the functions  $u$  and  $y$  as **signals**.

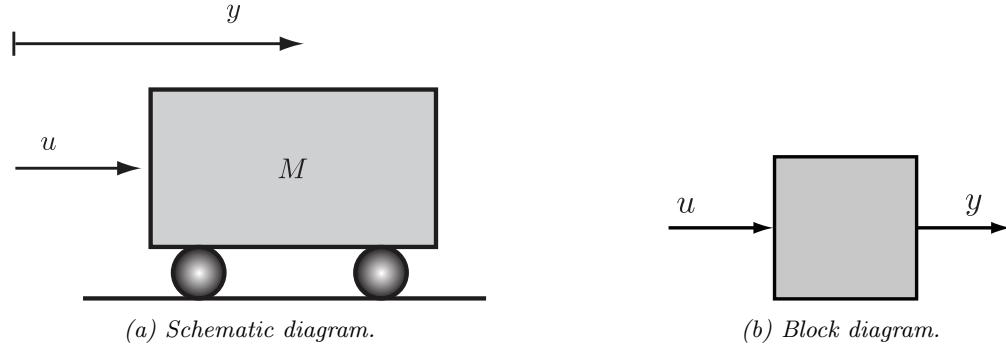


Figure 2.4: Schematic and block diagrams of a simple cart.

Newton's second law tells us that there's a mathematical relationship between  $u$  and  $y$ , namely,  $M\ddot{y} = u$ . We take the viewpoint that the force can be applied independently of anything else, that is, it's an input. Then  $y$  is an output. We represent this graphically by the **block diagram** in Figure 2.4b.

Suppose the cart starts at rest at the origin at time 0, i.e.,  $y(0) = \dot{y}(0) = 0$ . Then the position depends only on the force applied. However  $y$  at time  $t$  depends on  $u$  not just at time  $t$ , but on past times as well. So we can write  $y = f(u)$ , i.e.,  $y$  is a function of  $u$ , but we can't write  $y(t) = f(u(t))$  because the position at time  $t$  doesn't depend only on the force at that same time  $t$ . ▲

We may need to allow a block to have more than one input as in Figure 2.5. This means that  $y$  is a function of  $u$  and  $v$ ,  $y = f(u, v)$ . Block diagrams may also have **summing junctions** as in Figure 2.6.

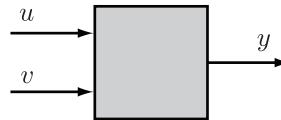
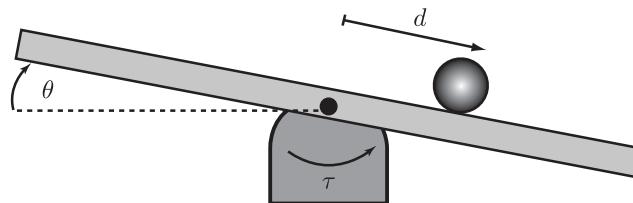


Figure 2.5: Block with multiple inputs.



Figure 2.6: Summing junctions.

**Example 2.2.2. (Ball and Beam)** This example concerns a beam balanced on a fulcrum as shown in Figure 2.7. This system is used in the lab for ECE481. Suppose a torque  $\tau$  is applied to the beam. Let  $\theta$  denote the angle of tilt and  $d$  the distance of roll. Then both  $\theta$  and  $d$  are functions of  $\tau$ . The block diagram could be

Figure 2.7: Ball and Beam. A ball sits on a beam that can be rotated by a torque applied at the centre of the beam. The position of the ball is denoted by  $d$ , the angle of the beam is  $\theta$  and the applied torque is  $\tau$ .

as in Figure 2.8a or, as in Figure 2.8b, or initial conditions, if they are not fixed, could be modelled as inputs too.



Figure 2.8: Block diagrams for ball and beam system.

## 2.3 Modelling based on first principles

The systems we model in this course are simple and are generally governed by linear differential equations. In this section we review examples of devices whose behaviour is reasonably well-modelled by such equations.

### 2.3.1 Mechanical devices

Consider a mass  $M$  restricted to move in a single direction. Let  $q \in \mathbb{R}$  denote its position. Then Newton's equations give

$$\begin{aligned} M\ddot{q} &= \sum \text{applied forces} \\ &= \sum F_i. \end{aligned} \tag{2.2}$$

The same sort of thing happens with rotary devices like the one in Figure 2.9. Let  $J$  denote the moment of

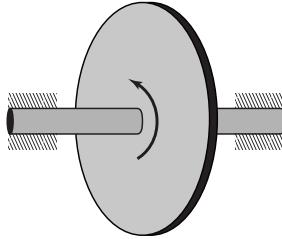


Figure 2.9: Rotor on a shaft. A rotor or disk is attached to a shaft. The rotation of the shaft causes the rotor to spin. The position of the disk is its angle  $\theta$  with respect to some fixed reference. Its velocity is  $\dot{\theta}$  and its acceleration is  $\ddot{\theta}$ .

inertia of the rotor about its axis of rotation. Let  $\theta$  denote the rotor's angular position. Then the governing equation for this system is

$$\begin{aligned} J\ddot{\theta} &= \sum \text{applied torques} \\ &= \sum \tau_i. \end{aligned} \tag{2.3}$$

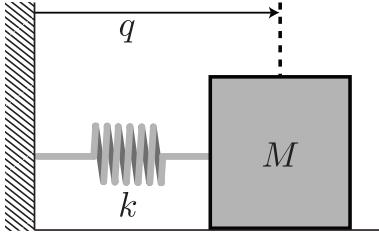
The forces and torques<sup>2</sup> that appear on the right-hand side of the above summations are usually modelled by various types of simple components. These simple components form the building blocks of mechanical models. Next we review the equations for them.

---

<sup>2</sup>A torque is sometimes called a *moment of force* or *moment* for short.

### Linear springs

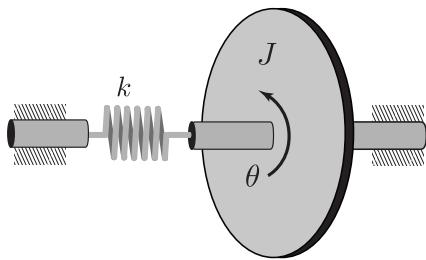
For linear translational springs the force due to the spring is given by



$$F_k(q) = kq, \quad k > 0, \quad (2.4)$$

Figure 2.10: Mass-Spring:  $M\ddot{q} = -F_k(q) = -kq$ .

where  $k$  is the spring constant. The torque due to a rotational or torsion spring obeys a rotational version of Hooke's law. It is often used to model flexible shafts and its equation is

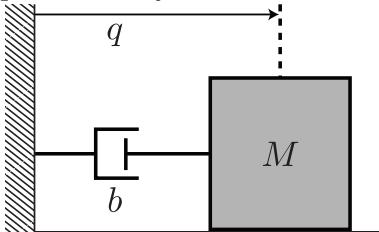


$$\tau_k(\theta) = k\theta, \quad k > 0. \quad (2.5)$$

Figure 2.11: Torsion-Spring:  $J\ddot{\theta} = -\tau_k(\theta) = -k\theta$ .

### Linear dampers

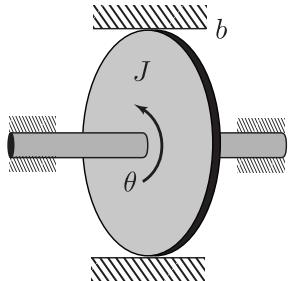
A linear damper is usually used to model viscous friction. For translational motion the equation is



$$F_b(\dot{q}) = b\dot{q}, \quad b > 0, \quad (2.6)$$

Figure 2.12: Mass-Damper:  $M\ddot{q} = -F_b(q) = -b\dot{q}$ .

where  $b$  is the coefficient of damping. The torque due to a rotational or inertia damper is given by



$$\tau_b(\theta) = b\dot{\theta}, \quad b > 0. \quad (2.7)$$

Figure 2.13: Inertia-Damper:  $J\ddot{\theta} = -\tau_b(\theta) = -b\dot{\theta}$ .

We follow the procedure from [Cannon Jr., 2003] for deriving the equations of motions for mechanical systems.

1. Define a reference frame from which to measure distance.
2. Choose a set of coordinates that determine the configuration of the system.
3. Separate the system into its mechanical components. Each component should be either a single point mass or a single rigid body.
4. For each component determine all external forces and moments (torques) acting on it.
5. For each component, express the position of the centre of mass in terms of the chosen coordinates.
6. Apply Newton's equations for translational motion to each mechanical component.
7. Apply Newton's equations for rotational motion to each mechanical component. For each component, the sum of moments about a point that is either (a) the centre of mass or (b) in the component but stationary, equals the moment of inertia of the component about that point multiplied by the angular acceleration.

We follow this procedure in the next example though it is too simple to be really representative.

**Example 2.3.1. (Car Suspension)** Find the governing differential equations for the simplified quarter model of a car suspension system shown in Figure 2.14. Here  $M_1$  represents 1/4 of the mass of the car chassis while  $M_2$  represents half the axle mass.

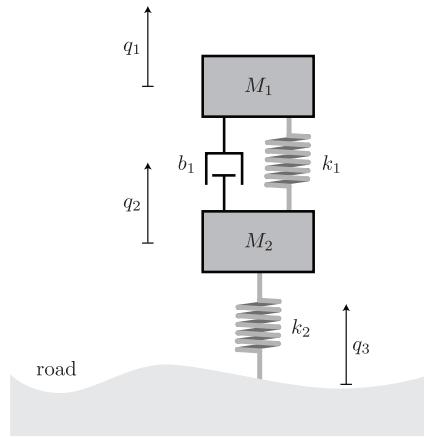


Figure 2.14: Simplified model of a car suspension system.

1. In this example the reference frame is already suggested by Figure 2.14. From the figure we see that up is taken as the positive direction.
2. The figure is already suggesting the coordinates  $q_1, q_2, q_3$  so we'll use them.
  - We assume that the origin of our coordinates, i.e., the zero position  $(q_1, q_2, q_3) = (0, 0, 0)$ , is the position at which all the springs are balanced by the force gravity. In this way we can ignore gravity. We can do this when we have linear springs.
  - The variable  $q_1$  represents how far  $M_1$  has moved from its rest position. The variable  $q_2$  represents how far  $M_2$  has moved from its rest position. The variable  $q_3$  represents the distance from the road to  $M_2$ . It is an **exogenous signal**, i.e., it comes from the “outside world.” We can view it as a disturbance to the suspension system.
3. In this example there is no rotational motion. We have two masses  $M_1$  and  $M_2$  which we treat as point masses.



Figure 2.15: Free body diagrams for Example 2.3.1.

4. Free body diagrams are very useful to accomplish this step. Figure 2.15 shows the free body diagrams for both masses.
5. Since this example is so simple, we already did this when we chose the set of coordinates in step 2.
6. Apply Newton's equations for translational motion to each mechanical component.

- For mass 1 we have

$$\begin{aligned} M_1 \ddot{q}_1 &= \sum \text{applied forces} \\ M_1 \ddot{q}_1 &= -k_1(q_1 - q_2) - b_1(\dot{q}_1 - \dot{q}_2) \\ \Rightarrow M_1 \ddot{q}_1 + b_1 \dot{q}_1 + k_1 q_1 &= b_1 \dot{q}_2 + k_1 q_2. \end{aligned}$$

- For mass 2 we have

$$\begin{aligned} M_2 \ddot{q}_2 &= \sum \text{applied forces} \\ M_2 \ddot{q}_2 &= k_1(q_1 - q_2) + b_1(\dot{q}_1 - \dot{q}_2) - k_2(q_2 - q_3) \\ \Rightarrow M_2 \ddot{q}_2 + b_1 \dot{q}_2 + (k_1 + k_2) q_2 &= k_1 q_1 + b_1 \dot{q}_1 + k_2 q_3. \end{aligned}$$

7. There is no rotational motion in this example so we can skip this step.

In summary, the differential equations governing this mechanical system are

$$\begin{aligned} M_1 \ddot{q}_1 + b_1 \dot{q}_1 + k_1 q_1 &= b_1 \dot{q}_2 + k_1 q_2 \\ M_2 \ddot{q}_2 + b_1 \dot{q}_2 + (k_1 + k_2) q_2 &= k_1 q_1 + b_1 \dot{q}_1 + k_2 q_3. \end{aligned}$$



The next example follows the same procedure for a system with both translational and rotational motion.

**Example 2.3.2. (Anti-lock Brake System)** An Anti-lock Braking System (ABS) can greatly improve the safety of a vehicle in extreme circumstances, as it maximizes the longitudinal tire-road friction while maintaining large lateral forces, which guarantee vehicle steerability. We'd like to obtain a model for a rolling wheel on a flat road as shown in Figure 2.16. In this figure  $r$  is the wheel's radius, the term  $\tau_b$  represents the braking torque, i.e., the input signal, and  $F(\dot{q})$  is the force due to friction with the road. We follow the same general procedure as in the previous example.

1. Define a reference frame from which to measure distance.
  - We take the left direction as positive translational motion and the counterclockwise rotation of the wheel as the positive direction of rotation. This choice is already suggested by Figure 2.16.
2. Choose a set of coordinates that determine the configuration of the system.

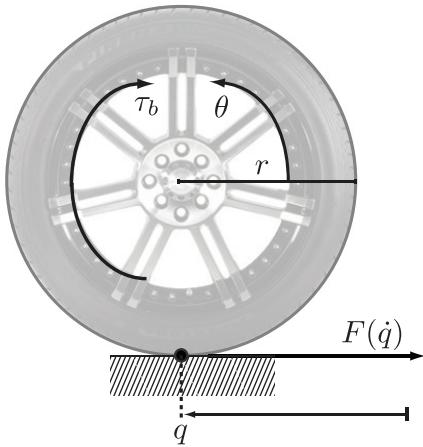


Figure 2.16: A rolling wheel.

- Let  $q$  denote the position of the contact point of the wheel and let  $\theta$  denote the angle that a point on the wheel makes with the road.
3. Separate the system into its mechanical components.
- In this example there is one rigid body, the wheel.
  - Let  $M$  denote the mass of the wheel.
  - Let  $J$  denote the moment of inertia of the wheel about its axis of rotation.
4. For each component determine all external forces and moments (torques) acting on it.
- Figure 2.16 already shows the relevant forces.
  - For translational motion, the only force is  $F(\dot{q})$  pointing in the negative  $q$  direction.
  - For rotational motion, the torques are  $rF(\dot{q})$  in the positive  $\theta$  direction and the braking torque  $\tau_b$  in the negative  $\theta$  direction.
5. For each component, express the position of the centre of mass in terms of the chosen coordinates.
- Since this example is so simple, we already did this when we chose the set of coordinates. The centre of mass has position  $q$ . We ignore the vertical coordinate since we are assuming a flat road surface.
6. Apply Newton's equations for translational motion to each mechanical component.
- We have
- $$M\ddot{q} = -F(\dot{q})$$
7. Apply Newton's equations for rotational motion to each mechanical component.
- In this case we have
- $$J\ddot{\theta} = rF(\dot{q}) - \tau_b.$$

In summary, the differential equations governing this mechanical system are

$$\begin{aligned} M\ddot{q} &= -F(\dot{q}) \\ J\ddot{\theta} &= rF(\dot{q}) - \tau_b. \end{aligned}$$

If we model the road friction as a linear damper, then  $F(\dot{q}) = b\dot{q}$ . When simulating an ABS system, the friction model is more sophisticated than just a linear damper. However, for the purpose of controller design, the linear model of friction is a reasonable starting point. ▲

**Example 2.3.3. (Physical Pendulum)** Consider the physical pendulum shown in Figure 2.17a. It consists of a rod made of a homogenous material of length  $\ell$  and mass  $M$  hanging from a pivot point. Its center of mass is at the midpoint  $\ell/2$ . A motor connected to the rod at the pivot point provides a torque  $\tau$ . Assume there is no friction.



Figure 2.17: A physical (rod) pendulum.

- It is standard practice to let counterclockwise rotation correspond to positive angular displacement. There is no translational motion so we don't need to define any other reference frames.
- Following Figure 2.17b we choose the downward position of the pendulum to correspond to the angle  $\theta = 0$ . By our choice above,  $\theta$  increases as the pendulum rotates counterclockwise.
- In this example there is only rotational motion. The moment of inertia about the rotational axis is<sup>3</sup>  $J = M\frac{\ell^2}{12} + M\left(\frac{\ell}{2}\right)^2 = \frac{1}{3}M\ell^2$ . You are not expected to know how to compute  $J$ .
- For each component determine all external forces and moments (torques) acting on it.
  - Figure 2.17b shows a free body diagram of the moments acting on the pendulum.
- The position of the centre of mass is completely determined by  $\theta$ .
- In light of the previous point, we can ignore translational motion in this example.
- Apply Newton's equations for rotational motion to each mechanical component about the pivot point.
  - Using basic trigonometry on the force due to gravity

$$J\ddot{\theta} = \tau - Mg\frac{\ell}{2} \sin(\theta).$$

In summary, the equation of motion is

$$\begin{aligned} \frac{1}{3}M\ell^2\ddot{\theta} &= \tau - Mg\frac{\ell}{2} \sin(\theta) \\ \Rightarrow \ddot{\theta} &= \frac{3}{M\ell^2}\tau - 1.5\frac{g}{\ell} \sin(\theta). \end{aligned}$$

In this example the governing equations are nonlinear. ▲

---

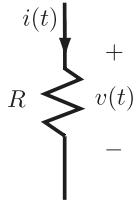
<sup>3</sup>Apply the parallel axis theorem.

### 2.3.2 Electrical devices

The fundamental principals of circuit analysis are **Kirchhoff's voltage law** and **Kirchhoff's current law**. Kirchhoff's Voltage Law (K.V.L.) states that the sum of voltage drops around a closed loop must be zero and Kirchhoff's Current Law (K.C.L.) states that the sum of the currents entering a node must be zero. The voltages and currents that must be summed in these laws depend on the electrical components in the circuit.

#### Resistors

The current-voltage characteristics of a resistor  $R$  are static. Namely, the voltage  $v(t)$  across a resistor at any fixed time  $t$  only depends on the current  $i(t)$  at that particular time. The equations are



$$v(t) = Ri(t), \quad R > 0. \quad (2.8)$$

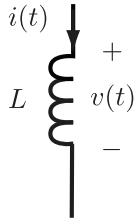
Figure 2.18: A linear resistor.

Equivalently, using complex impedances in the Laplace domain (see Section 2.7),

$$V(s) = RI(s). \quad (2.9)$$

#### Inductors

The current-voltage characteristics of an inductor  $L$  are dynamic. Namely, the current  $i(t)$  passing through an inductor at any fixed time  $t$  depends on the voltage across the inductor  $v(\tau)$  for  $\tau \leq t$ . The equations are



$$\begin{aligned} i(t) &= \frac{1}{L} \int_0^t v(\tau) d\tau \\ v(t) &= L \frac{di(t)}{dt}. \end{aligned} \quad (2.10)$$

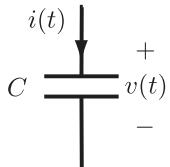
Figure 2.19: An inductor.

In the Laplace domain, if we assume zero initial conditions, equation (2.10) becomes

$$V(s) = sLI(s). \quad (2.11)$$

#### Capacitors

The current-voltage characteristics of a capacitor  $C$  are also dynamic. Namely, the voltage  $v(t)$  across a capacitor at any fixed time  $t$  depends on the current passing through the capacitor  $i(\tau)$  for  $\tau \leq t$ . The equations are



$$\begin{aligned} v(t) &= \frac{1}{C} \int_0^t i(\tau) d\tau \\ i(t) &= C \frac{dv(t)}{dt}. \end{aligned} \quad (2.12)$$

Figure 2.20: A capacitor

In the Laplace domain, if we assume zero initial conditions, equation (2.12) becomes

$$V(s) = \frac{1}{sC} I(s). \quad (2.13)$$

### Operational amplifiers

Operational amplifiers, often called **op-amps**, are among the most widely used electronic devices. We only consider ideal op-amps in this course. The circuit symbol for an ideal op-amp is shown in Figure 2.21. KCL and KVL are used to analyse op-amp circuits under two fundamental assumptions which are valid for ideal op-amps and approximately valid for more sophisticated models.

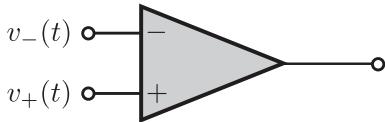


Figure 2.21: Ideal op-amp.

1. There is no current into, or out of, either the positive or negative terminals.
2. The voltage at the positive terminal  $v_+(t)$  equals the voltage at the negative terminal  $v_-(t)$ .

**Example 2.3.4. (Low Pass Filter)** Consider the RC low pass filter in Figure 2.22 with input voltage  $u(t)$  and output voltage  $y(t)$  — the voltage across the capacitor. This circuit only has one loop so that Kirchhoff's

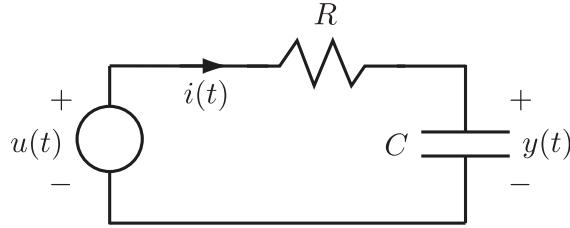


Figure 2.22: An RC filter.

voltage law gives

$$-u(t) + v_R(t) + y(t) = 0$$

where  $v_R(t)$  is the voltage across the resistor. Substituting in the device equations for the resistor and capacitor we get

$$-u(t) + Ri(t) + \frac{1}{C} \int_0^t i(\tau) d\tau = 0.$$

If we note that  $i(t) = C\dot{y}(t)$ , then the above can be written as an ODE

$$RC\dot{y}(t) + y(t) = u(t).$$



**Example 2.3.5.** Consider the RLC circuit in Figure 2.23. Apply KVL to this circuit to get

$$-u(t) + v_R(t) + v_C(t) + v_L(t) = 0$$

where  $v_R(t)$ ,  $v_C(t)$ ,  $v_L(t)$  are the voltages across, respectively, the resistor, capacitor and inductor. We substitute the device equations and use the definition of  $y(t)$  to obtain

$$-u(t) + Ry(t) + Ly(t) + \frac{1}{C} \int_0^t y(\tau) d\tau = 0.$$

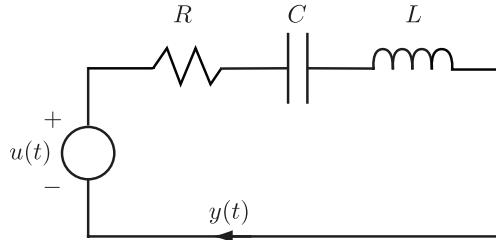


Figure 2.23: RLC circuit.

This is not a differential equation because of the integral. Take the time derivative of this equation to obtain an ODE model

$$-\dot{u} + R\dot{y} + L\ddot{y} + \frac{1}{C}y = 0.$$

▲

**Example 2.3.6. (Pure Gain Op-Amp)** An ideal op-amp circuit is shown in Figure 2.24. Determine the governing equations of this circuit. With the ideal op-amp assumptions in mind, i.e.,

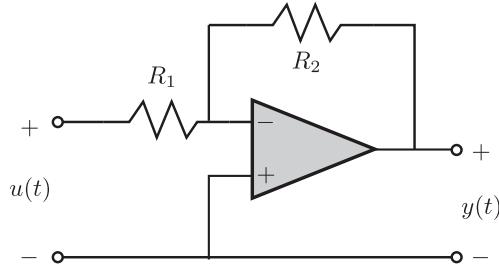


Figure 2.24: A pure gain op-amp circuit

- There is no current into, or out of, either the positive or negative terminals,
- The voltage at the positive terminal \$v\_+(t)\$ equals the voltage at the negative terminal \$v\_-(t)\$,

we perform KCL at the negative terminal

$$\frac{u(t) - v_-(t)}{R_1} + \frac{y(t) - v_-(t)}{R_2} = 0.$$

From the ideal op-amp assumptions we have \$v\_+(t) = v\_-(t)\$. Without loss of generality we can measure all voltages relative to the positive terminal so that \$v\_+(t) = 0\$. Therefore

$$R_2u(t) + R_1y(t) = 0.$$

This is a static system since the value of \$y\$ at time \$t\$ only depends on the value of \$u\$ at time \$t\$, i.e., \$y(t) = -R\_2/R\_1u(t)\$. For this reason the circuit is called a pure gain system.

▲

### 2.3.3 Electromechanical devices

**Example 2.3.7. (DC Motor)** A common actuator used to provide rotary motion in control systems is the **permanent magnet DC motor** which was discussed in Section 1.2.1. In this example we perform a detailed derivation of its equations of motion.

A DC motor works on the principle that a current-carrying conductor in a magnetic field experiences a force. The motor consists of a fixed **stator** and a movable **rotor** (also called the **armature**) that rotates inside the stator.

The stator produces a magnetic field. DC motors can be classified according to the way that the magnetic field is produced. Here we are discussing a permanent magnet motor whose stator consists of a permanent magnet. The key physical cause-and-effect relations are:

1. If the stator produces a magnetic field and a current  $i$  passes through the armature, then there will be a torque on the armature causing it to rotate.
2. Whenever a conductor moves inside a magnetic field, a voltage is created across its terminals that is proportional to the velocity of the conductor in the field. This voltage is called the **back emf**.

The “motor equations” model these two phenomena and provide the link between the electrical and mechanical subsystems of the DC motor. They give the torque  $\tau(t)$  on the rotor in terms of the armature current  $i(t)$  and express the back emf voltage<sup>4</sup>  $e(t)$  in terms of the shaft’s rotational velocity  $\dot{\theta}(t)$

$$\begin{aligned}\tau(t) &= K_\tau i(t) \\ e(t) &= K_e \dot{\theta}(t).\end{aligned}\tag{2.14}$$

Here  $K_\tau$  and  $K_e$  are physical constants.

To model a DC motor we first draw a functional block diagram in Figure 2.25. In this figure  $v(t)$  is the voltage applied to the rotor circuit,  $i(t)$  is the resulting current in the circuit. This current acts as an input to the mechanical subsystem through the first motor equation (2.14). The signal  $\theta(t)$  is the angular position of

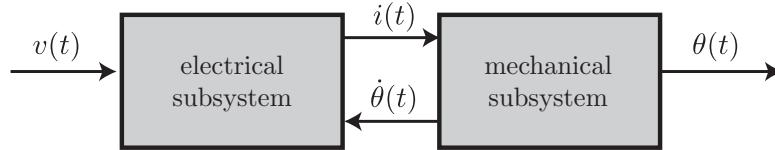


Figure 2.25: Functional block diagram of a DC motor.

the motor’s shaft while  $\dot{\theta}(t)$  is its angular velocity which acts as an input to the electrical subsystem via the second motor equation (2.14).

Next we model each block in the functional diagram. Start by drawing the equivalent circuit for the electrical subsystem and the free body diagram for the mechanical subsystem, the rotor, as shown in Figure 2.26. We denote the rotor shaft’s inertia with the symbol  $J$  and assume there is viscous friction, modelled using a rotational damper with coefficient  $b$ .

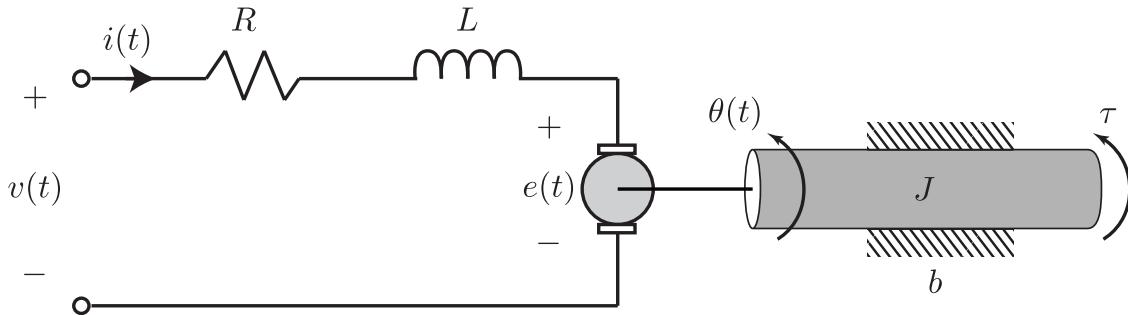


Figure 2.26: DC Motor: Circuit and free body diagram for the rotor / shaft.

<sup>4</sup>Since the generated Electromotive Force (E.M.F.) works against the applied armature voltage, it is called **back emf**.

Application of Newton's laws to the shaft, along with the motor equations (2.14), yield

$$\begin{aligned} J\ddot{\theta} &= -b\dot{\theta} + \tau \\ &= -b\dot{\theta} + K_\tau i. \end{aligned} \quad (2.15)$$

Notice that this equation assumes that the only source of inertia in the motor is from to the shaft itself. If we connect a load to the motor then this is no longer true and the load can be modelled as a disturbance torque on the right-hand side of (2.15). The electrical equation is found using KVL

$$\begin{aligned} -v + Ri + L \frac{di}{dt} + e &= 0 \\ \Rightarrow -v + Ri + L \frac{di}{dt} + K_e \dot{\theta} &= 0. \end{aligned} \quad (2.16)$$

In summary, (2.15) and (2.16) are the equations of motion governing a DC motor. ▲

**Remark 2.3.1.** If in Example 2.3.7 the response of the circuit is much faster than that of the mechanical system, e.g., if the inductance  $L \approx 0$ , then  $v(t) = Ri(t) + K_e \dot{\theta}(t)$  and the equations (2.15), (2.16) reduce to

$$J\ddot{\theta} = -\left(b + \frac{K_\tau K_e}{R}\right)\dot{\theta} + \frac{K_\tau}{R}v. \quad (2.17)$$

If we now define the constants

$$\omega(t) := \dot{\theta}(t), \quad \frac{1}{\tau} := b + \frac{K_\tau K_e}{R}, \quad k_E := \frac{K_\tau}{R},$$

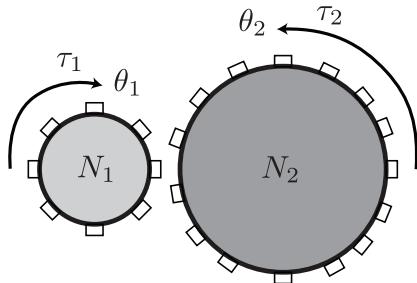
then (2.17) recovers the model used for motor speed control in Section 1.2.1. ◆

Our last example, adapted from [Gopal, 1963], provides a somewhat complete treatment of how to model the connection of mechanical components to DC motors.

**Example 2.3.8. (Geared Drive)** Electric motors generally produce their maximum power (= torque  $\times$  angular velocity) at high speeds. Therefore at high speeds, they generally produce small torques. As a consequence, gears are needed to drive large loads (that require high torque) at high speeds.

Consider the setup in Figure 2.27. A DC motor as in Example 2.3.7 is connected through a **gear train** to the physical pendulum from Example 2.3.3. The gear with  $N_1$  teeth is called the primary gear and the gear with  $N_2$  teeth is called the secondary gear. We denote the angular displacement of the motor shaft by  $\theta_1$  and the angular displacement of the pendulum by  $\theta_2$ . The coordinates are such that  $\theta_1 = 0$  when  $\theta_2 = 0$ . Let  $J_1$  denote the inertia of the motor shaft and the gears. Let  $\tau_2$  denote the torque on the pendulum.

To analyse this system we first need to understand the effect of the gears. Ideal gears have no dynamics but they impose algebraic constraints<sup>5</sup>



$$\begin{aligned} N_1 \theta_1 &= N_2 \theta_2 \\ \frac{\tau_1}{N_1} &= \frac{\tau_2}{N_2}. \end{aligned} \quad (2.18)$$

Figure 2.28: Gears

<sup>5</sup>In reality, there is always a certain amount of **backlash** (free play) between coupled gears. On the one hand, excessive backlash results in a nonlinear relationship between  $\theta_1$  and  $\theta_2$  and can lead to undesirable sustained oscillations and positional inaccuracy. On the other hand, keeping the backlash small inevitably increases the friction between the gears causing the teeth to wear out faster. We assume ideal gear equations (2.18).

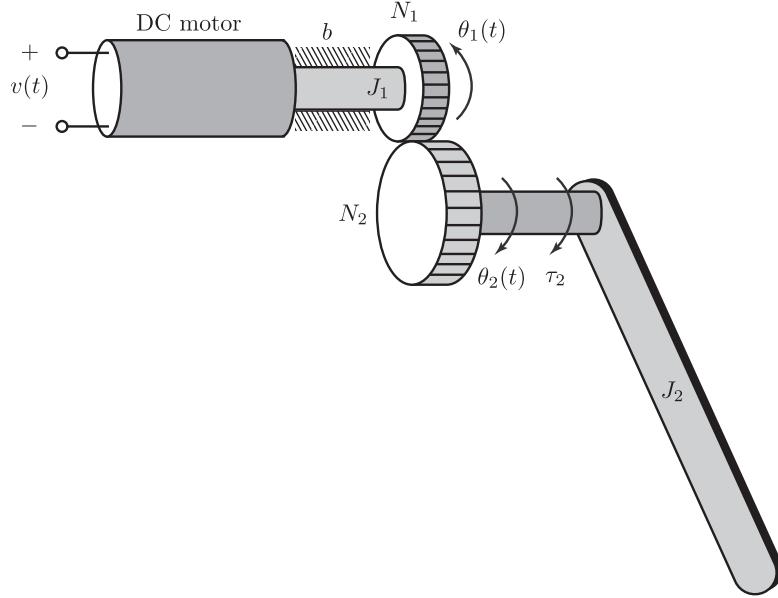


Figure 2.27: DC Motor connected to a physical pendulum.

From Example 2.3.7 the motor's dynamics are

$$\begin{aligned} J_1 \ddot{\theta}_1 &= -b\dot{\theta}_1 + K_\tau i - \frac{N_1}{N_2} \tau_2 \\ L \frac{di}{dt} &= -Ri - K_e \dot{\theta}_1 + v. \end{aligned}$$

The torque from the pendulum appears as a disturbance to the motor's angular acceleration (cf. Section 1.2.1). Notice that if  $N_1/N_2 \ll 1$  then the disturbance's effect is reduced. From Example 2.3.3 the pendulum's dynamics are

$$J_2 \ddot{\theta}_2 + Mg \frac{\ell}{2} \sin(\theta_2) = \tau_2.$$

If we replace the expression for  $\tau_2$  into the above equation and use the gear equations (2.18) we have

$$\begin{aligned} J_1 \ddot{\theta}_1 &= -b\dot{\theta}_1 + K_\tau i - \frac{N_1}{N_2} \left( \frac{N_1}{N_2} J_2 \ddot{\theta}_1 + Mg \frac{\ell}{2} \sin \left( \frac{N_1}{N_2} \theta_1 \right) \right) \\ L \frac{di}{dt} &= -Ri - K_e \dot{\theta}_1 + v. \end{aligned}$$

Simplifying the above we get the model

$$\begin{aligned} \left( J_1 + \left( \frac{N_1}{N_2} \right)^2 J_2 \right) \ddot{\theta}_1 &= -b\dot{\theta}_1 + K_\tau i - \frac{N_1}{N_2} Mg \frac{\ell}{2} \sin \left( \frac{N_1}{N_2} \theta_1 \right) \\ L \frac{di}{dt} &= -Ri - K_e \dot{\theta}_1 + v. \end{aligned}$$



## 2.4 State-space models

State-space models are a way of expressing mathematical models in a standard form.

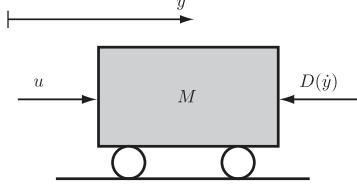


Figure 2.29: A simple cart subject to air resistance.

**Example 2.4.1. (Cart with Air Resistance)** Consider a cart on wheels, driven by a force  $u$  and subject to air resistance as in Figure 2.29. Typically air resistance creates a force depending on the velocity  $\dot{y}$ ; let's say this force is a possibly nonlinear function  $D(\dot{y})$ . Assuming  $M$  is constant, Newton's second law gives

$$M\ddot{y} = u - D(\dot{y}).$$

We are going to put this in a standard form by defining two so-called state variables, in this example position and velocity:

$$x_1 := y, \quad x_2 := \dot{y}.$$

Then

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{1}{M}u - \frac{1}{M}D(x_2) \\ y &= x_1.\end{aligned}$$

These equations have the form

$$\begin{aligned}\dot{x} &= f(x, u) && \text{(state equation)} \\ y &= h(x) && \text{(output equation)}\end{aligned}\tag{2.19}$$

where

$$\begin{aligned}x &:= \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}, & u &:= \text{applied force} \\ f : \mathbb{R}^2 \times \mathbb{R} &\rightarrow \mathbb{R}^2, & f(x_1, x_2, u) &= \begin{bmatrix} x_2 \\ \frac{1}{M}u - \frac{1}{M}D(x_2) \end{bmatrix} \\ h : \mathbb{R}^2 &\rightarrow \mathbb{R}, & h(x_1, x_2) &= x_1.\end{aligned}$$

The function  $f$  is nonlinear if  $D$  is;  $h$  is linear. Equation (2.19) constitutes a state-space model of the system, and  $x$  is called the **state** or **state vector**. Here the plant is a possibly nonlinear system,  $u$  (applied force) is the input,  $y$  (cart position) is the output, and

$$x = \begin{bmatrix} \text{cart position} \\ \text{cart velocity} \end{bmatrix}$$

is the state of the system. (We'll define state later.)

As a special case, suppose the air resistance is a linear function of velocity

$$D(x_2) = dx_2, \quad d \text{ is a real constant.}$$

Then  $f$  is linear function of  $x$ ,  $u$ :

$$f(x, u) = Ax + Bu, \quad A := \begin{bmatrix} 0 & 1 \\ 0 & -d/M \end{bmatrix}, \quad B := \begin{bmatrix} 0 \\ 1/M \end{bmatrix}.$$

Defining  $C := [1 \ 0]$ , we get the linear state model

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 0 & -\frac{d}{M} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix} u \\ y &= [1 \ 0] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \end{aligned} \quad (2.20)$$

In other words, when  $D(\dot{y})$  is linear, the state model has the structure

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

This model is of a linear, time-invariant system. ▲

Generalizing this example, we can say that an important class of models is

$$\begin{aligned} \dot{x} &= f(x, u), \quad f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n \\ y &= h(x, u), \quad h : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^p. \end{aligned} \quad (2.21)$$

This model is nonlinear, time-invariant. The input  $u$  has dimension  $m$ , the output  $y$  dimension  $p$ , and the state  $x$  dimension  $n$ . An example where  $m = 2, p = 2, n = 4$  is shown in Figure 2.30. In Figure 2.30 we have

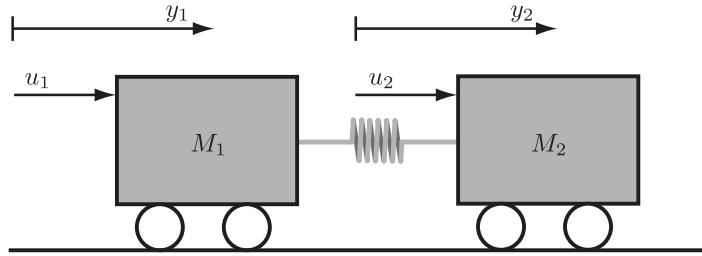


Figure 2.30: A multiple-input multiple-output example.

$$u = (u_1, u_2), \quad y = (y_1, y_2), \quad x = (y_1, \dot{y}_1, y_2, \dot{y}_2).$$

The LTI special case of (2.21) is

$$\begin{aligned} \dot{x} &= Ax + Bu, \quad A \in \mathbb{R}^{n \times n}, \quad B \in \mathbb{R}^{n \times m} \\ y &= Cx + Du, \quad C \in \mathbb{R}^{p \times n}, \quad D \in \mathbb{R}^{p \times m}. \end{aligned} \quad (2.22)$$

In this course we mainly deal with single-input single-output systems. Therefore we restrict ourselves to the case when  $m = 1$  and  $p = 1$ .

Now we turn to the concept of the **state of a system**. Roughly speaking,  $x(t_0)$  encapsulates all the system dynamics up to time  $t_0$ , that is, no additional prior information is required. More precisely, the concept is this: For any  $t_0$  and  $t_1$ , with  $t_0 < t_1$ , knowing  $x(t_0)$  and knowing  $\{u(t) : t_0 \leq t \leq t_1\}$ , we can compute  $x(t_1)$ , and hence  $y(t_1)$ .

**Example 2.4.2. (Choosing State Variables)** Consider the cart system in Figure 2.31 with no applied forces and no air resistance. If we were to try simply  $x = y$ , then knowing  $x(t_0)$  without  $\dot{y}(t_0)$ , we could not solve the initial value problem for the future cart position. Similarly  $x = \dot{y}$  won't work. Since the equation of motion,  $M\ddot{y} = 0$ , is second order, we need two initial conditions at  $t = t_0$ , implying we need a 2-dimensional state vector. In general for mechanical systems it is customary to take the vector  $x$  to consist of positions and velocities of all masses. ▲

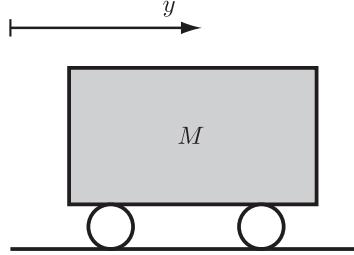


Figure 2.31: A simple cart with no air resistance, no applied force.

Even though this course is about continuous-time systems, it is instructive to look at the state of a simple discrete-time system.

**Example 2.4.3. (Digital Filter)** Consider a weighted averaging algorithm defined recursively by the difference equation

$$y[k] = 3y[k-1] + 2y[k-2] + u[k-2].$$

Here  $y[k] \in \mathbb{R}$  is the output of the algorithm at the  $k$ th iteration and the above equation represents the update rule. We can recursively solve for the output sequence  $\{y[k]\}_{k \geq 0}$  as long as we are given the initial values

$$y[-1], \quad y[-2]$$

and the input sequence  $\{u[k]\}_{k \geq -2}$ . The concept of a state extends to this system: The state vector  $x[k_0]$  encapsulates all the system dynamics up to time  $k_0$ . For any integers  $k_0$  and  $k_1$ , with  $k_0 < k_1$ , knowing  $x[k_0]$  and knowing  $\{u[k] : k_0 \leq k \leq k_1\}$ , we can compute  $x[k_1]$ , and hence  $y[k_1]$ . As in the previous example, neither  $y[k_0-1]$  nor  $y[k_0-2]$  on their own contain enough information to compute future values of  $y$ ; they aren't valid state vectors. You should check that a valid choice of state vector for this system is  $x[k] = (x_1[k], x_2[k]) := (y[k], y[k+1])$  and that the corresponding state-space model is

$$\begin{aligned} x_1[k+1] &= x_2[k] \\ x_2[k+1] &= 2x_1[k] + 3x_2[k] + u[k] \\ y[k] &= x_1[k]. \end{aligned}$$

▲

Why are state models useful? First, there's a rich control theory for them. This is covered in ECE488. You may have heard of the Kalman filter; it is based on a state model. Second, they give a convenient data type to store a linear model, namely, the matrices  $A, B, C, D$  which makes state models useful for computations. And third, they give a simple way to linearize, namely, compute Jacobians. More on this in Section 2.5.

**Example 2.4.4. (Mass-Spring-Damper)** Consider the system in Figure 2.32. The dynamic equation is<sup>6</sup>

$$M\ddot{y} = u + Mg - K(y - y_0) - b\dot{y}.$$

It's appropriate to take the position and velocity as state variables

$$x := (x_1, x_2), \quad x_1 := y, \quad x_2 := \dot{y}$$

and then we get the equations

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{1}{M}u + g - \frac{K}{M}x_1 + \frac{K}{M}y_0 - \frac{b}{M}x_2 \\ y &= x_1. \end{aligned}$$

---

<sup>6</sup>Here we are assuming that the spring is unstretched/uncompressed when  $y = y_0$  (cf. Example 2.3.1).

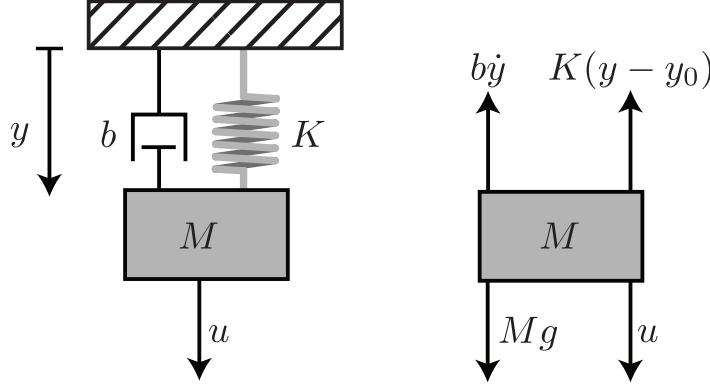


Figure 2.32: Another mass-spring-damper.

This has the form

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

where

$$A = \begin{bmatrix} 0 & 1 \\ -\frac{K}{M} & -\frac{b}{M} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ \frac{1}{M} \end{bmatrix}, \quad c = \begin{bmatrix} 0 \\ g + \frac{K}{M}y_0 \end{bmatrix}, \quad C = [1 \ 0].$$

The constant vector  $c$  is known, and hence is taken as part of the system rather than as a signal. This system is nonlinear. Can you see why? ▲

**Example 2.4.5. (RLC Circuit)** Recall the RLC circuit from Example 2.3.5.

$$-\dot{u} + R\dot{y} + L\ddot{y} + \frac{1}{C}y = 0.$$

It is natural to take the state variables to be voltage drop across  $C$  and current through  $L$  because together these variables characterize the energy stored in the circuit. Let

$$\begin{aligned}x_1 &:= \text{capacitor voltage} = \frac{1}{C} \int_0^t y(\tau) d\tau \\ x_2 &:= \text{inductor current} = y.\end{aligned}$$

Then the KVL equation becomes

$$-u + Rx_2 + L\dot{x}_2 + x_1 = 0$$

and the capacitor equation becomes

$$\dot{x}_1 = \frac{1}{C}x_2.$$

Thus

$$\begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} = \begin{bmatrix} 0 & \frac{1}{C} \\ -\frac{1}{L} & -\frac{R}{L} \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} u.$$

Since this is an LTI system, the state model of this system has the structure

$$\begin{aligned}\dot{x} &= Ax + Bu, & A &= \begin{bmatrix} 0 & \frac{1}{C} \\ -\frac{1}{L} & -\frac{R}{L} \end{bmatrix}, & B &= \begin{bmatrix} 0 \\ \frac{1}{L} \end{bmatrix} \\ y &= Cx + Du, & C &= \begin{bmatrix} 0 & 1 \end{bmatrix}, & D &= 0.\end{aligned}$$

**Example 2.4.6.** Recall the simplified car suspension system from Example 2.3.1

$$\begin{aligned} M_1\ddot{q}_1 + b_1\dot{q}_1 + k_1q_1 &= b_1\dot{q}_2 + k_1q_2 \\ M_2\ddot{q}_2 + b_1\dot{q}_2 + (k_1 + k_2)q_2 &= k_1q_1 + b_1\dot{q}_1 + k_2q_3. \end{aligned}$$

For mechanical systems, we take the position and velocities of the masses as our state variables. Namely, let

$$x := (x_1, x_2, x_3, x_4), \quad x_1 := q_1, \quad x_2 := \dot{q}_1, \quad x_3 := q_2, \quad x_4 := \dot{q}_2.$$

Let the input to the system be the road height  $u := q_3$ . Let the output of the system be the position of the chassis  $y := q_1$ . We get the equations

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{1}{M_1}(-k_1x_1 - b_1x_2 + k_1x_3 + b_1x_4) \\ \dot{x}_3 &= x_4 \\ \dot{x}_4 &= \frac{1}{M_2}(k_1x_1 + b_1x_2 - (k_1 + k_2)x_3 - b_1x_4 + k_2u) \\ y &= x_1. \end{aligned}$$

This has the form

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

where

$$A = \begin{bmatrix} 0 & 1 & 0 & 0 \\ -\frac{k_1}{M_1} & -\frac{b_1}{M_1} & \frac{k_1}{M_1} & \frac{b_1}{M_1} \\ 0 & 0 & 0 & 1 \\ \frac{k_1}{M_2} & \frac{b_1}{M_2} & -\frac{k_1+k_2}{M_2} & \frac{b_1}{M_2} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 0 \\ \frac{k_2}{M_2} \end{bmatrix}, \quad C = [1 \ 0 \ 0 \ 0].$$



**Example 2.4.7. (DC Motor)** We return to the DC motor from Example 2.3.7. The equations of motion are

$$\begin{aligned} J\ddot{\theta} &= -b\dot{\theta} + K_\tau i \\ -v + Ri + L\frac{di}{dt} + K_e\dot{\theta} &= 0. \end{aligned}$$

For state variables we take the position and velocities for the mechanical subsystem and the inductor current for the electrical subsystem

$$x := (x_1, x_2, x_3), \quad x_1 := \theta, \quad x_2 := \dot{\theta}, \quad x_3 := i.$$

Let the input to the system be the applied voltage  $u := v$ . Let the output of the system be the angular position of the motor's shaft  $y := \theta$ . We get the equations

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= \frac{1}{J}(-bx_2 + K_\tau x_3) \\ \dot{x}_3 &= \frac{1}{L}(-K_ex_2 - Rx_3 + u) \\ y &= x_1. \end{aligned}$$

This has the form

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

where

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & -\frac{b}{J} & \frac{K_T}{J} \\ 0 & -\frac{K_e}{L} & -\frac{R}{L} \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{L} \end{bmatrix}, \quad C = [1 \ 0 \ 0].$$



## 2.5 Linearization

Virtually *all* physical systems are nonlinear in nature. We saw that nonlinear systems can be modelled by equations of the form (2.21) which we re-write here

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

There might be disturbance signals present, but for now we suppose they are lumped into  $u$ . This state-space model includes the linear version (2.22) as a special case, i.e.,  $f(x, u) = Ax + Bu$ ,  $h(x, u) = Cx + Du$ , but also includes a much larger class of systems that includes robotic manipulators, mobile and humanoid robots, population dynamics, economic systems etc.

**Example 2.5.1. (Physical Pendulum)** Recall the governing ODE of the physical pendulum from Example 2.3.3

$$\ddot{\theta} = \frac{3}{M\ell^2}\tau - 1.5\frac{g}{\ell}\sin(\theta).$$

For state variables we take the position and velocity of the pendulum

$$x := (x_1, x_2), \quad x_1 := \theta, \quad x_2 := \dot{\theta}.$$

The input is the applied torque  $u := \tau$  and the output is taken to be the pendulum's angle  $y := \theta$ . We get the equations

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -1.5\frac{g}{\ell}\sin(x_1) + \frac{3}{M\ell^2}u \\ y &= x_1.\end{aligned}$$

This has the form (2.21) where

$$f(x, u) = \begin{bmatrix} x_2 \\ -1.5\frac{g}{\ell}\sin(x_1) + \frac{3}{M\ell^2}u \end{bmatrix}, \quad h(x, u) = x_1.$$

This system is nonlinear because  $f(x, u)$  contains the term  $\sin(x_1)$ ; you can't find constant matrices  $A, B$  such that  $f(x, u)$  equals  $Ax + Bu$ .



There are techniques for controlling nonlinear systems, but that's an advanced subject. However, many systems can be linearized about an equilibrium configuration. In other words, although almost every physical system contains nonlinearities, oftentimes its behaviour *within a certain operating range of an equilibrium configuration* can be reasonably approximated by that of a linear model. The linear approximation is then used as the basis for control design. One reason for approximating the nonlinear system (2.21) by a linear model is that, by doing so, one can apply rather simple and systematic linear control design techniques. Keep in mind, however, that a linearized model will only be a good approximation to the nonlinear model when the system operates in a sufficiently small range around an equilibrium configuration. In this section we see how to make these approximations. The idea is to use Taylor series.

**Example 2.5.2.** To review Taylor series, let's linearize the function  $y = f(x) = x^3$  about the point  $\bar{x} = 1$ . The Taylor series expansion is

$$\begin{aligned} f(x) &= \sum_{n=0}^{\infty} c_n(x - \bar{x})^n, \quad c_n = \frac{f^{(n)}(\bar{x})}{n!} \\ &= f(\bar{x}) + f'(\bar{x})(x - \bar{x}) + \frac{f''(\bar{x})}{2}(x - \bar{x})^2 + \dots . \end{aligned}$$

Taking only terms  $n = 0, 1$  gives

$$f(x) \approx f(\bar{x}) + f'(\bar{x})(x - \bar{x}),$$

that is, letting  $\bar{y} := f(\bar{x})$  denote the value of  $f$  at the nominal point  $\bar{x}$ ,

$$y - \bar{y} \approx f'(\bar{x})(x - \bar{x}).$$

Defining  $\delta y := y - \bar{y}$ ,  $\delta x := x - \bar{x}$ , we have the linearized function  $\delta y = f'(\bar{x})\delta x$ , or  $\delta y = 3\delta x$  in this case. Obviously, this approximation gets better and better as  $|\delta x|$  gets smaller and smaller. Figure 2.33 depicts the linearization in this example. ▲

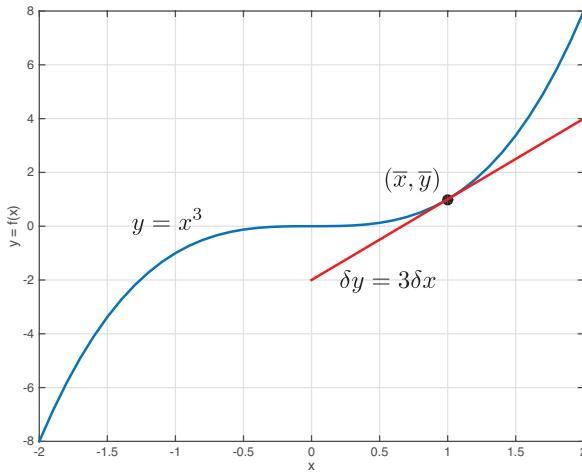


Figure 2.33: Linearizing  $y = x^3$  at  $\bar{x} = 1$ .

Taylor series extends to multivariable functions  $f : \mathbb{R}^n \rightarrow \mathbb{R}^m$ .

**Example 2.5.3.** Consider the function

$$f : \mathbb{R}^2 \rightarrow \mathbb{R}^3, \quad f(x_1, x_2) = \begin{bmatrix} f_1(x_1, x_2) \\ f_2(x_1, x_2) \\ f_3(x_1, x_2) \end{bmatrix} = \begin{bmatrix} x_1 + 1 \\ x_1 x_2 \\ x_1^2 - x_2 \end{bmatrix}.$$

We'll linearize the equation  $y = f(x)$  about a nominal point, say  $\bar{x} = (1, -1)$ . Let

$$\bar{y} := f(\bar{x}) = \begin{bmatrix} 2 \\ -1 \\ 2 \end{bmatrix}$$

denote the value of  $f$  at the nominal point  $\bar{x}$ .

The multivariable Taylor series expansion of  $f$  at the point  $\bar{x} = (1, -1)$  is

$$f(x) = f(\bar{x}) + \frac{\partial f}{\partial x} \Big|_{x=\bar{x}} (x - \bar{x}) + \text{higher order terms}$$

where

$$\begin{aligned} \frac{\partial f}{\partial x} \Big|_{x=\bar{x}} &= \text{Jacobian of } f \text{ evaluated at } \bar{x} \\ &= \begin{bmatrix} \frac{\partial f_1}{\partial x_1} & \frac{\partial f_1}{\partial x_2} \\ \frac{\partial f_2}{\partial x_1} & \frac{\partial f_2}{\partial x_2} \\ \frac{\partial f_3}{\partial x_1} & \frac{\partial f_3}{\partial x_2} \end{bmatrix}_{x=\bar{x}} \\ &= \begin{bmatrix} 1 & 0 \\ x_2 & x_1 \\ 2x_1 & -1 \end{bmatrix}_{x=(1,-1)} \\ &= \begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 2 & -1 \end{bmatrix}. \end{aligned}$$

Taking only the lower order terms in this expansion we have that

$$f(x) \approx f(\bar{x}) + \frac{\partial f}{\partial x} \Big|_{x=\bar{x}} (x - \bar{x}).$$

In other words

$$y - \bar{y} \approx \frac{\partial f}{\partial x} \Big|_{x=\bar{x}} (x - \bar{x}).$$

Defining the perturbations

$$\delta x := x - \bar{x}, \quad \delta y := y - \bar{y}$$

we have the linearized function

$$\delta y = A\delta x$$

where

$$A = \frac{\partial f}{\partial x} \Big|_{x=\bar{x}} = \begin{bmatrix} 1 & 0 \\ -1 & 1 \\ 2 & -1 \end{bmatrix}.$$

In what sense does this approximate the equation  $y = f(x)$  near  $x = \bar{x}$ ? The exact equation is

$$\delta y = f(\bar{x} + \delta x) - f(\bar{x})$$

and the linear approximation is

$$\delta y_a = A\delta x$$

so the error is  $\|\delta y - \delta y_a\|$ . It can be proved that if  $f$  is continuously differentiable, then there exists  $M > 0$  such that

$$\|\delta y - \delta y_a\| \leq M\|\delta x\|.$$

Thus the error  $\|\delta y - \delta y_a\|$  is arbitrarily small if  $\|\delta x\|$  is sufficiently small. ▲

By direct extension, if  $f : \mathbb{R}^n \times \mathbb{R}^m \rightarrow \mathbb{R}^n$ , then

$$f(x, u) \approx f(\bar{x}, \bar{u}) + \left. \frac{\partial f}{\partial x} \right|_{(\bar{x}, \bar{u})} \delta x + \left. \frac{\partial f}{\partial u} \right|_{(\bar{x}, \bar{u})} \delta u.$$

This idea extends to *differential* equations

$$\dot{x} = f(x, u). \quad (2.23)$$

**Definition 2.5.1.** Consider a system in state-variable form (2.23). A constant pair  $(\bar{x}, \bar{u})$  is called an **equilibrium configuration of** (2.23) if  $f(\bar{x}, \bar{u}) = (0, 0, \dots, 0)$ . The constant  $\bar{x}$  is called an **equilibrium point**.

**Example 2.5.4. (Equilibria of the Physical Pendulum)** Back to the pendulum example. Suppose we turn off the motor and apply no torque, i.e., we set  $u = \bar{u} = 0$ . Let's use the definition above to find all corresponding equilibria. To do this we must solve the equation  $f(\bar{x}, \bar{u}) = 0$  for  $\bar{x}$ . In other words, we must solve

$$\begin{aligned} \bar{x}_2 &= 0 \\ -1.5 \frac{g}{\ell} \sin(\bar{x}_1) &= 0 \end{aligned}$$

for  $\bar{x}_1, \bar{x}_2$ . Thus the equilibria  $(\bar{x}, \bar{u})$  of the pendulum with  $\bar{u} = 0$  are given by

$$\bar{x} = \begin{bmatrix} k\pi \\ 0 \end{bmatrix}, \quad k \text{ is an integer.}$$

Physically, this means that the pendulum is at equilibrium whenever the angle  $\theta$  is either 0 (pointed downward) or  $\pi$  (pointed upward), and the angular velocity  $\dot{\theta}$  is zero.

Now suppose that we turn on the motor in such a way that it produces the constant torque  $u = \bar{u} \neq 0$ . The corresponding equilibria satisfy  $f(\bar{x}, \bar{u}) = 0$ , i.e.,

$$\begin{aligned} \bar{x}_2 &= 0 \\ -1.5 \frac{g}{\ell} \sin(\bar{x}_1) + \frac{3}{M\ell^2} \bar{u} &= 0. \end{aligned}$$

This yields the equilibria

$$\bar{x} = \begin{bmatrix} \arcsin\left(\frac{2\bar{u}}{Mg\ell}\right) + 2\pi k \\ 0 \end{bmatrix}, \quad k \text{ is an integer.}$$

If we pick a position  $\bar{x}_1 \in (-\pi, \pi]$  and set  $u = \bar{u} = 0.5Mg\ell \sin(\bar{x}_1)$ , then the state

$$\bar{x} = \begin{bmatrix} \bar{x}_1 \\ 0 \end{bmatrix}$$

is an equilibrium point of the pendulum. Physically, this means that by imparting a suitable constant torque to the pendulum one can make the pendulum be at rest at any desired angle  $\bar{x}_1$ . For instance, by imparting the torque  $u = \bar{u} = 0.5Mg\ell$ , the configuration  $\bar{x}_1 = \pi/2, \bar{x}_2 = 0$  is an equilibrium of the pendulum. ▲

**Exercise 2.1.** Find an equilibrium configuration  $(\bar{x}, \bar{u})$  for system from Example 2.5.4 at which its output equals  $3\pi/4$ .

First, assume there is an equilibrium configuration  $(\bar{x}, \bar{u})$  of (2.23), that is, a constant solution  $x(t) = \bar{x}, u(t) = \bar{u}$ . Now consider the perturbations from the equilibrium configuration:

$$\delta x(t) = x(t) - \bar{x}, \quad \delta u(t) = u(t) - \bar{u}, \quad \delta x(t), \delta u(t) \text{ small.}$$

We have

$$\begin{aligned} \dot{x}(t) &= f(x(t), u(t)) \\ &= f(\bar{x}, \bar{u}) + A\delta x(t) + B\delta u(t) + \text{higher order terms} \end{aligned}$$

where

$$A := \left. \frac{\partial f}{\partial x} \right|_{(\bar{x}, \bar{u})}, \quad B := \left. \frac{\partial f}{\partial u} \right|_{(\bar{x}, \bar{u})}.$$

Since  $f(\bar{x}, \bar{u}) = 0$ , we have the linearized equation to be

$$\dot{\delta x} = A\delta x + B\delta u.$$

Similarly, the output equation  $y = h(x, u)$  linearizes to

$$\delta y = C\delta x + D\delta u,$$

where

$$C := \left. \frac{\partial h}{\partial x} \right|_{(\bar{x}, \bar{u})}, \quad D := \left. \frac{\partial h}{\partial u} \right|_{(\bar{x}, \bar{u})}.$$

## Summary

Linearizing  $\dot{x} = f(x, u)$ ,  $y = h(x, u)$ : Select, if one exists, an equilibrium configuration  $(\bar{x}, \bar{u})$ . Compute the four Jacobians,  $A, B, C, D$ , of  $f$  and  $h$  at the equilibrium point. Then the linearized system is

$$\begin{aligned} \dot{\delta x} &= A\delta x + B\delta u, \\ \delta y &= C\delta x + D\delta u. \end{aligned}$$

Under mild conditions (sufficient smoothness of  $f$  and  $h$ ), this linearized system is a valid approximation of the nonlinear one in a sufficiently small neighbourhood of the equilibrium configuration.

**Example 2.5.5. (Physical Pendulum)** Suppose that we want to control the physical pendulum

$$\begin{aligned} \dot{x}_1 &= x_2 \\ \dot{x}_2 &= -1.5 \frac{g}{\ell} \sin(x_1) + \frac{3}{M\ell^2} u \\ y &= x_1 \end{aligned}$$

near its upright position, i.e.,  $\bar{y} = \pi$ . In this example we compute a linear model that approximates the pendulum near this position. First we find the equilibrium configuration  $(\bar{x}, \bar{u})$  corresponding to the desired output  $\bar{y}$ . Using the results of Example 2.5.4 we get the equilibrium configuration (verify!)

$$(\bar{x}, \bar{u}) = \left( \begin{bmatrix} \pi \\ 0 \end{bmatrix}, 0 \right).$$

To get the linearization at  $(\bar{x}, \bar{u})$  we have to compute the four matrices  $A, B, C, D$ :

$$\begin{aligned} A &= \left. \frac{\partial f}{\partial x} \right|_{(\bar{x}, \bar{u})} = \begin{bmatrix} 0 & 1 \\ 1.5 \frac{g}{\ell} & 0 \end{bmatrix}, \quad B = \left. \frac{\partial f}{\partial u} \right|_{(\bar{x}, \bar{u})} = \begin{bmatrix} 0 \\ \frac{3}{M\ell^2} \end{bmatrix} \\ C &= \left. \frac{\partial h}{\partial x} \right|_{(\bar{x}, \bar{u})} = [1 \ 0], \quad D = \left. \frac{\partial h}{\partial u} \right|_{(\bar{x}, \bar{u})} = 0. \end{aligned}$$

The linearized model is

$$\begin{aligned}\dot{\delta x} &= \begin{bmatrix} 0 & 1 \\ 1.5\frac{g}{\ell} & 0 \end{bmatrix} \delta x + \begin{bmatrix} 0 \\ \frac{3}{M\ell^2} \end{bmatrix} \delta u \\ \delta y &= [1 \ 0] \delta x\end{aligned}$$

where  $\delta x = x - \bar{x}$ ,  $\delta u = u - \bar{u}$  and  $\delta y = y - \bar{y}$  measure how far the pendulum is from equilibrium. ▲

## 2.6 Simulation

Concerning the model

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

simulation involves numerically computing  $x(t)$  and  $y(t)$  given an initial state  $x(0)$  and an input  $u(t)$ . If the model is nonlinear, simulation requires an ODE solver, based on, for example, the Runge-Kutta method. Scilab and MATLAB have ODE solvers and very nice simulation GUIs called, respectively, Scicos and SIMULINK.

**Example 2.6.1.** Recall the model of the physical pendulum from Example 2.3.3. The MATLAB code below simulates this model for twenty seconds in the case  $M = 1$ ,  $\ell = 1$ ,  $g = 9.8$ ,  $\theta(0) = 0$ ,  $\dot{\theta}(0) = 0$  and  $u(t) = 1(t)$ . After solving the ODE, the code also plots two figures to visually display the results of the simulation.

```

1 options = odeset('RelTol',1e-12, 'abstol', 1e-12); % set options for ODE solver
2 t_span = [0 20]; % amount of time we will simulate for
3
4 x0 = [0;0]; % Set the initial conditions
5 M = 1; ell = 1; g = 9.8; % model parameters
6
7 % Now simulate the system
8 [t, x] = ode45(@examplesys, t_span, x0, options, M, g, ell); %% medium accuracy ODE ...
    solver, Runge-Kutta method
9
10 figure;
11 plot(t, x(:, 1), t, x(:, 2));
12 legend('$\theta(t)$', '$\dot{\theta}(t)$', 'Interpreter','latex', 'FontSize', 14);
13 xlabel('time (s)', 'Interpreter','latex', 'FontSize', 17);
14 grid on;
15
16 figure;
17 plot(x(:, 1), x(:, 2));
18 xlabel('$x_1(t) = \theta(t)$', 'Interpreter','latex', 'FontSize', 17);
19 ylabel('$x_2(t) = \dot{\theta}(t)$', 'Interpreter','latex', 'FontSize', 17);
20 grid on;
21
22 function dx = examplesys(t,x, M, g, ell)
23
24 x1 = x(1); %% theta
25 x2 = x(2); %% theta dot
26
27 u = 1; % control signal (applied torque)
28
29 f = [x2; -1.5*g/ell*sin(x1) + 3/M/ell^2*u]; % system vector field
30
31 dx = f; % The system model
32 end

```

## 2.7 The Laplace transform

The Laplace transform is the fundamental tool used in control systems to get to the frequency domain.<sup>7</sup> You have already met Laplace transforms in previous courses so we only have to give a brief review here.

In signal processing, the two-sided Laplace transform (LT) is used, but in control only the one-sided LT is used. Let  $f(t)$  be a signal (a real-valued function of time) defined for all  $t$  or just for  $t \geq 0$ . The one-sided **Laplace transform** of  $f(t)$  is

$$F(s) = \int_0^\infty f(t)e^{-st}dt. \quad (2.24)$$

Here  $s$  is a complex variable. We use  $\mathcal{L}(f)$  to denote the Laplace Transform of  $f$ . Normally, the integral converges for some values of  $s$  and not for others. That is, there is a **region of convergence**. It turns out that the **Region of Convergence (R.O.C.)** is always an open right half-plane, of the form  $\{s \in \mathbb{C} : \operatorname{Re}(s) > a\}$ . Within the ROC  $F(s)$  has no poles.

**Example 2.7.1.** The unit step:

$$f(t) = \mathbf{1}(t) = \begin{cases} 1 & , t \geq 0 \\ 0 & , t < 0. \end{cases}$$

Actually, the precise value at  $t = 0$  doesn't matter. The LT is

$$F(s) = \int_0^\infty e^{-st}dt = -\frac{e^{-st}}{s}\Big|_0^\infty = \frac{1}{s}$$

and the ROC is

$$\text{ROC} : \operatorname{Re}(s) > 0.$$

The same  $F(s)$  is obtained if  $f(t) = 1$  for all  $t$ , even  $t < 0$ . This is because the one-sided LT is oblivious to negative time. Notice that  $F(s)$  has a pole at  $s = 0$  on the western boundary of the ROC. ▲

The LT exists provided  $f(t)$  satisfies two conditions. The first is that it is piecewise continuous on  $t \geq 0$ . This means that, on any time interval  $(t_1, t_2)$ ,  $f(t)$  has at most a finite number of jumps, and between these jumps  $f(t)$  is continuous. A square wave has this property for example. The second condition is that it is of exponential order, meaning there exist constants  $M, c$  such that  $|f(t)| \leq M e^{ct}$  for all  $t \geq 0$ . This means that if  $f(t)$  blows up, at least there is some exponential that blows up faster. For example,  $\exp(t^2)$  blows up too fast.

**Example 2.7.2.** Some other examples: An exponential:

$$f(t) = e^{at}, \quad F(s) = \frac{1}{s-a}, \quad \text{ROC} : \operatorname{Re}(s) > a.$$

A sinusoid:

$$f(t) = \cos(\omega t) = \frac{1}{2} (e^{j\omega t} + e^{-j\omega t})$$

$$F(s) = \frac{s}{s^2 + \omega^2}, \quad \text{ROC} : \operatorname{Re}(s) > 0.$$

The LT thus maps a class of time-domain functions  $f(t)$  into a class of complex-valued functions  $F(s)$ . The mapping  $f(t) \mapsto F(s)$  is linear.

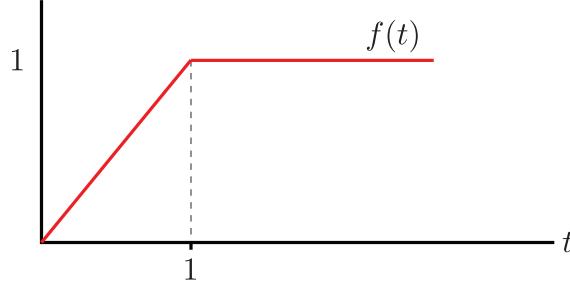


Figure 2.34: Time domain signal for Example 2.7.3.

**Example 2.7.3.** Let's use linearity to find the LT of the signal in Figure 2.34. We can write  $f = f_1 + f_2$ , where  $f_1(t) = t$  is the unit ramp starting at time 0 and  $f_2$  the ramp of slope  $-1$  starting at time 1. By linearity,  $F(s) = F_1(s) + F_2(s)$ . We compute that

$$\begin{aligned} F_1(s) &= \frac{1}{s^2}, \quad \operatorname{Re}(s) > 0 \\ F_2(s) &= -e^{-s} \frac{1}{s^2}, \quad \operatorname{Re}(s) > 0. \end{aligned}$$

Thus

$$F(s) = \frac{1 - e^{-s}}{s^2}, \quad \operatorname{Re}(s) > 0.$$



There are tables of LTs. So in practice, if you have  $F(s)$ , you can get  $f(t)$  using a table like Table 2.1. If

Table 2.1: Important (one-sided) Laplace transforms.

Description	Time domain $f(t)$	s-Domain $F(s)$
Unit step	$\mathbf{1}(t)$	$\frac{1}{s}$
Impulse	$\delta(t)$	1
Ramp	$t$	$\frac{1}{s^2}$
Exponential	$e^{at}$	$\frac{1}{s-a}$
Sine	$\sin(\omega t)$	$\frac{\omega}{s^2+\omega^2}$
Cosine	$\cos(\omega t)$	$\frac{s}{s^2+\omega^2}$
General exponential	$t^n e^{at}$	$\frac{n!}{(s-a)^{n+1}}$
Generalized sine	$e^{at} \sin(\omega t)$	$\frac{\omega}{(s-a)^2+\omega^2}$
Generalized cosine	$e^{at} \cos(\omega t)$	$\frac{s-a}{(s-a)^2+\omega^2}$
Sine with linear growth	$t \sin(\omega t)$	$\frac{2\omega s}{(s^2+\omega^2)^2}$
Cosine with linear growth	$t \cos(\omega t)$	$\frac{s^2-\omega^2}{(s^2+\omega^2)^2}$

$F(s)$  is a rational function, i.e.,

$$F(s) = \frac{N(s)}{D(s)}$$

<sup>7</sup>This is in contrast to communication theory where the Fourier transform dominates. Control problems frequently involve unstable systems, unbounded signals, and transient response requirements—all absent in communication problems.

where  $N, D \in \mathbb{R}[s]$  are polynomials and the degree of  $N$  is strictly less than the degree of  $D$ , then we can use partial fraction expansions and the linearity of the LT to obtain  $f(t)$ .

**Example 2.7.4.** Given  $F(s) = \frac{3s+17}{s^2-4}$ , let us find  $f(t)$ . We don't need the ROC to find  $f(t)$ , but actually we know what it is. Since we're using the one-sided LT the ROC must be a right half-plane, and because  $F(s)$  must be analytic within its ROC, the ROC of  $F(s)$  must be  $\text{Re}(s) > 2$ . We have

$$F(s) = \frac{3s+17}{s^2-4} = \frac{c_1}{s-2} + \frac{c_2}{s+2}, \quad c_1 = \frac{23}{4}, \quad c_2 = -\frac{11}{4}$$

and therefore

$$f(t) = c_1 e^{2t} + c_2 e^{-2t}, \quad t \geq 0.$$

We do not know if  $f(t)$  equals zero for  $t < 0$ . ▲

**Example 2.7.5.** Given

$$F(s) = \frac{s-2}{(s-1)(s+2)},$$

let us find  $f(t)$ . Once again, we don't need the ROC to find  $f(t)$ , but we can deduce that it is the right half-plane  $\text{Re}(s) > 1$ . We have

$$F(s) = \frac{s-2}{(s-1)(s+2)} = \frac{c_1}{s-1} + \frac{c_2}{s+2}, \quad c_1 = -\frac{1}{3}, \quad c_2 = \frac{4}{3}$$

and therefore

$$f(t) = -\frac{1}{3}e^t + \frac{4}{3}e^{-2t}, \quad t \geq 0.$$
▲

**Example 2.7.6.** Given

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

we write

$$F(s) = \frac{s+1}{s(s+2)^2} = \frac{c_1}{s} + \frac{c_2}{s+2} + \frac{c_3}{(s+2)^2}.$$

This implies

$$s+1 = c_1(s+s)^2 + c_2s(s+2) + c_3s.$$

Comparing coefficients we get three linear equations. You can verify that

$$c_1 = \frac{1}{4}, \quad c_2 = -\frac{1}{4}, \quad c_3 = \frac{1}{2}.$$

Therefore

$$f(t) = \frac{1}{4} - \frac{1}{4}e^{-2t} + \frac{1}{2}te^{-2t}, \quad t \geq 0.$$
▲

One use of the LT is in solving initial-value problems involving linear, constant-coefficient differential equations. In control engineering we do this by simulation. But let us look briefly at the LT method. We first observe that if we have the LT pair

$$f(t) \longleftrightarrow F(s)$$

and  $f$  is continuously differentiable at  $t = 0$ , then

$$\dot{f}(t) \longleftrightarrow sF(s) - f(0). \quad (2.25)$$

To prove this formula, start with the LT of  $\dot{f}(t)$  and integrate by parts:

$$\int_0^\infty e^{-st} \dot{f}(t) dt = e^{-st} f(t)|_0^\infty + s \int_0^\infty e^{-st} f(t) dt.$$

Now  $s$  is such that  $e^{-st} f(t)$  converges to 0 as  $t$  goes to  $\infty$ . Thus the right-hand side of the preceding equation becomes

$$-f(0) + sF(s).$$

This proves (2.25).

**Example 2.7.7.** Consider the initial-value problem

$$\dot{y} - 2y = t, \quad y(0) = 1.$$

The range of  $t$  for the differential equation isn't stated. Let us first assume that  $y(t)$  is continuously differentiable at  $t = 0$ . This implies the differential equation holds at least for  $-\varepsilon < t < \infty$  for some positive  $\varepsilon$ , that is, it holds for a little time before  $t = 0$ . Then we can apply (2.25) to get

$$sY(s) - 1 - 2Y(s) = \frac{1}{s^2}.$$

Solving for  $Y(s)$  we get

$$Y(s) = \frac{s^2 + 1}{s^2(s - 2)} = -\frac{1}{4s} - \frac{1}{2s^2} + \frac{5}{4(s - 2)}.$$

Therefore

$$y(t) = -\frac{1}{4} - \frac{1}{2}t + \frac{5}{4}e^{2t}. \quad (2.26)$$

Since we used the one-sided LT, which is oblivious to negative time, we can assert that (2.26) holds at least for  $t \geq 0$ . Note that it satisfies  $y(0) = 1$ .

On the other hand, suppose instead that the problem is posed as follows:

$$\dot{y} - 2y = t, \quad t > 0; \quad y(0) = 1.$$

That is, the differential equation holds for positive time and the initial value of  $y$  is 1. We aren't told explicitly that  $y(t)$  is continuously differentiable at  $t = 0$ , but we are justified in making that assumption, since any other solution, for example the one satisfying  $y(t) = 0$  for  $t < 0$ , satisfies (2.26) for  $t \geq 0$ .

▲

**Example 2.7.8.** Let us take the same system as in the previous example but modify the forcing term:

$$\dot{y} - 2y = 1, \quad y(0) = 1.$$

We assume  $y(t)$  is continuously differentiable at  $t = 0$  and the solution goes like this:

$$\begin{aligned} sY(s) - 1 - 2Y(s) &= \frac{1}{s} \\ Y(s) &= \frac{s + 1}{s(s - 2)} = -\frac{1}{2s} + \frac{3}{2(s - 2)} \\ y(t) &= -\frac{1}{2} + \frac{3}{2}e^{2t}, \quad t \geq 0. \end{aligned}$$

▲

If  $f(t)$  is twice continuously differentiable at  $t = 0$ , then

$$\ddot{f}(t) \longleftrightarrow s^2 F(s) - sf(0) - \dot{f}(0).$$

**Example 2.7.9.** The equation

$$\ddot{y} + 4\dot{y} + 3y = e^t, \quad y(0) = 0, \quad \dot{y}(0) = 2$$

can be solved as follows. We assume  $y(t)$  is twice continuously differentiable at  $t = 0$ . Then

$$s^2 Y(s) - 2 + 4sY(s) + 3Y(s) = \frac{1}{s-1}.$$

So

$$\begin{aligned} Y(s) &= \frac{2s-1}{(s-1)(s+1)(s+3)} \\ &= \frac{1}{8} \frac{1}{s-1} + \frac{3}{4} \frac{1}{s+1} - \frac{7}{8} \frac{1}{s+3} \\ y(t) &= \frac{1}{8} e^t + \frac{3}{4} e^{-t} - \frac{7}{8} e^{-3t}, \quad t \geq 0. \end{aligned}$$

The same solution would be arrived at for  $t > 0$  if, instead of assuming  $y(t)$  is twice continuously differentiable at  $t = 0$ , we were to allow jumps at  $t = 0$ . ▲

The LT of the product  $f(t)g(t)$  of two functions is *not* equal to  $F(s)G(s)$ , the product of the two transforms. Then what operation in the time domain does correspond to multiplication of the transforms? The answer is **convolution**. Let  $f(t), g(t)$  be defined on  $t \geq 0$ . Define a new function

$$h(t) = \int_0^t f(t-\tau)g(\tau)d\tau, \quad t \geq 0.$$

We say  $h$  is the convolution of  $f$  and  $g$ . Note that another equivalent way of writing  $h$  is

$$h(t) = \int_0^t f(\tau)g(t-\tau)d\tau, \quad t \geq 0.$$

We also frequently use the star notation  $h = f * g$  or  $h(t) = f(t) * g(t)$ .

**Theorem 2.7.1.** *The LT of  $f * g$  is  $F(s)G(s)$ .*

*Proof.* Let  $h := f * g$ . Then the LT of  $h$  is

$$H(s) = \int_0^\infty h(t)e^{-st}dt.$$

Substituting for  $h$  we have

$$H(s) = \int_0^\infty \int_0^t f(t-\tau)g(\tau)e^{-st}d\tau dt.$$

Now change the order of integration (draw the picture):

$$H(s) = \int_0^\infty \int_\tau^\infty f(t-\tau)e^{-st}dt g(\tau) d\tau.$$

In the inner integral change variables,  $r = t - \tau$ :

$$H(s) = \int_0^\infty \left( \int_0^\infty f(r)e^{-sr} dr \right) e^{-s\tau} g(\tau) d\tau.$$

Thus

$$H(s) = \int_0^\infty F(s)e^{-s\tau} g(\tau) d\tau.$$

Pull  $F(s)$  out and you get  $H(s) = F(s)G(s)$ . □

**Properties of Laplace transforms:** Let  $f(t)$  and  $g(t)$  be real-valued functions, continuously differentiable at  $t = 0$ , and let  $a$  be a real constant.

- (i)  $\mathcal{L}\{f + g\} = \mathcal{L}\{f\} + \mathcal{L}\{g\}$
- (ii)  $\mathcal{L}\{af\} = a\mathcal{L}\{f\}$
- (iii)  $\mathcal{L}\left\{\frac{df}{dt}\right\} = s\mathcal{L}\{f\} - f(0)$
- (iv)  $\mathcal{L}\{f * g\} = \mathcal{L}\{f\}\mathcal{L}\{g\}$
- (v)  $\mathcal{L}\left\{\int_0^t f(\tau)d\tau\right\} = \frac{1}{s}\mathcal{L}\{f\}$
- (vi)  $\mathcal{L}\{f(t - T)\} = e^{-sT}\mathcal{L}\{f\}, \quad T \geq 0.$

### 2.7.1 Fourier Transform

The **Fourier transform** of  $x(t)$  is

$$X(j\omega) = \int_{-\infty}^{\infty} x(t)e^{-j\omega t} dt.$$

We use  $\mathcal{F}(x)$  to denote the Fourier Transform of  $x$ . The inversion formula is

$$x(t) = \frac{1}{2\pi} \int_{-\infty}^{\infty} X(j\omega)e^{j\omega t} d\omega.$$

If the absolute value of  $x(t)$  is integrable, that is,

$$\int_{-\infty}^{\infty} |x(t)| dt < \infty$$

then  $X(j\omega)$  is a continuous function of  $\omega$ . An example is  $x(t) = e^{-|t|} \cos(t)$ . If the absolute value of  $x(t)$  is only square integrable, that is,

$$\int_{-\infty}^{\infty} |x(t)|^2 dt < \infty,$$

then  $X(j\omega)$  may not be a continuous function of  $\omega$ . An example is  $\sin(t)/t$ .

The constant signal that equals one for all time is neither absolutely integrable nor square integrable. Its **Fourier Transform (F.T.)** is defined to be  $2\pi\delta(\omega)$ . We can convince ourselves of this by noting that the inversion formula is an instance of the sifting formula

$$\frac{1}{2\pi} \int_{-\infty}^{\infty} 2\pi\delta(\omega)e^{j\omega t} d\omega = 1.$$

The forward FT is defined to be

$$\int_{-\infty}^{\infty} e^{-j\omega t} dt = 2\pi\delta(\omega)$$

and it has no meaning in the sense of ordinary functions. Likewise, the sinusoidal signal  $e^{j\omega_0 t}$  is neither absolutely integrable nor square integrable. Its FT is defined to be  $2\pi\delta(\omega - \omega_0)$ .

In general, FTs where either  $x(t)$  or  $X(j\omega)$  is not a function (i.e. has an impulse) must be treated with care to make sure that the result is correct.

## 2.7.2 Laplace vs Fourier

Here's an interesting question: Suppose you have a real-valued signal  $g(t)$ . Assume  $g(t) = 0$  for  $t < 0$ . Let  $G(s)$  be the Laplace transform of  $g(t)$  and then substitute  $j\omega$  for  $s$  in  $G(s)$ , so now you have  $G(j\omega)$ . Is  $G(j\omega)$  the Fourier transform of  $g(t)$ ? The answer is: not necessarily.

To illustrate this, for this section only, denote the LT of  $g(t)$  as  $G_{LT}(s)$  and the FT of  $g(t)$  as  $G_{FT}(j\omega)$ . Let's look at two examples.

**Example 2.7.10.** Consider the step function  $u(t) = \mathbf{1}(t)$ . The LT and ROC are

$$U_{LT}(s) = \frac{1}{s}, \quad \text{ROC} = \{s \in \mathbb{C} : \operatorname{Re}(s) > 0\}.$$

That is, the ROC is the open right half complex plane and  $U_{LT}(s)$  has a pole on the boundary of that region.

Next consider the causal signal  $g(t) = e^{-t}\mathbf{1}(t)$ . Then

$$G_{LT}(s) = \frac{1}{s+1}, \quad \text{ROC} = \{s \in \mathbb{C} : \operatorname{Re}(s) > -1\}.$$

Notice that the ROC contains the imaginary axis. ▲

**Example 2.7.11.** Let's turn to FTs and take the FT of the two signals from Example 2.7.10. The FT of the unit step is

$$U_{FT}(j\omega) = \int_0^{\infty} e^{-j\omega t} dt = \pi\delta(\omega) + \frac{1}{j\omega}.$$

Thus the FT is a distribution having an impulse. Notice that the FT and LT are not equivalent

$$U_{FT}(j\omega) \neq U_{LT}(j\omega).$$

Indeed, setting  $s = j\omega$  in  $U_{LT}(s)$  requires some justification since the imaginary axis is not in the ROC.

For  $g(t) = e^{-t}\mathbf{1}(t)$  we have

$$G_{FT}(j\omega) = \frac{1}{j\omega + 1}.$$

In this case the FT is a function. Furthermore, the FT and LT are equivalent

$$G_{FT}(j\omega) = G_{LT}(j\omega).$$

In general, the LT and FT are equivalent when the ROC of the LT includes the imaginary axis. ▲

## 2.8 Transfer functions

Linear time-invariant systems, and *only* LTI systems, have transfer functions. The **transfer function** of an LTI system is defined to be the ratio  $Y(s)/U(s)$  where the LTs are taken with zero initial conditions.

**Example 2.8.1. (Time-Varying System)** A system governed by the differential equation

$$\dot{y} + (\sin t)y = u.$$

does not have a transfer function—it isn't time invariant. ▲

**Example 2.8.2. (Low Pass Filter)** Recall the *RC* filter from Example 2.3.4

$$RC\dot{y} + y = u.$$

Apply Laplace transforms with zero initial conditions:

$$sRCY(s) + Y(s) = U(s).$$

Therefore the TF is

$$\frac{Y(s)}{U(s)} = \frac{1}{RCs + 1}.$$

This transfer function is **rational**, a ratio of polynomials. ▲

**Exercise 2.2.** Re-derive the TF from Example 2.8.2 using the voltage-divider rule and complex impedances.

**Example 2.8.3. (Mass-Spring-Damper)** A linear mass-spring-damper:

$$M\ddot{y} = u - Ky - b\dot{y}.$$

We get

$$\frac{Y(s)}{U(s)} = \frac{1}{Ms^2 + bs + K}.$$

This transfer function also is rational. ▲

**Example 2.8.4. (RLC Circuit)** In Example 2.3.5 we derived the governing ODE for an RLC circuit to be

$$-\dot{u} + R\dot{y} + L\ddot{y} + \frac{1}{C}y = 0.$$

Apply Laplace transforms with zero initial conditions:

$$-sU(s) + sRY(s) + s^2LY(s) + \frac{1}{C}Y(s) = 0.$$

The resulting TF is

$$\frac{Y(s)}{U(s)} = \frac{Cs}{LCs^2 + RCs + 1}.$$
▲

**Example 2.8.5. (DC Motor)** In Remark 2.3.1 we found the simplified model of a DC motor, assuming  $L \approx 0$ , to be

$$J\ddot{\theta} = -\left(b + \frac{K_\tau K_e}{R}\right)\dot{\theta} + \frac{K_\tau}{R}v.$$

If we now define

$$a_1 := \frac{1}{J} \left( b + \frac{K_\tau K_e}{R} \right), \quad b_0 := \frac{K_\tau}{JR},$$

then the TF from the voltage  $v$  to the shaft angle  $\theta$  is

$$\frac{\Theta(s)}{V(s)} = \frac{b_0}{s(s + a_1)}.$$

The TF from the voltage  $v$  to the shaft velocity  $\omega$  is

$$\frac{\Omega(s)}{V(s)} = \frac{b_0}{s + a_1}.$$



Table 2.2 lists other examples of transfer functions.

Table 2.2: Common Transfer Functions.

Description	Governing Equation	Transfer Function
Pure gain	$y(t) = u(t)$	1
Integrator	$\dot{y}(t) = u(t)$	$\frac{1}{s}$
Double integrator	$\ddot{y}(t) = u(t)$	$\frac{1}{s^2}$
Ideal differentiator	$y(t) = \dot{u}(t)$	$s$
Time delay	$y(t) = u(t - T), T > 0$	$e^{-sT}$ (not rational)
Prototype second order system	$\ddot{y}(t) + 2\zeta\omega_n\dot{y}(t) + \omega_n^2y(t) = K\omega_n^2u(t)$	$\frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$
Proportional-integral-derivative controller	$y(t) = K_p u(t) + K_i \int_0^t u(\tau) d\tau + K_d \dot{u}(t)$	$K_p + \frac{K_i}{s} + K_d s$

**Definition 2.8.1.** (i) A transfer function  $G(s)$  is **(real) rational** if it is the quotient of two polynomials

$$G(s) = \frac{b_m s^m + \dots + b_1 s + b_0}{s^n + a_{n-1} s^{n-1} + \dots + a_1 s + a_0}$$

where the coefficients  $a_i, b_i$  are real constants. The numbers  $m, n$  are the **degrees** of the numerator and denominator polynomials. Let  $\mathbb{R}(s)$  denote the set of rational functions in  $s \in \mathbb{C}$  with coefficients in  $\mathbb{R}$ .

(ii) A rational transfer function is **proper** if  $n \geq m$ . This is equivalent to the condition

$$\lim_{s \rightarrow \infty} G(s) \text{ exists in } \mathbb{C}.$$

(iii) A rational transfer function is **strictly proper** if  $n > m$ . This is equivalent to the condition

$$\lim_{s \rightarrow \infty} G(s) = 0.$$

(iv) A rational transfer function is **improper** if it is not proper.

It is understood, once and for all, that the ratio of polynomials

$$\frac{s+1}{s(s+2)}, \quad \frac{(s+1)(s-1)}{s(s+2)(s-1)}$$

represent the *same* rational function. In the first function the numerator and denominator polynomials are coprime<sup>8</sup> while in the second function these polynomials are not coprime. They represent the same rational function in the same way that the expressions  $\frac{5}{10}$  and  $\frac{1}{2}$  represent the same rational number.

**Definition 2.8.2.** A complex number  $p \in \mathbb{C}$  is a **pole** of a transfer function  $G(s)$  if

$$\lim_{s \rightarrow p} |G(s)| = \infty.$$

A complex number  $z \in \mathbb{C}$  is a **zero** of a transfer function  $G(s)$  if

$$\lim_{s \rightarrow z} G(s) = 0.$$

If  $G(s)$  is rational and proper and if its numerator and denominator polynomials are coprime, then the roots of the denominator are the poles of  $G(s)$  and the roots of the numerator are the zeros of  $G(s)$ . For example the functions

$$\frac{s+1}{s(s+2)}, \quad \frac{(s+1)(s-1)}{s(s+2)(s-1)}$$

have poles at  $s = 0$  and  $s = -2$  and zeros at<sup>9</sup>  $s = -1$ . In the second expression,  $s = 1$  is a root of the denominator but isn't a pole since the numerator and denominator aren't coprime.

### 2.8.1 Obtaining a transfer function from a state model

Let's find the transfer function for the LTI state model

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

Take Laplace transforms with zero initial conditions<sup>10</sup>

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

Eliminate  $X(s)$

$$\begin{aligned}(sI - A)X(s) &= BU(s) \\ \Rightarrow X(s) &= (sI - A)^{-1}BU(s) \\ \Rightarrow Y(s) &= \underbrace{\left( C(sI - A)^{-1}B + D \right) U(s)}_{\text{transfer function } G(s)}.\end{aligned}$$

<sup>8</sup>Two polynomials are coprime if they have no common roots.

<sup>9</sup>Since both functions approach zero as  $|s| \rightarrow \infty$  we sometimes say that the TFs have zeros at infinity. Unless otherwise stated, we only consider finite poles and zeros.

<sup>10</sup>Here  $X(s)$  denotes the component wise LT of the vector  $x(t)$ . That is, if  $x(t) = (x_1(t), \dots, x_n(t))$  then  $X(s) = (X_1(s), \dots, X_n(s))$ .

**Remark 2.8.3.** The transfer function obtained from a state model is always real rational and proper. We can see this if we write  $(sI - A)^{-1}$  using Cramer's rule as

$$(sI - A)^{-1} = \frac{1}{\det(sI - A)} \text{adj}(sI - A)$$

where  $\text{adj}(\cdot)$  is the classical adjoint of a matrix. The  $(i, j)$  entry of  $\text{adj}(sI - A)$  is  $(-1)^{i+j}$  times the determinant of the matrix obtained from  $(sI - A)$  by eliminating its  $j$ th row and  $i$ th column. The determinant  $\det(sI - A)$  is an element of  $\mathbb{R}[s]$  with degree  $n$ . The adjoint is formed by cofactors of the elements of the matrix. Therefore  $\text{adj}(sI - A)$  is a matrix whose entries are elements of  $\mathbb{R}[s]$ ; each of which has degree strictly less than  $n$ . Therefore  $(sI - A)^{-1}$  is a matrix in which each entry is a strictly proper element of  $\mathbb{R}(s)$ . It follows that  $G(s) = C(sI - A)^{-1}B + D \in \mathbb{R}(s)$  is proper. ♦

**Exercise 2.3.** Use Cramer's rule to show that the TF associated with a state-space model can also be computed using

$$\frac{Y(s)}{U(s)} = \frac{\det \begin{bmatrix} sI - A & B \\ -C & D \end{bmatrix}}{\det(sI - A)}.$$

This construction naturally leads to the converse question known as the **realization problem**: Given  $G(s)$ , find  $A, B, C, D$  such that

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

A solution exists if and only if  $G(s)$  is rational and proper. The solution is never unique.

## Summary

Let us recap our procedure for getting the transfer function of a system:

1. Apply the laws of physics or first principles to get differential equations governing the behaviour of the system. Put these equations in state form. In general these are nonlinear.
2. Find an equilibrium, if there is one. If there is more than one equilibrium, you have to select one. If there isn't even one, this method doesn't apply.
3. Linearize about the equilibrium point.
4. Take Laplace transform of the linearized system with zero initial state.
5. Solve for the output  $Y(s)$  in terms of the input  $U(s)$ .

The transfer function from input to output satisfies

$$Y(s) = G(s)U(s).$$

In general  $G(s)$  is a matrix. In the SISO case,  $G(s)$  is a scalar-valued transfer function.

**Example 2.8.6. (TF of the Linearized Pendulum)** In Example 2.5.1 we found the nonlinear state model of a physical pendulum (step 1). In Example 2.5.5 we found the pendulum's equilibrium configuration corresponding to being upright (step 2). In Example 2.5.5 we also linearized about the equilibrium configuration to obtain a LTI state model (step 3)

$$\begin{aligned} \dot{\delta}x &= \begin{bmatrix} 0 & 1 \\ 1.5\frac{g}{\ell} & 0 \end{bmatrix} \delta x + \begin{bmatrix} 0 \\ \frac{3}{M\ell^2} \end{bmatrix} \delta u \\ \delta y &= [1 \ 0] \delta x. \end{aligned}$$

We now find the system's transfer function (step 4, step 5) as

$$\begin{aligned}
 G(s) &= C(sI - A)^{-1}B + D \\
 &= \frac{1}{\det(sI - A)} C \operatorname{adj}(sI - A)B + D \\
 &= \frac{1}{s^2 - 1.5\frac{g}{\ell}} \begin{bmatrix} 1 & 0 \end{bmatrix} \begin{bmatrix} s & 1 \\ 1.5\frac{g}{\ell} & s \end{bmatrix} \begin{bmatrix} 0 \\ \frac{3}{M\ell^2} \end{bmatrix} \\
 &= \frac{3}{M\ell^2} \frac{1}{s^2 - 1.5\frac{g}{\ell}}.
 \end{aligned}$$



## 2.9 Block diagram manipulations

We take the point of view that a block diagram is a picture of a function (see Section 2.2). With this in mind, manipulating block diagrams becomes pretty straightforward. In this section we assume that each block represents a transfer function and that the inputs and outputs to each block are the LTs of the appropriate signal as in Figure 2.35. In that figure  $Y(s) = G(s)U(s)$  so  $Y(s)$  is a linear function of  $U(s)$ .

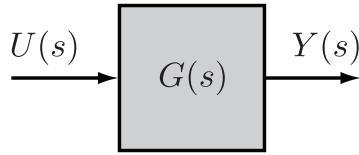


Figure 2.35: A block diagram of a transfer function.

### Series

Consider a system that is the **cascade connection** of systems with transfer functions  $G_1(s)$  and  $G_2(s)$ , as shown in Figure 2.36. Let  $V(s)$  denote the LT of the signal between the blocks. Then

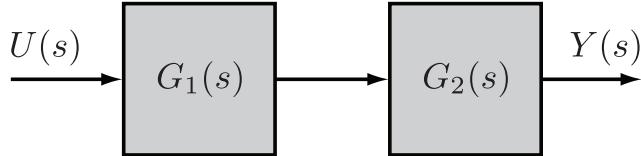


Figure 2.36: Cascade connection of transfer function.

$$V(s) = G_1(s)U(s), \quad Y(s) = G_2(s)V(s) \implies Y(s) = G_2(s)G_1(s)U(s).$$

The transfer function of the series connection is thus  $G(s) = G_2(s)G_1(s)$ . The order of the individual transfer functions is due to the fact that we place the input signal on the right-hand side of this expression, hence we first multiply by  $G_1$  and then by  $G_2$ . Unfortunately, this has the opposite ordering from the diagrams that we use, where we typically have the signal flow from left to right, so one needs to be careful. The ordering is important if either  $G_1$  or  $G_2$  is a vector-valued transfer function. If  $G_1(s)$  and  $G_2(s)$  are scalar TFs, then the order doesn't matter.

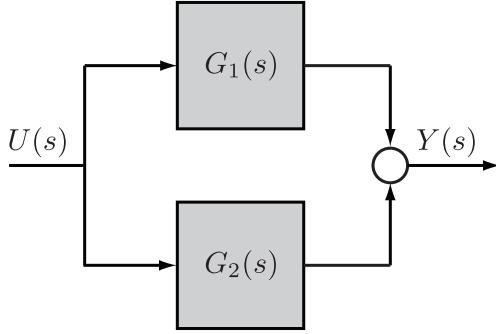


Figure 2.37: Parallel connection of transfer function.

### Parallel

Consider a system that is the parallel connection of systems with transfer functions  $G_1(s)$  and  $G_2(s)$ , as shown in Figure 2.37. Let  $V_1(s)$  denote the LT of the output of the top branch and let  $V_2(s)$  denote the LT of the output of the bottom branch. Then we have

$$V_1(s) = G_1(s)U(s), \quad V_2(s) = G_2(s)U(s) \implies Y(s) = V_1(s) + V_2(s) = (G_1(s) + G_2(s))U(s)$$

and the transfer function for a parallel connection is  $G(s) = G_1(s) + G_2(s)$ .

### Negative feedback

Consider the negative feedback connection of systems with transfer functions  $G_1(s)$  and  $G_2(s)$ , as shown in Figure 2.38. We have

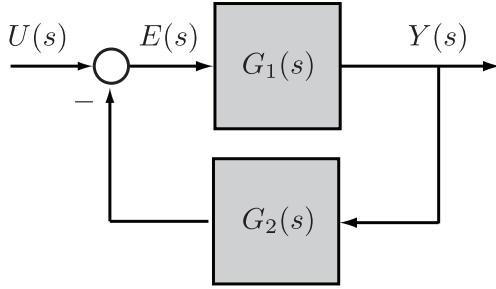


Figure 2.38: Negative feedback connection of transfer function.

$$\begin{aligned} E(s) &= U(s) - G_2(s)Y(s), \quad Y(s) = G_1(s)E(s) \implies Y(s) = G_1(s)(U(s) - G_2(s)Y(s)) \\ &\implies (1 + G_1(s)G_2(s))Y(s) = G_1(s)U(s) \\ &\implies Y(s) = \frac{G_1(s)}{1 + G_1(s)G_2(s)}U(s). \end{aligned}$$

Thus the transfer function for a negative feedback connection is  $G(s) = G_1(s)/(1 + G_1(s)G_2(s))$ . These three basic interconnections can be used as the basis for computing transfer functions for more complicated systems.

**Example 2.9.1. (Proportional error feedback for a first order plant)** Consider the block diagram in Figure 2.39. This is a negative feedback interconnection in which the TF  $G_1(s)$  is the series connection of  $K_p$  and  $K/(\tau s + 1)$  and  $G_2(s) = 1$ . Therefore the TF from  $r$  to  $y$  is

$$\frac{Y(s)}{R(s)} = \frac{K_p \frac{K}{\tau s + 1}}{1 + K_p \frac{K}{\tau s + 1}} = \frac{K_p K}{\tau s + 1 + K_p K}.$$

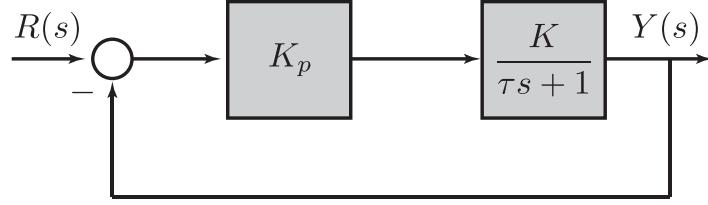


Figure 2.39: Proportional error feedback on a first order plant.

This TF has a pole at  $s = (-1 - K_p K)/\tau$  and therefore, by picking the proportional gain  $K_p$  appropriately, we can place the pole anywhere along the real axis.  $\blacktriangle$

### Moving blocks past summing junctions

It's not hard to convince yourself, by writing the expressions for  $Y(s)$ , that the two block diagrams in Figure 2.40 are equivalent.

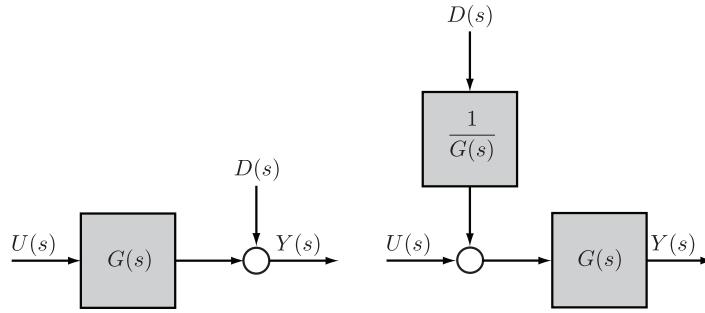


Figure 2.40: Changing the order of summing junctions and transfer functions.

**Example 2.9.2.** Find the transfer function from  $U(s)$  to  $Y(s)$  for the system in Figure 2.41. Our approach is

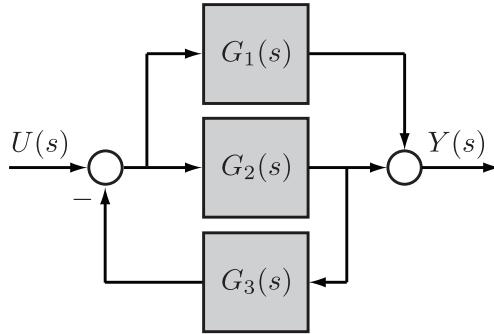


Figure 2.41: Example 2.9.2.

to move blocks around until common configurations are revealed. We start by moving the input to  $G_1(s)$  past the block  $G_2(s)$ . This yields the system in Figure 2.42. The feedback portion in Figure 2.42 has TF

$$\frac{G_2}{1 + G_2 G_3}.$$

The series portion has TF  $G_1/G_2$ . Therefore the parallel portion has TF

$$1 + \frac{G_1}{G_2}.$$

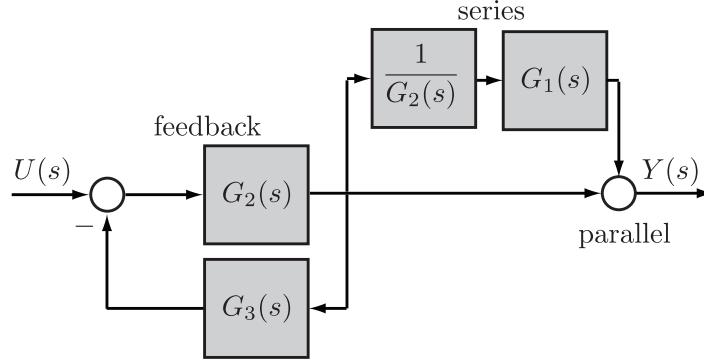


Figure 2.42: Rearranging the block diagram from Example 2.9.2.

The final TF from  $U$  to  $Y$  is the series connection of these TFs so we have

$$Y(s) = \left( \frac{G_2}{1 + G_2 G_3} \right) \left( 1 + \frac{G_1}{G_2} \right) U(s) = \frac{G_1 + G_2}{1 + G_2 G_3} U(s).$$

The approach used in Example 2.9.2 is *ad-hoc*; it would be hopeless for a really complicated interconnection of systems (and these do occur in practice). Since all of the functions in our block diagrams are linear, there is a more systematic way to do block diagram reduction. The systematic approach turns the block reduction problem into the problem of solving a set of linear equations.

1. Introduce new variables  $\{v_1, v_2, \dots\}$  representing the output of every summing junction.
2. Write down the expression for the inputs to each summing junction in terms of the signals  $\{u, y, v_1, v_2, \dots\}$  and the TF blocks.
3. Write the equation of each summing junction using the expressions from the previous step.
4. Eliminate the variables  $\{v_1, v_2, \dots\}$ .

**Example 2.9.3.** We re-do Example 2.9.2 using the above procedure. Consider the system in Figure 2.41.

1. Let  $v_1$  be the output of left-most summing junction. We don't need a variable for the summing junction on the right since it is  $y$ .
2. The inputs to the left-most summing junction are  $u$  and  $G_2 G_3 v_1$ . The inputs to the right-most summing junction are  $G_1 v_1$  and  $G_2 v_1$ .
3. The equations are

$$\begin{aligned} v_1 &= u - G_2 G_3 v_1 \\ y &= G_1 v_1 + G_2 v_1. \end{aligned}$$

Or, in matrix notation,

$$\begin{bmatrix} 1 + G_2 G_3 & 0 \\ G_1 + G_2 & -1 \end{bmatrix} \begin{bmatrix} v_1 \\ y \end{bmatrix} = \begin{bmatrix} u \\ 0 \end{bmatrix}.$$

4. We can solve for  $y$  easily since this is such a small problem but I'll use Cramer's rule since it scales better to larger problems. We have

$$Y(s) = \frac{\det \begin{bmatrix} 1 + G_2 G_3 & u \\ G_1 + G_2 & 0 \end{bmatrix}}{\det \begin{bmatrix} 1 + G_2 G_3 & 0 \\ G_1 + G_2 & -1 \end{bmatrix}} = \frac{-G_1 - G_2}{-1 - G_2 G_3} U(s) = \frac{G_1 + G_2}{1 + G_2 G_3} U(s).$$



**Example 2.9.4. (2019 Midterm)** Find the transfer function from  $r$  to  $y$  for the system shown in Figure 2.43.

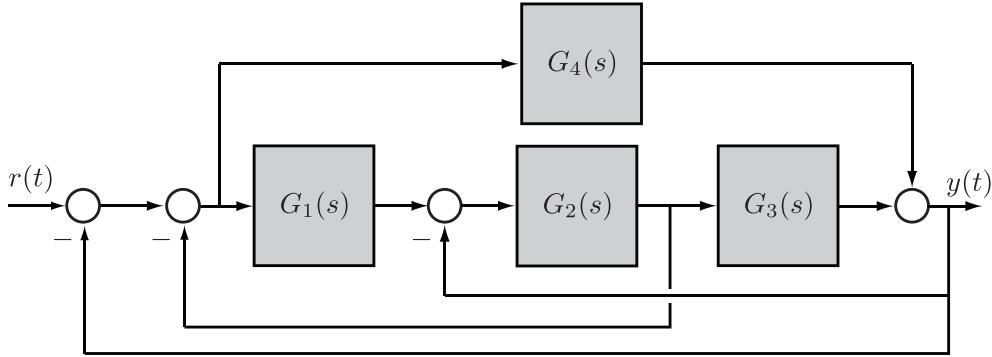


Figure 2.43: A complicated block diagram.

1. Label, from left to right, the outputs of the summing junctions as  $v_1$ ,  $v_2$  and  $v_3$ .
2. Working from left to right, the inputs to the left-most summing junction are  $r$  and  $y$ . The inputs to the next summing junction are  $v_1$  and  $G_2v_3$ . The inputs to the next summing junction are  $G_1v_2$  and  $y$ . The inputs to the right-most summing junction are  $G_4v_2$  and  $G_3G_2v_3$ .
3. The equations are

$$\begin{aligned} v_1 &= r - y \\ v_2 &= v_1 - G_2v_3 \\ v_3 &= G_1v_2 - y \\ y &= G_4v_2 + G_3G_2v_3. \end{aligned} \tag{2.27}$$

Or, in matrix notation,

$$\begin{bmatrix} 1 & 0 & 0 & 1 \\ 1 & -1 & -G_2 & 0 \\ 0 & G_1 & -1 & -1 \\ 0 & G_4 & G_3G_2 & -1 \end{bmatrix} \begin{bmatrix} v_1 \\ v_2 \\ v_3 \\ y \end{bmatrix} = \begin{bmatrix} r \\ 0 \\ 0 \\ 0 \end{bmatrix}.$$

4. In principle we can now solve for  $y$  using Cramer's rule except that no one likes finding the determinant of a  $4 \times 4$  matrix by hand. To make the problem tractable for hand calculations, let's make some simple substitutions in (2.27). Substitute the  $v_1$  expression into  $v_2$

$$v_2 = r - y - G_2v_3.$$

Now substitute this expression for  $v_2$  into the  $v_3$  and  $y$  equations

$$\begin{aligned} v_3 &= G_1(r - y - G_2v_3) - y \\ y &= G_4(r - y - G_2v_3) + G_3G_2v_3. \end{aligned}$$

In matrix notation

$$\begin{bmatrix} 1 + G_1G_2 & 1 + G_1 \\ G_4G_2 - G_3G_2 & 1 + G_4 \end{bmatrix} \begin{bmatrix} v_3 \\ y \end{bmatrix} = \begin{bmatrix} G_1r \\ G_4r \end{bmatrix}$$

Now we can easily apply Cramer's rule:

$$Y(s) = \frac{\det \begin{bmatrix} 1 + G_1G_2 & G_1R \\ G_4G_2 - G_3G_2 & G_4R \end{bmatrix}}{\det \begin{bmatrix} 1 + G_1G_2 & 1 + G_1 \\ G_4G_2 - G_3G_2 & 1 + G_4 \end{bmatrix}} = \frac{G_4 + G_1G_2G_3}{1 + G_4 + G_2(G_1 - G_4 + G_3 + G_1G_3)} R(s).$$



## 2.10 Summary

In this chapter we introduced the very important topic of modelling. The key things you should know after reading this chapter are as follows.

1. The usefulness of mathematical models and the key role they play in all branches of engineering. We presented various examples showing that there are different types of models (ODEs, PDEs, discrete-time).
2. No model is perfect.
3. If you can't draw the block diagram of a system, then you don't understand the system very well.
4. The basic building blocks of linear models for mechanical and electrical systems were introduced in Section 2.3. You should be comfortable modelling (deriving the governing ODEs) simple linear systems including those with electrical and mechanical subsystems.
5. In Section 2.4 we introduced state models which provide a standard way of expressing mathematical models. You should know what the state of a system is and how to convert an ODE model into a state model. You should know the difference between a nonlinear state model and a linear state model.
6. We learned how to linearize nonlinear state models in Section 2.5. You must be able to find equilibrium configurations of nonlinear state models and then find the system's linear state space approximation.
7. You must know the basic properties of a Laplace transform and be comfortable going from the time domain to the  $s$ -domain and vice versa.
8. Section 2.8 is very important because most of the systems we study hereafter are modelled by transfer functions. You should know what a rational transfer function is, what poles and zeros are, and what it means for a TF to be proper. You should be able to obtain a TF model for a nonlinear state model near an equilibrium configuration and you must be able to convert an LTI state model into a TF model.
9. You should understand why TF models are also called input-output models.
10. You should be able to reduce complex block diagrams into simpler forms using the basic rules in Section 2.9.

## 2.A Appendix: Linear functions

In this course we deal only with linear systems. Or rather, whenever we get a nonlinear system, we linearize it as soon as possible. So we had all better be very certain about what a linear function is. Let us recall even what a function is. If  $X$  and  $Y$  are two sets, a **function** from  $X$  to  $Y$  is a rule that assigns to every element of  $X$  an unambiguous element of  $Y$ —there cannot be two possible different values for some  $x$  in  $X$ . The terms function, mapping, and transformation are synonymous. The notation

$$f : X \rightarrow Y$$

means that  $f$  is a function from  $X$  to  $Y$ . We typically write  $y = f(x)$  for a function. To repeat, for each  $x$  there must be one and only one  $y$  such that  $y = f(x)$ ;  $y_1 = f(x)$  and  $y_2 = f(x)$  with  $y_1 \neq y_2$  is not allowed.

Let  $f$  be a function  $\mathbb{R} \rightarrow \mathbb{R}$ . This means that  $f$  takes a real variable  $x$  and assigns a real variable  $y$ , written  $y = f(x)$ . So  $f$  has a graph in the  $(x, y)$  plane. To say that  $f$  is linear means the graph is a straight line through the origin; there's only one straight line that is not allowed—the  $y$ -axis. Thus  $y = ax$  defines a linear function for any real constant  $a$ ; the equation defining the  $y$ -axis is  $x = 0$ . The function  $y = 2x + 1$  is not linear—its graph is a straight line, but not through the origin.

In your linear algebra course you were taught that a linear function is a function  $f$  from a *vector space*  $X$  to another (or the same) vector space  $Y$  having the property

$$f(a_1x_1 + a_2x_2) = a_1f(x_1) + a_2f(x_2)$$

for all vectors  $x_1, x_2$  in  $X$  and all real numbers<sup>11</sup>  $a_1, a_2$ . If the vector spaces are  $X = \mathbb{R}^n$ ,  $Y = \mathbb{R}^m$ , and if  $f$  is linear, then it has the form  $f(x) = Ax$ , where  $A$  is an  $m \times n$  matrix. Conversely, every function of this form is linear.

This concept extends beyond vectors to signals. For example, consider a capacitor, whose constitutive law is

$$i = C \frac{dv}{dt}.$$

Here,  $i$  and  $v$  are not constants, or vectors—they are functions of time. If we think of the current  $i$  as a function of the voltage  $v$ , then the function is linear. This is because

$$C \frac{d(a_1v_1 + a_2v_2)}{dt} = a_1C \frac{dv_1}{dt} + a_2C \frac{dv_2}{dt}.$$

On the other hand, if we try to view  $v$  as a function of  $i$ , then we have a problem, because we need in addition an initial condition  $v(0)$  (or some other initial time) to uniquely define  $v$ , not just  $i$ . Let us set  $v(0) = 0$ . Then  $v$  is a linear function of  $i$ . You can see this from the integral form of the capacitor equation:

$$v(t) = \frac{1}{C} \int_0^t i(\tau) d\tau.$$

## 2.B Appendix: The impulse and convolution

Now we take a little time to discuss the problematical object, the impulse  $\delta(t)$ . The impulse, also called the Dirac delta function, is not really a legitimate function, because its “value” at  $t = 0$  is not a real number. And you can't rigorously get  $\delta$  as the limit of a sequence of ever-narrowing rectangles, because that sequence does not converge in any ordinary sense. Yet the impulse is a useful concept in system theory and so we have to make it legitimate. The French mathematician Laurent Schwartz worked out a very nice, consistent way of dealing with the impulse, and more general “functions.” The impulse is an example of a **distribution**, sometimes called a **generalized function**.

The main idea is that  $\delta(t)$  is not a function, but rather it is a way of defining the linear transformation  $\phi \mapsto \phi(0)$  that maps a signal  $\phi(t)$  to its value at  $t = 0$ . This linear transformation should properly be written as  $\delta(\phi) = \phi(0)$  (i.e.,  $\delta$  transforms  $\phi$  to  $\phi(0)$ ) but historically it has been written as

$$\int_{-\infty}^{\infty} \phi(t) \delta(t) dt = \phi(0).$$

---

<sup>11</sup>This definition assumes that the vector spaces are over the field of real numbers.

You know this as the “sifting formula.” Let us emphasize that the expression

$$\int_{-\infty}^{\infty} \phi(t)\delta(t)dt \quad (2.28)$$

is not intended to mean integration of the product  $\phi(t)\delta(t)$  of functions— $\delta$  isn’t a function; rather, the expression is that of an operation on  $\phi$  whose value is defined to be  $\phi(0)$ . The expression is defined to be valid for all functions  $\phi(t)$  that are smooth at  $t = 0$ ; smooth means continuously differentiable up to every order.

Needless to say, we have to be careful with  $\delta$ ; for example, there’s no way to make sense of  $\delta^2$  because the expression (2.28) is not valid for  $\phi = \delta$ , again, because  $\delta$  isn’t a function, let alone a smooth one. Because the unit step is not smooth at  $t = 0$ ,  $\mathbf{1}(t)\delta(t)$  is undefined too. However, (2.28) does apply for  $\phi(t) = e^{-st}$ , because it is smooth:

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

Thus the LT of  $\delta$  equals 1.

Initial-value problems involving  $\delta$ , such as

$$\dot{y} + 2y = \delta,$$

or worse,

$$\dot{y} + 2y = \dot{\delta},$$

require more advanced theory, because  $y(t)$  cannot be continuously differentiable at  $t = 0$ . Instead of pursuing this direction, since initial-value problems are of minor concern in control we turn instead to convolution equations.

As you learned in signals and systems, linear time-invariant systems are modelled in the time domain by a convolution equation:

$$y(t) = \int_{-\infty}^{\infty} g(t - \tau)u(\tau)d\tau.$$

If  $g(t)$  is a smooth function for all  $t$ , then expression (2.28) applies and we get

$$g(t) = \int_{-\infty}^{\infty} g(t - \tau)\delta(\tau)d\tau.$$

Thus  $g(t)$  equals the output when the input is the unit impulse. We call  $g(t)$  the **impulse-response function**, or the impulse response. The case where  $g$  itself equals  $\delta$  isn’t covered by what has been said so far, but distribution theory can be used to justify  $\delta * \delta = \delta$ .

**Example 2.B.1.** Consider a low pass  $RC$  filter:

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

The inverse LT of  $G(s)$  is

$$\frac{1}{RC}e^{-\frac{t}{RC}}.$$

Since the filter is causal, to get the impulse response we should take that time-function that is zero for  $t < 0$ :

$$g(t) = \frac{1}{RC}e^{-\frac{t}{RC}}\mathbf{1}(t).$$

For the high pass filter,

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

$$g(t) = \delta(t) - \frac{1}{RC} e^{-\frac{t}{RC}} \mathbf{1}(t).$$



In general, if  $G(s)$  is strictly proper, then  $g(t)$  is a regular function. If  $G(s)$  is proper but not strictly proper, then  $g(t)$  contains an impulse.

As a final comment, some texts (Dorf and Bishop [Dorf and Bishop, 2011]) give the LT pair

$$\dot{f}(t) \longleftrightarrow sF(s) - f(0^-)$$

indicating that  $f(0)$  and  $f(0^-)$  may be different, that is,  $f(t)$  may be discontinuous at  $t = 0$ . If that is the case, then  $\dot{f}$  has an impulse and distribution theory needs to be invoked. For more details see [Lundberg et al., 2007].

### 2.B.1 Summary

Rigorously speaking, the impulse is not a function. Instead, it is defined only indirectly via the sifting formula. However, in the subject of control systems, and other signals and systems subjects, the impulse is used just as though it were a function. Usually no harm is done, i.e., you get the right answer. But sometimes questions arise that require a rigorous way to answer them; distribution theory is that way. For example, the derivative of the impulse,  $\dot{\delta}(t)$ , can be defined rigorously, and its LT is  $s$ . In this way we can obtain the impulse response of the differentiator  $y = \dot{u}$ —it is  $g(t) = \dot{\delta}(t)$ . The transfer function is  $G(s) = s$ .

# Chapter 3

---

## Linear system theory

In the preceding chapter we saw nonlinear state models and how to linearize them about an equilibrium point. A linearized system has the form (dropping the  $\delta$  notation)

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

or, using transfer functions,

$$\frac{Y(s)}{U(s)} = G(s) = C(sI - A)^{-1}B + D.$$

In this chapter we study such models.

The main concepts are internal stability, bounded-input bounded-output stability, steady-state gain and frequency response. We start with the time domain (state equations), then move into the  $s$ -domain (transfer functions) and conclude with the frequency domain.

## Contents

3.1 Initial-state response . . . . .	60
3.2 Input-response . . . . .	63
3.3 Total response . . . . .	65
3.4 Stability of state-space models . . . . .	66
3.5 Bounded-input bounded-output stability . . . . .	68
3.6 Steady-state gain . . . . .	73
3.7 Frequency response . . . . .	74
3.8 Graphical representations of the frequency response . . . . .	76
3.9 Summary . . . . .	90
3.A Appendix: Complex numbers . . . . .	90
3.B Appendix: Time domain, s-domain and frequency domain . . . . .	91

### 3.1 Initial-state response

Consider an LTI state-space model where the input is set to zero

$$\dot{x} = Ax, \quad x(0) = x_0, \quad A \in \mathbb{R}^{n \times n}.$$

Recall two facts

1. If  $n = 1$  then  $A$  is a scalar and the solution to the ODE is  $x(t) = e^{At}x_0$ .

2. The Taylor series expansion of  $e^t$  at  $t = 0$  is

$$e^t = 1 + t + \frac{t^2}{2} + \dots$$

and this series converges for all  $t$ . Therefore, when  $A$  is a scalar,

$$e^{At} = 1 + At + \frac{A^2 t^2}{2} + \dots$$

The second fact suggests the following definition.

**Definition 3.1.1.** The **matrix exponential** of  $A \in \mathbb{R}^{n \times n}$  is

$$e^A := \sum_{k=0}^{\infty} \frac{1}{k!} A^k = I + A + \frac{1}{2!} A^2 + \frac{1}{3!} A^3 + \dots$$

It can be proved that the matrix exponential converges for every matrix  $A$ . If  $A$  is  $n \times n$  then  $e^A$  is too; if  $A$  is not square, then  $e^A$  is undefined.

**Example 3.1.1.**

$$A = \begin{bmatrix} 0 & 0 \\ 0 & 0 \end{bmatrix}, \quad e^A = \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix}.$$

Notice that  $e^A$  is not obtained by taking the exponent of each element of  $A$ . ▲

**Example 3.1.2. (Diagonal Matrix)**

$$A = \begin{bmatrix} 3 & 0 \\ 0 & 1 \end{bmatrix}, \quad e^A = \begin{bmatrix} e^3 & 0 \\ 0 & e \end{bmatrix}.$$

When  $A$  is a diagonal matrix then  $e^A$  is obtained by taking the exponent of each diagonal element of  $A$ . ▲

**Example 3.1.3. (Nilpotent Matrix)**

$$A = \begin{bmatrix} 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix}.$$

This matrix has the property that  $A^3 = 0$ , i.e.,  $A$  is **nilpotent**<sup>1</sup>. Therefore the power series of  $e^A$  has only a finite number of non-zero terms

$$e^A = I + A + \frac{1}{2} A^2 = \begin{bmatrix} 1 & 1 & \frac{1}{2} \\ 0 & 1 & 1 \\ 0 & 0 & 1 \end{bmatrix}.$$
▲

**Exercise 3.1.** Compute  $e^A$  for

$$A = \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix}.$$

Using the matrix exponential we can define the matrix valued function  $\mathbb{R} \rightarrow \mathbb{R}^{n \times n}$  given by  $t \mapsto e^{tA}$ . It has the properties

---

<sup>1</sup>A square matrix  $A$  is nilpotent if there exists an integer  $k \geq 0$  such that  $A^k = 0$ . It is nilpotent if, and only if, all of its eigenvalues are zero.

1.  $e^{tA}|_{t=0} = I$  (identity matrix).
2.  $e^{t_1 A} e^{t_2 A} = e^{A(t_1+t_2)}$ . Note that  $e^{tA_1} e^{tA_2} \neq e^{t(A_1+A_2)}$ . Equality holds if and only if  $A_1$  and  $A_2$  commute, i.e.,  $A_1 A_2 = A_2 A_1$ .
3.  $(e^A)^{-1} = e^{-A}$  which implies that  $(e^{At})^{-1} = e^{-At}$ .
4.  $A$  and  $e^{At}$  commute.
5.  $\frac{d}{dt} e^{tA} = A e^{tA} = e^{tA} A$ .

**Theorem 3.1.2.** *The unique solution to  $\dot{x} = Ax$ ,  $x(0) = x_0$ ,  $A \in \mathbb{R}^{n \times n}$  is  $x(t) = e^{At}x_0$ .*

If we take the Laplace transform of the equation  $\dot{x} = Ax$  without setting the initial conditions to zero, we get that<sup>2</sup>

$$X(s) = (sI - A)^{-1}x_0.$$

Comparing this expression to the time-domain solution from Theorem 3.1.2 we have that  $e^{At}$  and  $(sI - A)^{-1}$  are Laplace transform pairs

$$\mathcal{L}\{e^{At}\} = (sI - A)^{-1}.$$

This fact can be used to compute  $e^{At}$  by hand for small problems ( $n \leq 3$ ).

**Example 3.1.4.** Calculate  $e^{tA}$  for the matrix

$$A = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}.$$

From above we know that  $e^{tA} = \mathcal{L}^{-1}\{(sI - A)^{-1}\}$ . We proceed to perform the necessary calculations:

$$sI - A = \begin{bmatrix} s & -1 \\ 2 & s+3 \end{bmatrix},$$

inverting a  $2 \times 2$  matrix is easy

$$\begin{aligned} (sI - A)^{-1} &= \frac{1}{s^2 + 3s + 2} \begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix} \\ &= \frac{1}{(s+1)(s+2)} \begin{bmatrix} s+3 & 1 \\ -2 & s \end{bmatrix}. \end{aligned}$$

Now use partial fractions on each entry of the above matrix

$$(sI - A)^{-1} = \begin{bmatrix} \frac{2}{s+2} - \frac{1}{s+1} & \frac{1}{s+1} - \frac{1}{s+2} \\ \frac{2}{s+1} + \frac{2}{s+2} & \frac{1}{s+1} + \frac{1}{s+2} \end{bmatrix}.$$

Find the inverse Laplace transform of each entry in  $(sI - A)^{-1}$  to obtain

$$e^{tA} = \begin{bmatrix} 2e^{-t} - e^{-2t} & e^{-t} - e^{-2t} \\ -2e^{-t} + 2e^{-2t} & -e^{-t} + 2e^{-2t} \end{bmatrix}.$$



**Exercise 3.2.** Compute  $e^{At}$  using the Laplace transform for the  $A$  matrix in Exercise 3.1.

Another way to compute  $e^{At}$  is by using eigenvalues and eigenvectors. We won't cover the general theory and instead present an example.

<sup>2</sup>Here  $X(s) = \mathcal{L}(x)$  is the component wise LT of the vector  $x$ . That is, if  $x(t) = (x_1(t), \dots, x_n(t))$  then  $X(s) = (X_1(s), \dots, X_n(s))$

**Example 3.1.5.**

$$A = \begin{bmatrix} 0 & 1 \\ -2 & -3 \end{bmatrix}.$$

The MATLAB command `[V, D] = eigs(A)` produces

$$V = \begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix}, \quad D = \begin{bmatrix} -1 & 0 \\ 0 & -2 \end{bmatrix}.$$

The eigenvalues of  $A$  are the diagonal elements of  $D$ . The columns of  $V$  are the corresponding eigenvectors. So, for example,

$$A \begin{bmatrix} 1 \\ -1 \end{bmatrix} = -1 \begin{bmatrix} 1 \\ -1 \end{bmatrix}.$$

It follows that  $AV = VD$  (verify this) and therefore that  $e^{At}V = Ve^{Dt}$  (prove this). The nice thing about  $e^{Dt}$  is that it is easy to compute since  $D$  is a diagonal matrix (cf. Example 3.1.2). In this case

$$e^{Dt} = \begin{bmatrix} e^{-t} & 0 \\ 0 & e^{-2t} \end{bmatrix}.$$

Then

$$e^{At} = Ve^{Dt}V^{-1} = \begin{bmatrix} 1 & -1 \\ -1 & 2 \end{bmatrix} \begin{bmatrix} e^{-t} & 0 \\ 0 & e^{-2t} \end{bmatrix} \begin{bmatrix} 2 & 1 \\ 1 & 1 \end{bmatrix} = \begin{bmatrix} 2e^{-t} - e^{-2t} & e^{-t} - e^{-2t} \\ -2e^{-t} + 2e^{-2t} & -e^{-t} + 2e^{-2t} \end{bmatrix}.$$

▲

The approach in Example 3.1.5 works when  $A$  has  $n$  linearly independent eigenvectors so that  $V$  is invertible. Otherwise the theory is more complicated and requires the **Jordan form** of the matrix  $A$ .

**Remark 3.1.3.** Examples 3.1.4 and 3.1.2 present two different methods of computing the function  $e^{At}$ . For large problems, both methods are tedious to apply by hand. Programs like MATLAB, Mathematica and Maple can do these calculations. The command in MATLAB is `expm`. ◆

## 3.2 Input-response

Now set the initial state to zero and consider the response from the input:

$$\dot{x} = Ax + Bu, \quad x(0) = 0. \quad (3.1)$$

We can use the properties of the matrix exponential to solve this equation. Here's a derivation of the solution [Chen, 1999]. Write

$$\dot{x} - Ax = Bu.$$

Multiply by  $e^{-tA}$ :

$$e^{-tA}\dot{x} - Ae^{-tA}x = e^{-tA}Bu.$$

The left-hand side equals  $\frac{d}{dt}[e^{-tA}x(t)]$ . Thus

$$\frac{d}{dt}[e^{-tA}x(t)] = e^{-tA}Bu.$$

Integrate from  $t = 0$  to  $t$ :

$$e^{-tA}x(t) - \underbrace{x(0)}_{=0} = \int_0^t e^{-\tau A}Bu(\tau)d\tau.$$

Multiply by  $e^{tA}$

$$x(t) = \int_0^t e^{(t-\tau)A} Bu(\tau) d\tau. \quad (3.2)$$

This is the solution of (3.1). It gives the value of the state and at time  $t$  as a function of  $u(\tau)$ ,  $0 \leq \tau \leq t$  when the initial state equals zero. Substituting the solution into the output equation leads to

$$y(t) = \int_0^t C e^{A(t-\tau)} Bu(\tau) d\tau + Du(t). \quad (3.3)$$

### Special case: Single-input single-output

In the special case when  $\dim u = \dim y = 1$ , i.e., the system is single-input, single-output, then  $D$  a scalar and so is  $y(t)$  in (3.3). If  $u = \delta$ , the unit impulse, then

$$y(t) = C e^{At} B \mathbf{1}(t) + D \delta(t),$$

where  $\mathbf{1}(t)$  denotes the unit step. We conclude that the **impulse response** of the system is

$$g(t) = C e^{At} B \mathbf{1}(t) + D \delta(t) \quad (3.4)$$

and equation (3.3) is a convolution equation

$$y(t) = (g * u)(t).$$

**Example 3.2.1.** Consider a single-input single output system in state-space form

$$\begin{aligned} \dot{x}(t) &= -3x(t) + 2u(t) \\ y(t) &= 4x(t). \end{aligned}$$

According to the development in this section, this system's input-response is given by the integral (convolution) equation (3.3) which for this example is

$$y(t) = \int_0^t 8e^{-3(t-\tau)} u(\tau) d\tau.$$

If we apply a unit step input  $u(t) = \mathbf{1}(t)$ , then

$$\begin{aligned} y(t) &= 8 \int_0^t e^{-3(t-\tau)} d\tau \\ &= 8e^{-3t} \int_0^t e^{3\tau} d\tau \\ &= \frac{8}{3} e^{-3t} (e^{3t} - 1) \\ &= \frac{8}{3} (1 - e^{-3t}). \end{aligned}$$



### 3.3 Total response

Consider the state equation forced by both an initial state and an input:

$$\dot{x} = Ax + Bu, \quad x(0) = x_0.$$

The system is linear in the sense that the state at time  $t$  equals the initial-state-response at time  $t$  plus the input-response at time  $t$

$$x(t) = e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau.$$

Similarly, the output is

$$y(t) = Ce^{At}x_0 + \int_0^t Ce^{A(t-\tau)}Bu(\tau)d\tau + Du(t).$$

These two equations constitute a solution in the time domain.

### Summary

We began with an LTI system modelled by a differential equation in state form:

$$\begin{aligned}\dot{x} &= Ax + Bu, \quad x(0) = x_0 \\ y &= Cx + Du.\end{aligned}$$

We solved the equations to get

$$\begin{aligned}x(t) &= e^{At}x_0 + \int_0^t e^{A(t-\tau)}Bu(\tau)d\tau \\ y(t) &= Ce^{At}x_0 + \int_0^t Ce^{A(t-\tau)}Bu(\tau)d\tau + Du(t).\end{aligned}\tag{3.5}$$

These are integral (convolution) equations giving  $x(t)$  and  $y(t)$  explicitly in terms of  $x_0$  and  $u(\tau)$ ,  $0 \leq \tau \leq t$ . In the SISO case, if  $x_0 = 0$  then

$$y = g * u = \int_0^t g(t-\tau)u(\tau)d\tau$$

where  $g(t) = Ce^{At} B \mathbf{1}(t) + D\delta(t)$ .

**Example 3.3.1. (Mass-Spring)** We illustrate these calculations in an example. We stress that normally time domain solutions are obtained through simulations. These calculations are useful for gaining intuition about the structure of the solution (3.5) and especially for understanding the stability arguments in the next section. Recall the mass-spring-damper system of Example 2.1.1 with linear damping

$$M\ddot{q} + b\dot{q} + kq = u.$$

Suppose  $M = 1$ ,  $k = 4$  and  $b = 0$ , i.e., no damping. The state equation is

$$\begin{aligned}\dot{x}_1 &= x_2 \\ \dot{x}_2 &= -4x_1 + u\end{aligned}$$

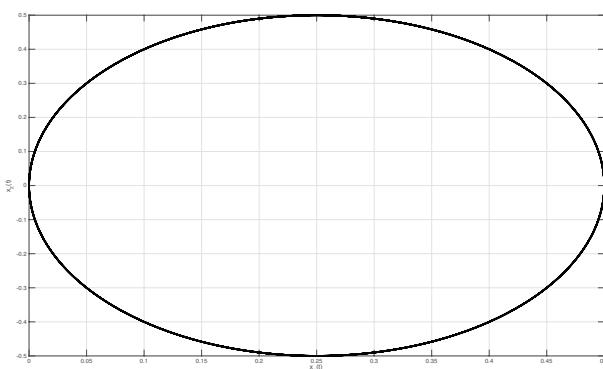
where  $(x_1, x_2) := (q, \dot{q})$ . We are interested in the solution to this system when  $u(t) = \mathbf{1}(t)$  and  $x_1(0), x_2(0)$  are arbitrary. We compute

$$e^{At} = \mathcal{L}^{-1}\{(sI - A)^{-1}\} = \mathcal{L}^{-1}\left\{\begin{bmatrix} s & -1 \\ 4 & s \end{bmatrix}^{-1}\right\} = \mathcal{L}^{-1}\left\{\begin{bmatrix} \frac{s}{s^2+4} & \frac{1}{s^2+4} \\ -\frac{4}{s^2+4} & \frac{s}{s^2+4} \end{bmatrix}\right\} = \begin{bmatrix} \cos(2t) & \frac{1}{2}\sin(2t) \\ -2\sin(2t) & \cos(2t) \end{bmatrix}.$$

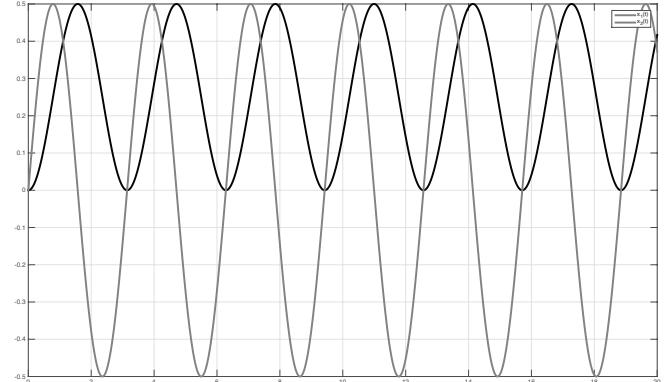
We can now compute

$$\begin{aligned} \begin{bmatrix} x_1(t) \\ x_2(t) \end{bmatrix} &= \begin{bmatrix} \cos(2t) & \frac{1}{2}\sin(2t) \\ -2\sin(2t) & \cos(2t) \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} + \int_0^t \begin{bmatrix} \cos(2(t-\tau)) & \frac{1}{2}\sin(2(t-\tau)) \\ -2\sin(2(t-\tau)) & \cos(2(t-\tau)) \end{bmatrix} \begin{bmatrix} 0 \\ \mathbf{1}(\tau) \end{bmatrix} d\tau \\ &= \begin{bmatrix} \cos(2t) & \frac{1}{2}\sin(2t) \\ -2\sin(2t) & \cos(2t) \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} + \int_0^t \begin{bmatrix} \frac{1}{2}\sin(2(t-\tau)) \\ \cos(2(t-\tau)) \end{bmatrix} d\tau \\ &= \begin{bmatrix} \cos(2t) & \frac{1}{2}\sin(2t) \\ -2\sin(2t) & \cos(2t) \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} + \int_0^t \begin{bmatrix} \frac{1}{2}\sin(2\lambda) \\ \cos(2\lambda) \end{bmatrix} d\lambda \quad (\text{change of variable } \lambda := t - \tau) \\ &= \begin{bmatrix} \cos(2t) & \frac{1}{2}\sin(2t) \\ -2\sin(2t) & \cos(2t) \end{bmatrix} \begin{bmatrix} x_1(0) \\ x_2(0) \end{bmatrix} + \begin{bmatrix} \frac{1}{4}(1 - \cos(2t)) \\ \frac{1}{2}\sin(2t) \end{bmatrix}. \end{aligned}$$

Figure 3.1 shows simulation results for this system when  $x_1(0) = 0$  and  $x_2(0) = 0$ . Figure 3.1a is a plot of  $(x_1(t), x_2(t))$  as they evolve in the plane. It is called a **phase plot** and the curve is called a **phase curve**. Figure 3.1b plots  $x_1$  and  $x_2$  versus time. If we are interested in controlling the position of the mass then we



(a) Phase curve.



(b) Position ( $x_1$ , black) & velocity ( $x_2$ , grey) vs. time.

Figure 3.1: Time domain response of a mass-spring system with zero initial conditions and unit step input.

take  $x_1$  as the system output while if we were interested in velocity control we take  $x_2$  as the output. ▲

## 3.4 Stability of state-space models

In this course there are two different notions of stability. The one we study in this section concerns state-space models. The stability question is: For the system

$$\dot{x} = Ax \tag{3.6}$$

will  $x(t)$  converge to 0 as  $t$  goes to infinity, for every  $x(0)$ ?

**Definition 3.4.1.** System (3.6) is **asymptotically stable** if  $x(t) \rightarrow 0$  as  $t \rightarrow \infty$  for every initial condition  $x(0)$ .

As we saw before, the trajectory of (3.6) is specified by  $x(t) = e^{At}x(0)$ . So asymptotic stability is equivalent to the condition that  $e^{At} \rightarrow 0$  as  $t \rightarrow \infty$ .

**Example 3.4.1.** Recall the matrix  $e^{At}$  for the mass-spring system from Example 3.3.1

$$e^{At} = \begin{bmatrix} \cos(2t) & \frac{1}{2}\sin(2t) \\ -2\sin(2t) & \cos(2t) \end{bmatrix}.$$

Since  $e^{At}$  does not approach zero as  $t \rightarrow \infty$  this system is not asymptotically stable. Physically this corresponds to the mass oscillating back and forth indefinitely if it is initialized with a non-zero initial position or velocity and no applied force. Our intuition suggests that friction should prevent such a phenomenon. Indeed, if we set  $b = 1$  so that there is a damper to model friction, then (cf. Example 3.3.1)

$$A = \begin{bmatrix} 0 & 1 \\ -4 & -1 \end{bmatrix}$$

and

$$e^{At} = \frac{e^{-\frac{t}{2}}}{15} \begin{bmatrix} 15\cos(\sqrt{15}t/2) + \sqrt{15}\sin(\sqrt{15}t/2) & 2\sqrt{15}\sin(\sqrt{15}t/2) \\ -8\sqrt{15}\sin(\sqrt{15}t/2) & 15\cos(\sqrt{15}t/2) - \sqrt{15}\sin(\sqrt{15}t/2) \end{bmatrix}.$$

In this case  $e^{At} \rightarrow 0$  as  $t \rightarrow \infty$  and the system is asymptotically stable. This is consistent with our physical intuition that friction should cause energy loss in the system eventually resulting in the mass returning to its rest position.  $\blacktriangle$

Fortunately, we do not need to compute  $e^{At}$  to determine whether or not (3.6) is asymptotically stable. Asymptotic stability only depends on the eigenvalues of  $A$ .

**Proposition 3.4.2.**  $e^{At} \rightarrow 0$  as  $t \rightarrow \infty$  if, and only if, every eigenvalue of  $A$  has negative real part.

*Proof.* If  $A$  has  $n$  linearly independent eigenvectors, then it can be diagonalized by a similarity transformation, so that

$$e^{At} = V e^{Dt} V^{-1}.$$

Then

$$\begin{aligned} e^{At} \rightarrow 0 &\iff e^{Dt} \rightarrow 0 \\ &\iff e^{\lambda_i t} \rightarrow 0, \forall i \\ &\iff \operatorname{Re} \lambda_i < 0, \forall i. \end{aligned}$$

The proof in the general case is more complicated but not more enlightening.  $\square$

**Example 3.4.2.** Re-visiting the systems from Example 3.4.1, when there is no damping

$$A = \begin{bmatrix} 0 & 1 \\ -4 & 0 \end{bmatrix}$$

and the eigenvalues of  $A$  are found by solving  $\det(sI - A) = 0$  to yield  $s = \pm j2$ . The eigenvalues have zero real part and therefore by Proposition 3.4.2 the system is not asymptotically stable.

When there is friction

$$A = \begin{bmatrix} 0 & 1 \\ -4 & -1 \end{bmatrix}$$

and the eigenvalues are  $-\frac{1}{2}(1 \pm j\sqrt{15})$ . By Proposition 3.4.2 the system is asymptotically stable. These conclusions are consistent with the observations made in Example 3.4.1.  $\blacktriangle$

## 3.5 Bounded-input bounded-output stability

Consider an LTI system with a single input, a single output, and a strictly proper rational transfer function. The model is therefore  $y = g * u$  in the time domain, or  $Y(s) = G(s)U(s)$  in the  $s$ -domain. We ask the question: Does a bounded input (BI) always produce a bounded output (BO)? First, let's define precisely what boundedness of a signal means.

**Definition 3.5.1.** Let  $u(t)$  be a real-valued signal defined for  $t \geq 0$ . We say  $u$  is **bounded** if there exists a constant  $b$  such that, for all  $t \geq 0$ ,  $|u(t)| \leq b$ .

Familiar bounded signals are steps  $u(t) = \mathbf{1}(t)$  and sinusoids  $u(t) = \sin(t)$ , but not ramps  $u(t) = t$  nor the exponential  $u(t) = e^t$ . The **least upper bound** of a signal  $u$  is denoted  $\|u\|_\infty$ . You can think of<sup>3</sup>  $\|u\|_\infty$  as  $\max_{t \geq 0} |u(t)|$ .

**Definition 3.5.2.** A linear time-invariant system with input  $u$  and output  $y$  is **Bounded-Input Bounded-Output (B.I.B.O.)** stable if every bounded input  $u$  produces a bounded output  $y$ , i.e.,

$$\|u\|_\infty \text{ finite} \Rightarrow \|y\|_\infty \text{ finite.}$$

The system is **unstable** if it is not BIBO stable.

**Example 3.5.1.** Consider a system with transfer function

$$G(s) = \frac{1}{s+2}.$$

Is this system BIBO stable? The impulse response of this system is

$$g(t) = \mathcal{L}^{-1}\{G(s)\} = e^{-2t}.$$

Then

$$\begin{aligned} y(t) &= (g * u)(t) \\ &= \int_0^t e^{-2\tau} u(t - \tau) d\tau. \end{aligned}$$

---

<sup>3</sup>To be more precise,  $\|u\|_\infty := \sup_{t \geq 0} |u(t)|$  where sup is the **supremum**.

This implies

$$\begin{aligned}
 |y(t)| &= \left| \int_0^t e^{-2\tau} u(t-\tau) d\tau \right| \\
 &\leq \int_0^t |e^{-2\tau} u(t-\tau)| d\tau \\
 &= \int_0^t e^{-2\tau} |u(t-\tau)| d\tau \\
 &\leq \int_0^t e^{-2\tau} d\tau \|u\|_\infty \\
 &\leq \frac{1}{2} \|u\|_\infty.
 \end{aligned}$$

In other words

$$\|y\|_\infty \leq \frac{1}{2} \|u\|_\infty$$

which means that every bounded  $u$  produces a bounded  $y$ . ▲

**Example 3.5.2.** Consider a system with transfer function

$$G(s) = \frac{1}{s-2}.$$

Is this system BIBO stable? The impulse response of this system is

$$g(t) = \mathcal{L}^{-1}\{G(s)\} = e^{2t}.$$

Then

$$\begin{aligned}
 y(t) &= (g * u)(t) \\
 &= \int_0^t e^{2\tau} u(t-\tau) d\tau.
 \end{aligned}$$

If we pick  $u(t) = \mathbf{1}(t)$  then

$$y(t) = \int_0^t e^{2\tau} d\tau = \frac{1}{2} (e^{2t} - 1), \quad t \geq 0$$

which is unbounded. ▲

**Remark 3.5.3.** BIBO stability asks that **every** bounded input produce a bounded output. If you pick one bounded input and the output happens to be bounded, you cannot conclude that the system is BIBO stable. Sometimes you have to be clever about how you choose bad inputs. In the above example the input

$$U(s) = \frac{s-2}{s(s+1)},$$

which corresponds to the bounded signal

$$u(t) = -2 + 3e^{-t}, \quad t \geq 0,$$

produces a bounded output. ♦

Fortunately, we do not need to compute convolution integrals to determine if a system is BIBO stable. BIBO stability only depends on the poles of  $G(s)$ .

**Theorem 3.5.4.** *Assume  $G(s)$  is strictly proper, rational. Then the following three statements are equivalent:*

1. *The system is BIBO stable.*
2. *The impulse-response function  $g(t)$  is absolutely integrable, i.e.,  $\int_0^\infty |g(t)|dt < \infty$ .*
3. *Every pole of the transfer function  $G(s)$  has negative real part.*

**Example 3.5.3. (Low Pass Filter)** Consider the  $RC$  filter from Example 2.8.2

$$G(s) = \frac{1}{RCs + 1}, \quad g(t) = \frac{1}{RC}e^{-t/RC}\mathbf{1}(t).$$

The transfer function  $G(s)$  has one pole at  $s = -\frac{1}{RC}$ . Since  $R$  is a resistance and  $C$  is a capacitance, they are both positive and so the pole has negative real part. According to the theorem, we expect that every bounded  $u$  produces a bounded  $y$ . Let's see

$$\begin{aligned} |y(t)| &= \left| \int_0^t g(\tau)u(t-\tau)d\tau \right| \\ &\leq \int_0^t |g(\tau)||u(t-\tau)|d\tau \\ &\leq \int_0^t |g(\tau)|d\tau \|u\|_\infty \\ &\leq \int_0^\infty |g(\tau)|d\tau \|u\|_\infty \\ &= \int_0^\infty \frac{1}{RC}e^{-\tau/RC}d\tau \|u\|_\infty \\ &= \|u\|_\infty. \end{aligned}$$

Thus  $\|y\|_\infty \leq \|u\|_\infty$  which confirms, as expected, that every bounded input produces a bounded output. ▲

**Example 3.5.4. (Integrator)** The integrator has

$$G(s) = \frac{1}{s}, \quad g(t) = \mathbf{1}(t).$$

According to the theorem, the system is not BIBO stable; there exists some bounded input that produces an unbounded output. For example the unit step  $u(t) = \mathbf{1}(t)$  is bounded, but it produces the output  $y(t) = t\mathbf{1}(t)$ , which is an unbounded ramp. It is not true that every bounded input produces an unbounded output, only that some bounded input does. For example, if the input is  $u(t) = (\sin t)\mathbf{1}(t)$ , then the output is bounded. ▲

Using Theorem 3.5.4 we conclude that  $\frac{1}{s+1}$ ,  $\frac{1}{(s+1)^2}$ ,  $\frac{1}{s^2+2s+1}$  are BIBO stable, but not  $\frac{1}{s}$  nor  $\frac{1}{s-1}$ .

The theorem can be extended to the case where  $G(s) \in \mathbb{R}(s)$  is only proper (and not strictly proper). Then write

$$G(s) = G_1(s) + c, \quad G_1(s) \text{ strictly proper, } c \in \mathbb{R}.$$

Then the impulse response has the form

$$g(t) = g_1(t) + c\delta(t).$$

Theorem 3.5.4 remains true with the second statement changed to say that  $g_1(t)$  is absolutely integrable. On the other hand, the next result shows that improper systems are never BIBO stable.

**Theorem 3.5.5.** *If  $G(s) \in \mathbb{R}(s)$  is improper, then  $G(s)$  is not BIBO stable.*

*Proof.* Write the transfer function  $G(s)$  as

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

where  $G_1(s) \in \mathbb{R}(s)$  is strictly proper and  $G_2(s) \in \mathbb{R}[s]$  has degree at least 1. For any input  $u$  the output  $y$  is, by linearity,  $y = y_1 + y_2$  where

$$Y_1(s) = G_1(s)U(s), \quad Y_2(s) = G_2(s)U(s).$$

If  $G_1(s)$  has poles with non-negative real parts, then the result follows from Theorem 3.5.4. So let's assume  $G_1(s)$  has all its poles in  $\mathbb{C}^-$  and consider the bounded input  $u(t) = \sin(t^2)$ . Since  $G_1$  has all poles in  $\mathbb{C}^-$ , the signal  $y_1$  is bounded. We'll now show that  $y_2$  is not bounded. Any derivative of  $u$  will involve terms polynomial in  $t$  and such terms will not be bounded as  $t \rightarrow \infty$ . But  $y_2$  is a linear combination of  $u$  and its derivatives so the result follows.  $\square$

Theorem 3.5.5 shows that the differentiator system  $G(s) = s$  is not BIBO stable.

### 3.5.1 Stability of state-space models and BIBO stability

Consider a single-input single-output system modelled by

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

or

$$\begin{aligned}Y(s)/U(s) &= G(s) \\ &= C(sI - A)^{-1}B + D \\ &= \frac{1}{\det(sI - A)}C \operatorname{adj}(sI - A)B + D.\end{aligned}$$

From the last expression we see that the poles of  $G(s)$  are contained in the eigenvalues of  $A$ . Thus

$$\text{asymptotic stability of state-space model} \implies \text{BIBO stability.}$$

Usually the poles of  $G(s)$  are identical to the eigenvalues of  $A$ , that is, the two polynomials

$$\det(sI - A), \quad C \operatorname{adj}(sI - A)B + D \det(sI - A)$$

have no common roots, i.e., they're coprime. In these cases, the two stability concepts are equivalent.

**Example 3.5.5. (Mass-Spring)** The  $A$  matrix for the mass-spring system from Example 3.3.1 was

$$A = \begin{bmatrix} 0 & 1 \\ -4 & 0 \end{bmatrix}.$$

The eigenvalues of  $A$  are found by solving

$$\det(sI - A) = s^2 + 4 = 0$$

to get  $s = \pm j2$ . By Proposition 3.4.2 this system is not asymptotically stable. The system's transfer function is given by  $G(s) = C(sI - A)^{-1}B + D$ . The matrices  $C, D$  depend on what we want to chose as the output. If we take the position of the mass as the output then

$$G(s) = [1 \ 0] \begin{bmatrix} \frac{s}{s^2+4} & \frac{1}{s^2+4} \\ -\frac{4}{s^2+4} & \frac{s}{s^2+4} \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{s^2+4}.$$

By Theorem 3.5.4 the system is not BIBO stable. In this case the two notions of stability are equivalent because the polynomials

$$\det(sI - A) = s^2 + 4$$

and

$$C \operatorname{adj}(sI - A)B + D \det(sI - A) = [1 \ 0] \begin{bmatrix} s & 1 \\ -4 & s \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = 1$$

are coprime.  $\blacktriangleleft$

**Example 3.5.6.** Consider the parallel interconnection in Figure 3.2. If we take the state vector to be  $x =$

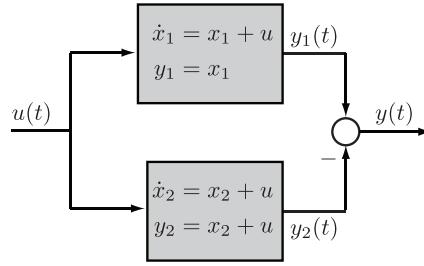


Figure 3.2: Parallel connection of first order systems.

$(x_1, x_2)$ , then the state-space model is

$$\begin{aligned} \dot{x} &= \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} x + \begin{bmatrix} 1 \\ 1 \end{bmatrix} u \\ y &= \begin{bmatrix} 1 & -1 \end{bmatrix} x - u. \end{aligned}$$

The eigenvalues of the  $A$  matrix are  $\{1, 1\}$  so that by Proposition 3.4.2 this system is not asymptotically stable. The system's transfer function is

$$G(s) = [1 \ -1] \begin{bmatrix} \frac{1}{s-1} & 0 \\ 0 & \frac{1}{s-1} \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} - 1 = -1.$$

By Theorem 3.5.4 the system is BIBO stable. In this case the two notions of stability are not equivalent because the polynomials

$$\det(sI - A) = (s - 1)^2$$

and

$$C \operatorname{adj}(sI - A)B + D \det(sI - A) = [1 \ -1] \begin{bmatrix} s-1 & 0 \\ 0 & s-1 \end{bmatrix} \begin{bmatrix} 1 \\ 1 \end{bmatrix} - (s-1)^2 = 1$$

have roots in common.  $\blacktriangleleft$

**Exercise 3.3.** Show that the two notions of stability presented are *not* equivalent for the system

$$\begin{aligned} \begin{bmatrix} \dot{x}_1 \\ \dot{x}_2 \end{bmatrix} &= \begin{bmatrix} 0 & 1 \\ 1 & 0 \end{bmatrix} \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u \\ y &= [1 \ -1] \begin{bmatrix} x_1 \\ x_2 \end{bmatrix}. \end{aligned}$$

## 3.6 Steady-state gain

In this section we show that if we apply a constant input to a BIBO stable system then the steady-state output is also constant. The ratio of the steady-state output to the constant input is also a constant, i.e., it doesn't depend on the magnitude of the input. The fundamental result used in this section is the **Final-Value Theorem (F.V.T.)**.

**Theorem 3.6.1** (Final-value theorem). *Let  $f(t)$  be a signal defined for  $t \geq 0$  and let its Laplace transform  $F(s)$  be rational and proper.*

(a) *If  $F(s)$  has all its poles in  $\mathbb{C}^-$ , then  $f(t)$  converges to zero as  $t \rightarrow \infty$ .*

(b) *If  $F(s)$  has all its poles in  $\mathbb{C}^-$  except for a simple pole at  $s = 0$ , then*

$$\lim_{t \rightarrow \infty} f(t) = \lim_{s \rightarrow 0} sF(s). \quad (3.7)$$

(c) *In all other cases,  $f(t)$  does not approach a constant as  $t \rightarrow \infty$ .*

Remember, you have to know that  $f(t)$  has a final value, by examining the poles of  $F(s)$ , before you can apply the final value theorem. Equation (3.7) is valid if  $sF(s)$  has no poles in the closed right half complex plane  $\overline{\mathbb{C}}^+$ .

**Theorem 3.6.2.** *If  $G(s)$  is BIBO stable and  $u(t) = b\mathbf{1}(t)$ ,  $b$  a real constant, then*

$$y_{ss} := \lim_{t \rightarrow \infty} y(t) = bG(0).$$

*Proof.* If  $u(t) = b\mathbf{1}(t)$  then

$$y(t) = \int_0^t g(\tau)u(t - \tau)d\tau = b \int_0^t g(\tau)d\tau$$

where  $g(t) = \mathcal{L}^{-1}\{G(s)\}$  is the impulse response. This implies that

$$y_{ss} = b \int_0^\infty g(\tau)d\tau = bG(0).$$

In the last deduction we used the definition of the Laplace Transform (2.24) with  $s = 0$ . We are allowed to make the substitution  $s = 0$  in (2.24) because  $G(s)$  is BIBO stable which means, by Theorem 3.5.4, that all its poles are to the left of the imaginary axis and hence  $s = 0$  is in the region of convergence of (2.24).  $\square$

**Exercise 3.4.** Re-prove Theorem 3.6.2 using the final-value theorem.

**Definition 3.6.3.** Let  $G(s)$  be a BIBO stable transfer function and let  $u(t) = b\mathbf{1}(t)$ ,  $b$  a real constant. The **steady-state gain** of  $G(s)$  is

$$\frac{y_{ss}}{b} = G(0).$$

**Exercise 3.5.** Using MATLAB, plot the unit step responses of

$$G_1(s) = \frac{20}{s^2 + 0.9s + 50}, \quad G_2(s) = \frac{-20s + 20}{s^2 + 0.9s + 50}.$$

They have the same steady-state gains and the same poles, but notice the big difference in transient response.

## 3.7 Frequency response

If you apply a sinusoidal input to a BIBO stable LTI system, then, in steady-state, the output is also a sinusoid of the same frequency with a possible change in magnitude and phase. This is illustrated in Figure 3.3.

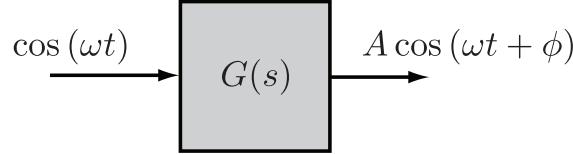


Figure 3.3: Steady-state output of a BIBO stable LTI system excited by a sinusoidal input.

To understand the above fact, consider a single-input, single-output LTI system. It will then be modelled by

$$y = g * u \quad \text{or} \quad Y(s) = G(s)U(s).$$

Let us assume  $G(s)$  is rational, proper, and has all its poles in  $\mathbb{C}^-$ . Then the system is BIBO stable by Theorem 3.5.4. The first fact we want to see is this: Complex exponentials are eigenfunctions<sup>4</sup>.

*Proof.*

$$u(t) = e^{j\omega t}, \quad y_{ss} = \int_{-\infty}^{\infty} g(t-\tau)u(\tau)d\tau.$$

Notice that the convolution integral used in this proof goes from  $t = -\infty$  to  $t = +\infty$ . That is, the input was applied starting at  $t = -\infty$ . If the time of application of the sinusoid is  $t = 0$ , there is a transient component in  $y(t)$  too. Taking the limits as above removes the transient terms. Continuing we have

$$\begin{aligned} y_{ss} &= \int_{-\infty}^{\infty} g(\tau)e^{j\omega(t-\tau)}d\tau \\ &= \int_{-\infty}^{\infty} g(\tau)e^{j\omega t}e^{-j\omega\tau}d\tau \\ &= \left( \int_{-\infty}^{\infty} g(\tau)e^{-j\omega\tau}d\tau \right) e^{j\omega t} \\ &= G(j\omega)e^{j\omega t}. \end{aligned}$$

□

Thus, if the input is the complex sinusoid  $e^{j\omega t}$ , then the steady-state output is the complex sinusoid

$$G(j\omega)e^{j\omega t} = |G(j\omega)|e^{j(\omega t + \angle G(j\omega))}. \quad (3.8)$$

Now consider the input<sup>5</sup>

$$u(t) = \cos(\omega t) = \frac{e^{j\omega t} + e^{-j\omega t}}{2}.$$

---

<sup>4</sup>The equation  $g * e^{j\omega t} = G(j\omega)e^{j\omega t}$  has the same form as the equation  $Ax = \lambda x$  where  $A$  is a square matrix,  $x$  is a vector and  $\lambda$  is a complex number. We say that  $\lambda$  is an eigenvalue of  $A$  and  $x$  is an eigenvector. Likewise, in the preceding equation we say that sinusoids are eigenfunctions of LTI systems and Fourier transforms  $G(j\omega)$  are eigenvalues.

<sup>5</sup>Here we are using Euler's formula  $e^{jx} = \cos(x) + j \sin(x)$ .

By the above analysis and linearity of the system, the steady-state output is<sup>6</sup>

$$\begin{aligned} y(t) &= \frac{1}{2}G(j\omega)e^{j\omega t} + \frac{1}{2}G(-j\omega)e^{-j\omega t} \\ &= \operatorname{Re}\{G(j\omega)e^{j\omega t}\}. \end{aligned}$$

Expressing  $G(j\omega)$  in polar form

$$G(j\omega) = |G(j\omega)| e^{\angle G(j\omega)}$$

we get that

$$y(t) = |G(j\omega)| \cos(\omega t + \angle G(j\omega)).$$

In conclusion, we have shown the following.

**Theorem 3.7.1.** *Assume  $G(s)$  is rational, proper, and has all its poles in  $\mathbb{C}^-$ . Then the steady-state response to the input  $u(t) = ae^{j\omega t}$  is*

$$y(t) = aG(j\omega)e^{j\omega t}. \quad (3.9)$$

*In particular*

- The steady-state response to the input  $u(t) = a \cos(\omega t)$  is

$$y(t) = a|G(j\omega)| \cos(\omega t + \angle G(j\omega)). \quad (3.10)$$

- The steady-state response to the input  $u(t) = a \sin(\omega t)$  is

$$y(t) = a|G(j\omega)| \sin(\omega t + \angle G(j\omega)). \quad (3.11)$$

- The steady-state response to the input  $u(t) = a\mathbf{1}(t)$  is

$$y(t) = aG(0). \quad (3.12)$$

Therefore the response of a BIBO stable LTI system to a continuous-time sinusoid input of frequency  $\omega$  is also a continuous-time sinusoid of the same frequency. The amplitude of the output sinusoid is  $|G(j\omega)|$  times the input amplitude, and the phase of the output is shifted by  $\angle G(j\omega)$  with respect to the input phase. This justifies the picture in Figure 3.3 and leads to the following definition.

**Definition 3.7.2.** Assume  $G(s)$  is rational, proper, and has all its poles in  $\mathbb{C}^-$ .

- (a) The function  $\mathbb{R} \rightarrow \mathbb{C}$ ,  $\omega \mapsto G(j\omega)$  is the **frequency response** of  $G$ .
- (b) The function  $\mathbb{R} \rightarrow \mathbb{R}$ ,  $\omega \mapsto |G(j\omega)|$  is the **amplitude or magnitude response** of  $G$ .
- (c) The function  $\mathbb{R} \rightarrow (-\pi, \pi]$ ,  $\omega \mapsto \angle G(j\omega)$  is the **phase response** of  $G$ .

Hence, the frequency response of a BIBO stable continuous-time LTI system  $G(s)$  is determined by setting  $s = j\omega$  and varying  $\omega$ . Here's the important point: Under the stated assumptions about the system ( $G(s)$  is rational, proper, and all poles in  $\mathbb{C}^-$ ), the region of convergence of the Laplace transform includes the imaginary axis. Therefore it is legitimate to set  $s = j\omega$  in  $G(s)$  to get  $G(j\omega)$ , which, by the way, equals the Fourier transform of the impulse response  $g(t)$ .

<sup>6</sup>Here we are using the fact that if  $G(s) \in \mathbb{R}(s)$ , then  $G(-j\omega) = \overline{G(j\omega)}$ . See Remark 3.8.1.

## 3.8 Graphical representations of the frequency response

One of the reasons why frequency response is a powerful tool is that it is possible to succinctly understand its primary features by means of plotting functions or parameterized curves. In this section we see how this is done.

**Example 3.8.1.** Consider a system with transfer function

$$G(s) = \frac{-10}{s + 10}.$$

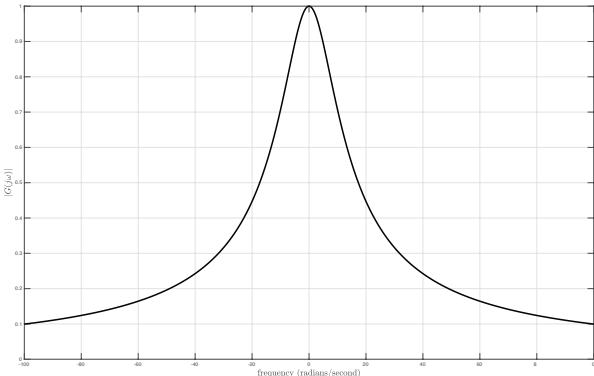
Its frequency response is the function

$$G(j\omega) = \frac{-10}{j\omega + 10}.$$

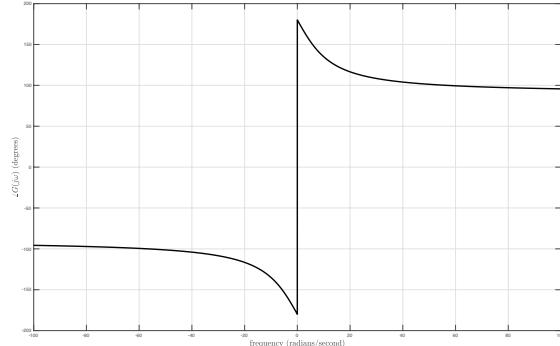
The magnitude and phase responses are plotted in Figure 3.4 for  $\omega \in [-100, 100]$ . The basic plots can be generated in MATLAB using the script below.

```

1 w = -100:0.01:100;
2 s = 1i.*w;
3 G = -10./(s+10);
4 plot(w, abs(G), 'LineWidth', 2); grid on;
5 figure; plot(w, rad2deg(angle(G)), 'LineWidth', 2); grid on;
```



(a) Magnitude response.



(b) Phase response.

Figure 3.4: A plot of the magnitude and phase response for Example 3.8.1.

**Remark 3.8.1.** From now on, when we graphically represent the functions from Definition 3.7.2 we will only consider non-negative values of  $\omega$ . One may think that we are losing information by not considering negative values of frequency. This is not the case because when  $G(s) \in \mathbb{R}(s)$  (real rational)<sup>7</sup>, then its magnitude is symmetric about  $\omega = 0$ , i.e.,  $|G(j\omega)| = |G(-j\omega)|$  and its phase is anti-symmetric about  $\omega = 0$ , i.e.,  $\angle(G(j\omega)) = -\angle(G(-j\omega))$ . See Figure 3.4. ♦

### 3.8.1 Polar plots

The idea here is to plot the frequency response  $G(j\omega)$  as a curve in the complex plane parametrized by  $\omega \in [0, \infty)$ , i.e., we draw  $\text{Im}(G(j\omega))$  vs  $\text{Re}(G(j\omega))$  as  $\omega$  ranges from 0 to  $\infty$ . The resulting curve is called a **polar plot** of the frequency response. While polar plots aren't used too much for design, they provide insight into the shape of a Bode plot and are fundamental to Nyquist plots (see Chapter 8). In anticipation of asymptotic Bode plots (Section 3.8.2), we present the polar plots for four basic transfer functions.

<sup>7</sup>The remark remains true when the transfer function has a time-delay, i.e., is of the form  $e^{-sT}G(s)$  with  $T > 0$  and  $G(s) \in \mathbb{R}(s)$ .

### Pure gain

Consider  $G(s) = K$ . Then  $G(j\omega) = K$  and the frequency response is constant. Figure 3.5 shows the polar plots.

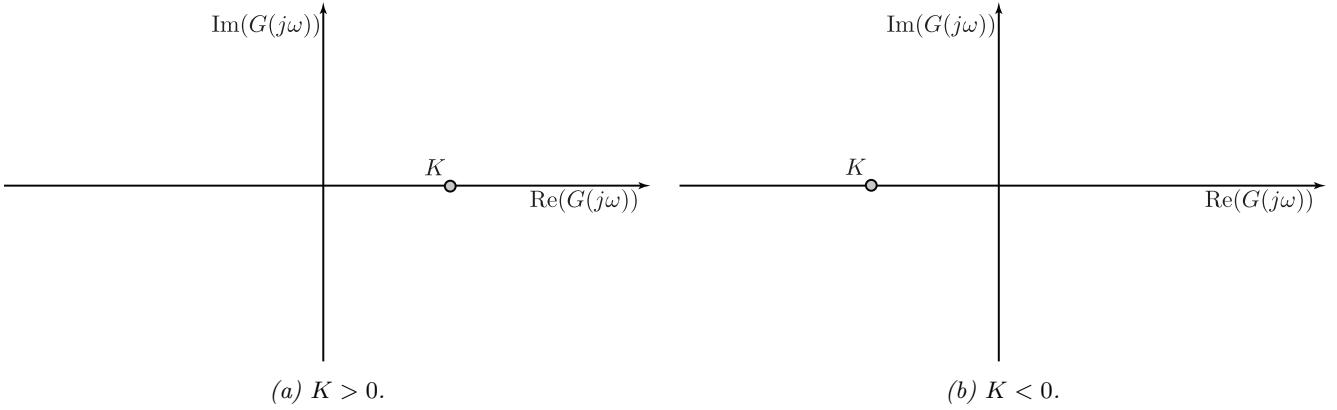


Figure 3.5: Polar plot of frequency response for  $G(s) = K$ .

### Zero at the origin

Consider  $G(s) = s^n$ . Then  $G(j\omega) = (j\omega)^n$ . When  $n = 1$ ,  $G(j\omega) = j\omega$  and is therefore an imaginary number. When  $n = 2$ ,  $G(j\omega) = -\omega^2$  is a negative real number. Continuing this way for different values of  $n$  we get the polar plot in Figure 3.6.

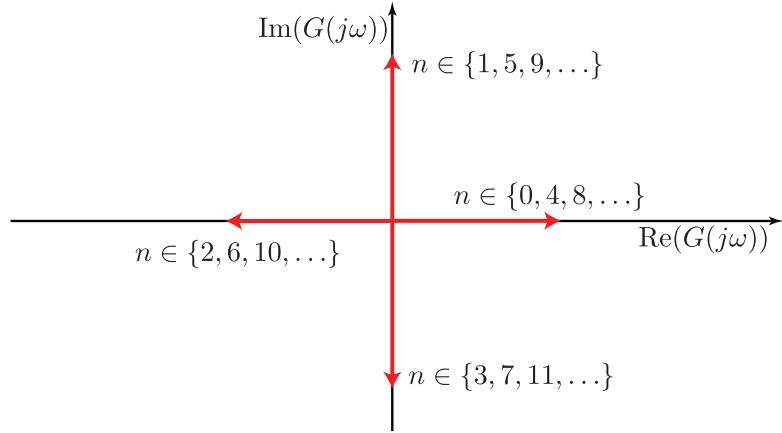


Figure 3.6: Polar plot of frequency response for  $G(s) = s^n$ .

### Real zero

Consider  $G(s) = \tau s \pm 1$  with  $\tau > 0$ . For  $\tau s + 1$  the zero is located at  $s = -\frac{1}{\tau}$  and lies in  $\mathbb{C}^-$ . For  $\tau s - 1$  the zero is located at  $s = \frac{1}{\tau}$  and lies in  $\mathbb{C}^+$ . The frequency response is  $G(j\omega) = j\tau\omega \pm 1$ . Figure 3.7 shows the polar plots for both cases.

### Complex conjugate zeros

Consider the case when  $G(s)$  has complex conjugate zeros

$$G(s) = s^2 + 2\zeta\omega_n s + \omega_n^2 = \omega_n^2 \left( \frac{s^2}{\omega_n^2} + 2\frac{\zeta}{\omega_n} s + 1 \right), \quad \zeta \in [0, 1], \quad \omega_n \neq 0. \quad (3.13)$$

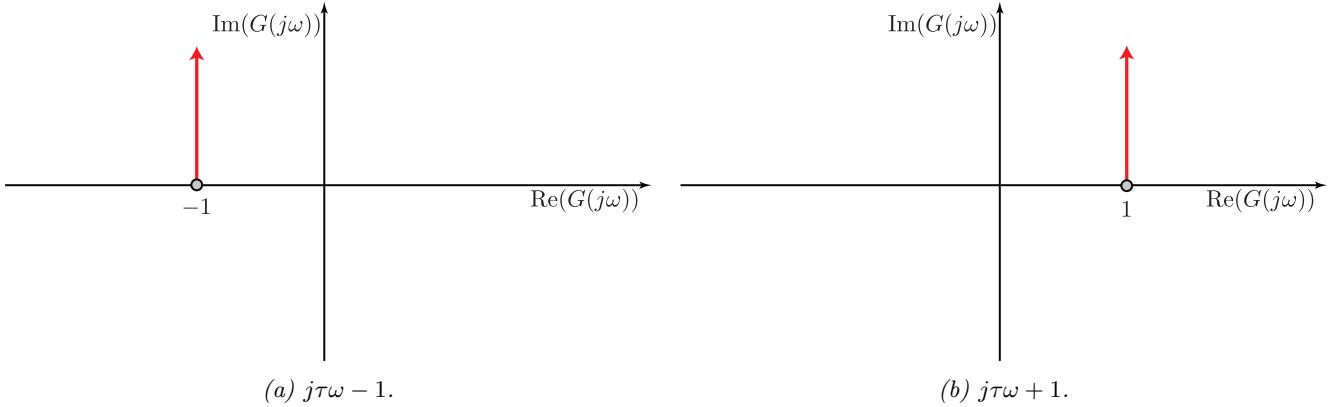


Figure 3.7: Polar plot of frequency response for  $G(s) = \tau s \pm 1$ ,  $\tau > 0$ .

We consider three sub-cases. The first is when  $\zeta = 0$  so that

$$G(s) = \omega_n^2 \left( \frac{s^2}{\omega_n^2} + 1 \right), \quad \zeta = 0, \omega_n \neq 0$$

has purely imaginary roots. Then

$$G(j\omega) = \omega_n^2 \left(1 - \frac{\omega^2}{\omega_n^2}\right)$$

is a real number and its polar plot is shown in Figure 3.8. Note that when  $\omega = \omega_n$ ,  $G(j\omega) = 0$  and when  $\omega$  changes sign the phase of  $G(j\omega)$  changes abruptly from  $0^\circ$  to  $180^\circ$ . The second sub-case is when  $G(s)$  has

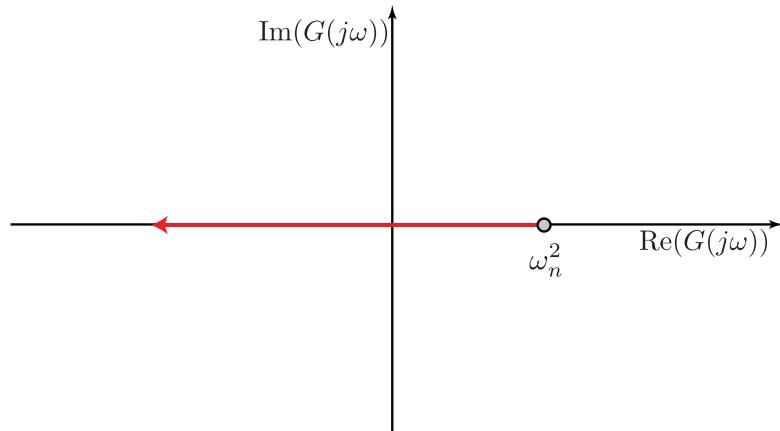


Figure 3.8: Polar plot of frequency response for  $G(s) = s^2 + \omega_n^2$ , purely imaginary roots.

complex conjugate roots in  $\mathbb{C}^-$

$$G(s) = \omega_n^2 \left( \frac{s^2}{\omega_n^2} + 2 \frac{\zeta}{\omega_n} s + 1 \right), \quad \zeta \in (0, 1), \quad \omega_n > 0.$$

In this case

$$G(j\omega) = \omega_n^2 \left( \left( 1 - \frac{\omega^2}{\omega_n^2} \right) + j2 \frac{\zeta}{\omega_n} \omega \right).$$

The polar plot of  $G(j\omega)$  is shown in Figure 3.9a. The last sub-case is when  $G(s)$  has complex conjugate roots in  $\mathbb{C}^+$

$$G(s) = \omega_n^2 \left( \frac{s^2}{\omega_n^2} - 2 \frac{\zeta}{\omega_n} s + 1 \right), \quad \zeta \in (0, 1), \omega_n > 0$$

for which

$$G(j\omega) = \omega_n^2 \left( \left( 1 - \frac{\omega^2}{\omega_n^2} \right) - j2\frac{\zeta}{\omega_n} \omega \right).$$

The polar plot of  $G(j\omega)$  is shown in Figure 3.9b. Note that when making a polar plot, the thing one loses is

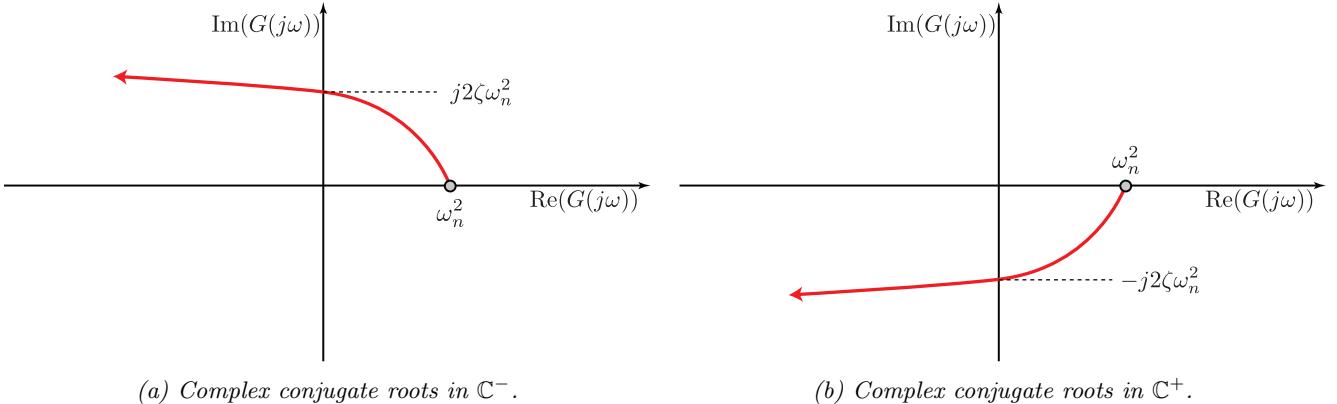


Figure 3.9: Polar plot of frequency response for  $G(s) = s^2 \pm 2\zeta\omega_n s + \omega_n^2$  for  $\zeta \in (0, 1)$ ,  $\omega_n > 0$ .

frequency information. That is, one can no longer read from the plot the frequency at which, say, the magnitude of the frequency response is maximum.

**Example 3.8.2. (Non-Minimum Phase)** Consider the two rational transfer functions

$$G_1(s) = \frac{1+s}{s^2+s+1}, \quad G_2(s) = \frac{1-s}{s^2+s+1}.$$

On the one hand, both transfer functions are BIBO stable since they both have poles at  $s = -\frac{1}{2}(1 \pm \sqrt{3}) \in \mathbb{C}^-$ . On the other hand,  $G_1(s)$  has a zero at  $s = -1 \in \mathbb{C}^-$  while  $G_2(s)$  has a zero at  $s = 1 \in \mathbb{C}^+$ . For reasons that will become apparent shortly, systems with zeros in  $\mathbb{C}^-$  are called **minimum phase** while systems with zeros in  $\mathbb{C}^+$  are called **non-minimum phase**. The polar plots of their frequency responses

$$G_1(j\omega) = \frac{1+j\omega}{-\omega^2+j\omega+1}, \quad G_2(j\omega) = \frac{1-j\omega}{-\omega^2+j\omega+1}$$

are shown in Figure 3.10. As the names suggest, the minimum phase system undergoes a smaller phase change as we move along its parameterized polar curve. ▲

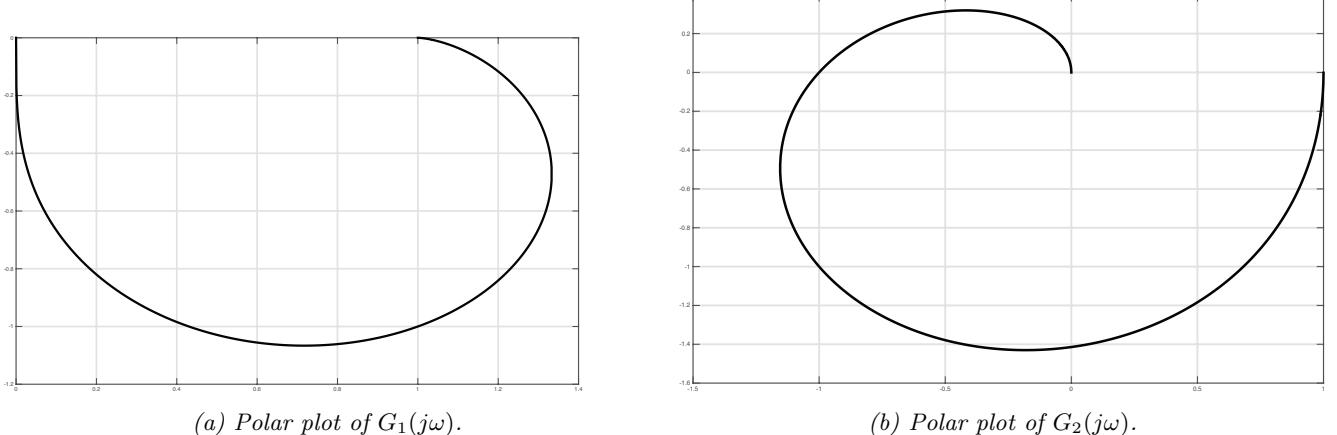


Figure 3.10: Polar plots of minimum and non-minimum phase transfer functions in Example 3.8.2.

**Definition 3.8.2.** A rational, proper transfer function is **minimum phase** if all its zeros have  $\operatorname{Re}(s) \leq 0$ . The transfer function is **non-minimum phase** if it is not minimum phase.

### 3.8.2 Bode plots

Control design is typically done in the frequency domain using Bode plots. A Bode plot is simply a plot of the amplitude and phase response functions of  $G$ , see Definition 3.7.2. These plots, however, are done in a very particular way: we plot  $20 \log |G(j\omega)|$  vs  $\omega$  and  $\angle G(j\omega)$  vs  $\omega$  with  $\omega$  plotted on log scales (all logarithms are base 10 in this course). The two plots (i)  $20 \log |G(j\omega)|$  vs  $\log(\omega)$  and (ii)  $\angle G(j\omega)$  vs  $\log(\omega)$  together are called the **Bode plot** of  $G(s)$ . The units of the plot of  $20 \log |G(j\omega)|$  are called **decibels** (dB).

A Bode plot tells us how the system  $G(s)$  responds in steady-state to sinusoidal inputs. Since the Fourier series can be used to represent a large class of signals, Bode plots can be used to tell us about a system response to almost all input signals.

**Example 3.8.3. (RLC Circuit)** In this example we'll use software to create the Bode plot of the RLC circuit

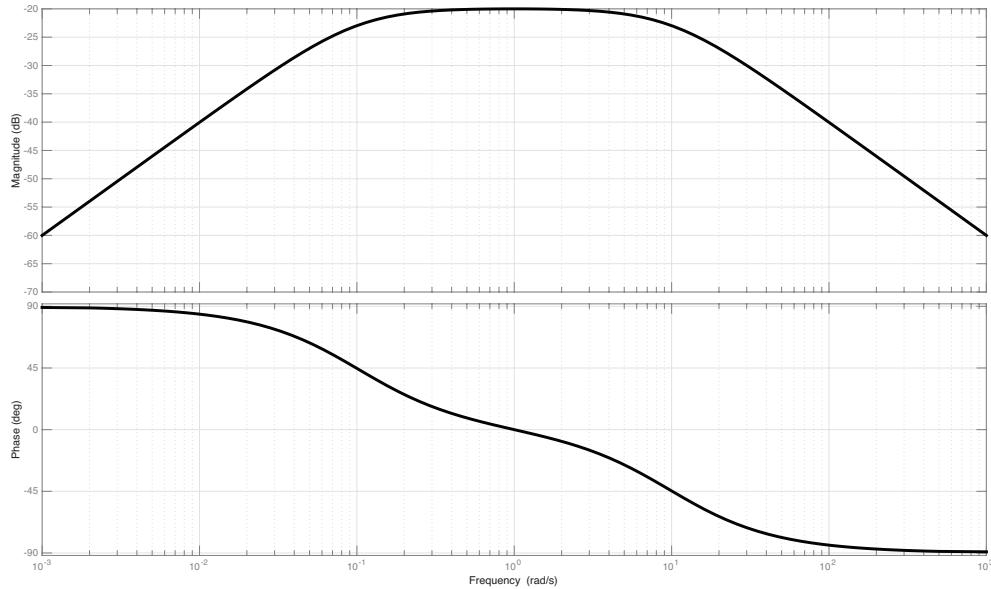


Figure 3.11: Bode plot of RLC circuit with  $R = 10$ ,  $L = 1$ ,  $C = 1$ .

from Example 2.3.5. Later we'll discuss how to sketch them by hand. The governing equation for the circuit is

$$-\dot{u} + R\dot{y} + L\ddot{y} + \frac{1}{C}y = 0.$$

The TF of this system is (verify)

$$\frac{Y(s)}{U(s)} = G(s) = \frac{s}{Ls^2 + Rs + \frac{1}{C}}$$

and the frequency response is

$$G(j\omega) = \frac{j\omega}{-jL\omega^2 + jR\omega + \frac{1}{C}}.$$

Figure 3.11 shows the Bode plot for this system and was generated using the following script in MATLAB

```

1 s = tf('s');
2 R = 10; L = 1; C = 1;
3 G = s/(L*s^2 + R*s + 1/C);
4 bode(G);
5 grid on;

```



**Example 3.8.4. (Low Pass Filter)** Recall the TF of the so-called low pass RC filter from Example 2.8.2

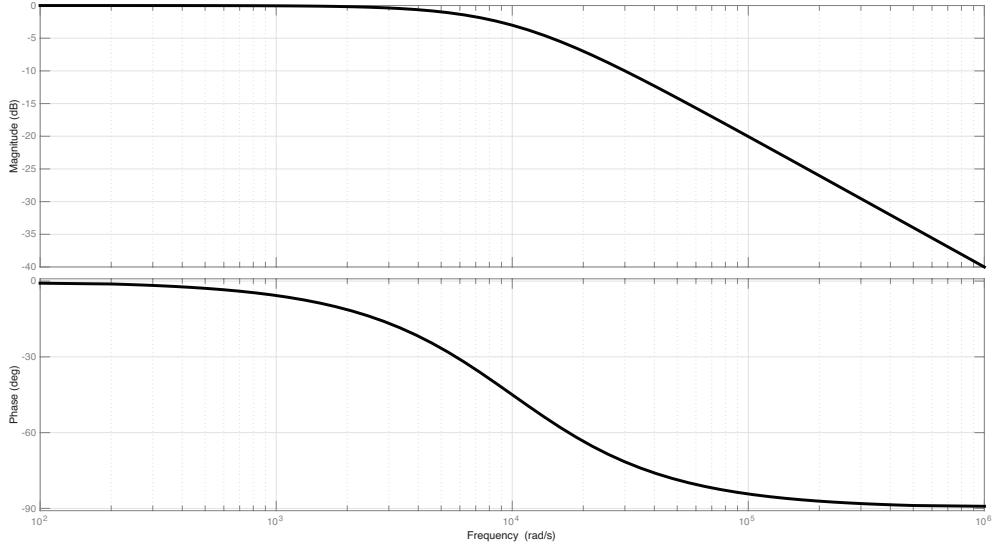


Figure 3.12: Bode plot of RC circuit with  $RC = 10^{-4}$ .

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

The associated frequency response is

$$G(j\omega) = \frac{1}{jRC\omega + 1}.$$

Figure 3.12 shows the Bode plot for this system with  $RC = 10^{-4}$ . The Bode plot provides justification for the name low-pass filter. From the plot of the magnitude response we see that when we apply sinusoidal inputs with frequencies above  $10^4$  rad/s, the amplitude of the steady-state output signal is heavily attenuated. For instance, if the input has amplitude one and frequency  $\omega = 10^5$ , then  $20 \log |G(j\omega)| \approx -20$ dB and therefore the amplitude of the output is  $|G(j\omega)| \approx 10^{-1}$ . ▲

The above example helps to motivate the notion of bandwidth for control systems. The term bandwidth is used by engineers in many different contexts like (i) bandwidth of filters (ii) bandwidth of communication channels (iii) bandwidth of control systems. In feedback control, where the plant is often a low pass filter, the higher the bandwidth the better because, as we will see in Chapter 4, the higher the bandwidth of a control system, the faster its response.

**Definition 3.8.3.** Let  $G(s)$  be a rational, proper transfer function with all its poles in  $\mathbb{C}^-$ . Let  $G_{\max} := \sup_{\omega \in [0, \infty)} |G(j\omega)|$  denote the maximum magnitude of  $G(j\omega)$ . The **bandwidth** of  $G$  is the width of the frequency range in  $[0, \infty)$  in which, for every  $\omega$  in this range

$$|G(j\omega)| \geq \frac{1}{\sqrt{2}} G_{\max}.$$

In terms of decibels, for every  $\omega$  in this frequency range,  $20 \log |G(j\omega)| - 20 \log G_{\max} \geq -3\text{dB}$ .

**Example 3.8.5.** The Bode plot in Figure 3.11 of Example 3.8.3 shows that the RLC circuit has a bandwidth of 9.9 radians/second. Input signals with frequencies outside of the interval  $[0.1, 10]$  are heavily attenuated. ▲

**Example 3.8.6.** The Bode plot in Figure 3.12 of Example 3.8.4 shows that the RC circuit has a bandwidth of  $10^4$  radians/second. Input signals with frequencies outside of the interval  $[0, 10^4]$  are heavily attenuated. ▲

When, as in Example 3.8.6, the maximum magnitude of  $G(j\omega)$  occurs at  $\omega = 0$ , we will often say that a system's bandwidth is simply  $\omega_{\text{BW}}$  with the understanding that this refers to the width of the interval  $[0, \omega_{\text{BW}}]$ . For such systems, the bandwidth  $\omega_{\text{BW}}$  indicates that the system will essentially pass all signals in the band  $[0, \omega_{\text{BW}}]$  without much attenuation. The bandwidth is easily obtained from the amplitude response plot of  $G(s)$  as illustrated in Figure 3.13.

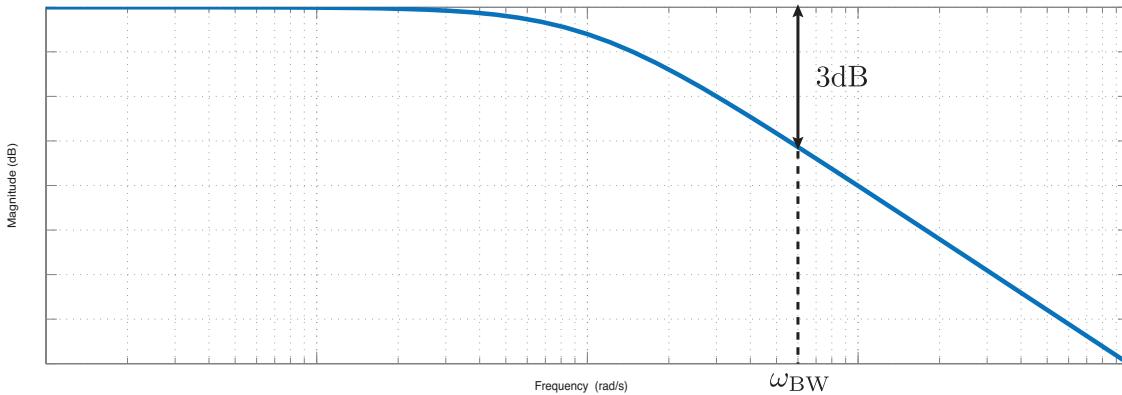


Figure 3.13: Obtaining the bandwidth of a system from its amplitude response.

**Remark 3.8.4.** The significance of the ratio  $1/\sqrt{2}$  (-3dB) in Definition 3.8.3 has its origins in the notion of signal power. The (instantaneous) power in a signal  $u(t)$  is typically proportional to the amplitude of  $u^2(t)$ . If we apply an input signal of frequency  $\omega_{\text{BW}}$  to a system with  $G(0) = 1$ , the output signal will have power proportional to  $(u(t)/\sqrt{2})^2 = u^2/2$ . In other words, the output will have *half* the power of the input signal. ♦

### 3.8.3 Drawing Bode plots by hand

We consider only rational  $G(s)$  with real coefficients, i.e.,  $G(s) \in \mathbb{R}(s)$ . Then  $G(s)$  has a numerator and denominator, each of which can be factored into terms of the following forms (cf. Section 3.8.1):

1. Pure gain:  $K$
2. Pole or zero at  $s = 0$ :  $s^n$ .
3. Real nonzero pole or zero:  $\tau s \pm 1$ ,  $\tau > 0$ .
4. Complex conjugate pole or zero:  $\frac{s^2}{\omega_n^2} + 2\frac{\zeta}{\omega_n}s + 1$ ,  $\zeta \in [0, 1)$ ,  $\omega_n \neq 0$ .

**Example 3.8.7.**

$$G(s) = \frac{40s^2(s-2)}{(s+5)(s^2+4s+100)} = \frac{(40)(2)}{(5)(100)} \frac{s^2(\frac{1}{2}s-1)}{(\frac{1}{5}s+1)(\frac{1}{100}s^2+\frac{1}{25}s+1)}.$$

Once  $G(s)$  has been factored as above, we find the frequency response by setting  $s = j\omega$ . Using the basic properties of logarithms and complex numbers it is easy to show that  $20 \log |G(j\omega)|$  and  $\angle(G(j\omega))$  consist of sums and differences of the four basic terms. This is best illustrated by example.

**Example 3.8.8.** Continuing from Example 3.8.7 consider

$$20 \log |G(s)| = 20 \log \left| \frac{80}{500} \right| + 20 \log |s^2| + 20 \log \left| \frac{1}{2}s - 1 \right| - 20 \log \left| \frac{1}{5}s + 1 \right| - 20 \log \left| \frac{1}{100}s^2 + \frac{1}{25}s + 1 \right|$$

and

$$\angle(G(s)) = \angle\left(\frac{80}{500}\right) + \angle(s^2) + \angle\left(\frac{1}{2}s - 1\right) - \angle\left(\frac{1}{5}s + 1\right) - \angle\left(\frac{1}{100}s^2 + \frac{1}{25}s + 1\right).$$



Example 3.8.8 suggests that if we are able to plot the Bode plot for the four basic factors then we can add and subtract these plots to obtain the Bode plot of  $G(s)$ . This is the main idea of this section.

### Pure gain

Consider  $G(s) = K$ . Then  $G(j\omega) = K$  and the frequency response is constant. The Bode plots are shown in Figure 3.14. It is instructive to compare the Bode plots to the polar plots in Figure 3.5.

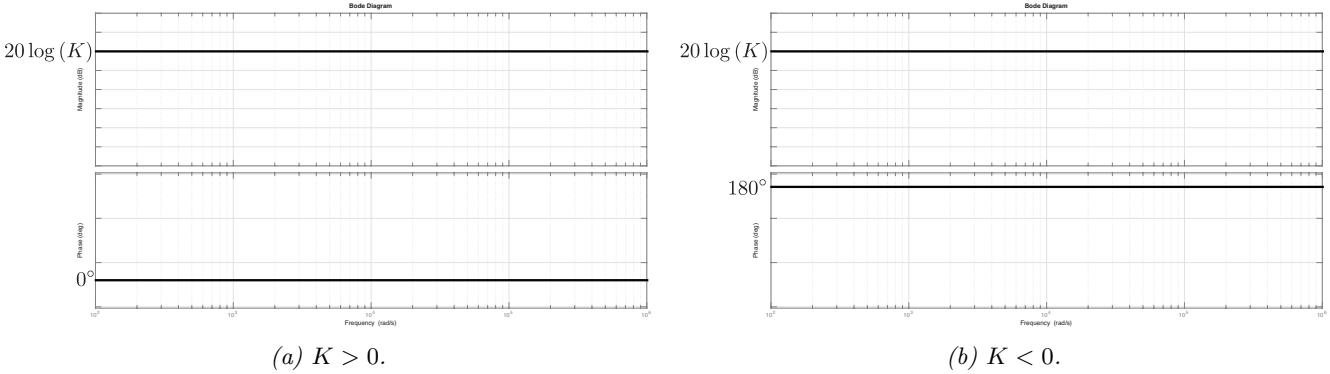


Figure 3.14: Bode plots of  $G(s) = K$ .

### Zero at the origin

Consider  $G(s) = s^n$ . Then  $G(j\omega) = (j\omega)^n$  and

$$20 \log |G(j\omega)| = 20 \log |(j\omega)^n| = 20 \log |\omega|^n = n20 \log |\omega|$$

and

$$\angle(G(j\omega)) = \angle(j\omega)^n = \begin{cases} 0^\circ & n \in \{0, 4, 8, \dots\} \\ 90^\circ & n \in \{1, 5, 9, \dots\} \\ 180^\circ & n \in \{2, 6, 10, \dots\} \\ -90^\circ & n \in \{3, 7, 11, \dots\}. \end{cases}$$

It is useful to compare these expressions, especially for the phase response, to the polar plots in Figure 3.6. Notice that when  $\omega = 1 = 10^0$  then  $20 \log |G(j\omega)| = 0$ ; when  $\omega = 10 = 10^1$  then  $20 \log |G(j\omega)| = n20$ ; when  $\omega = 100 = 10^2$  then  $20 \log |G(j\omega)| = n40$ ; etc. In other words every time the frequency increases by a factor of 10 (called a **decade** increase), the magnitude response increases by  $n20$  dB. Thus on a  $20 \log |G(j\omega)|$  vs  $\log(\omega)$  plot the magnitude response is a straight line with slope  $n20$  dB which passes through 0dB at  $\omega = 10^0$ . The Bode plots for  $n \in \{1, 2, 3, 4\}$  are shown in Figure 3.15.

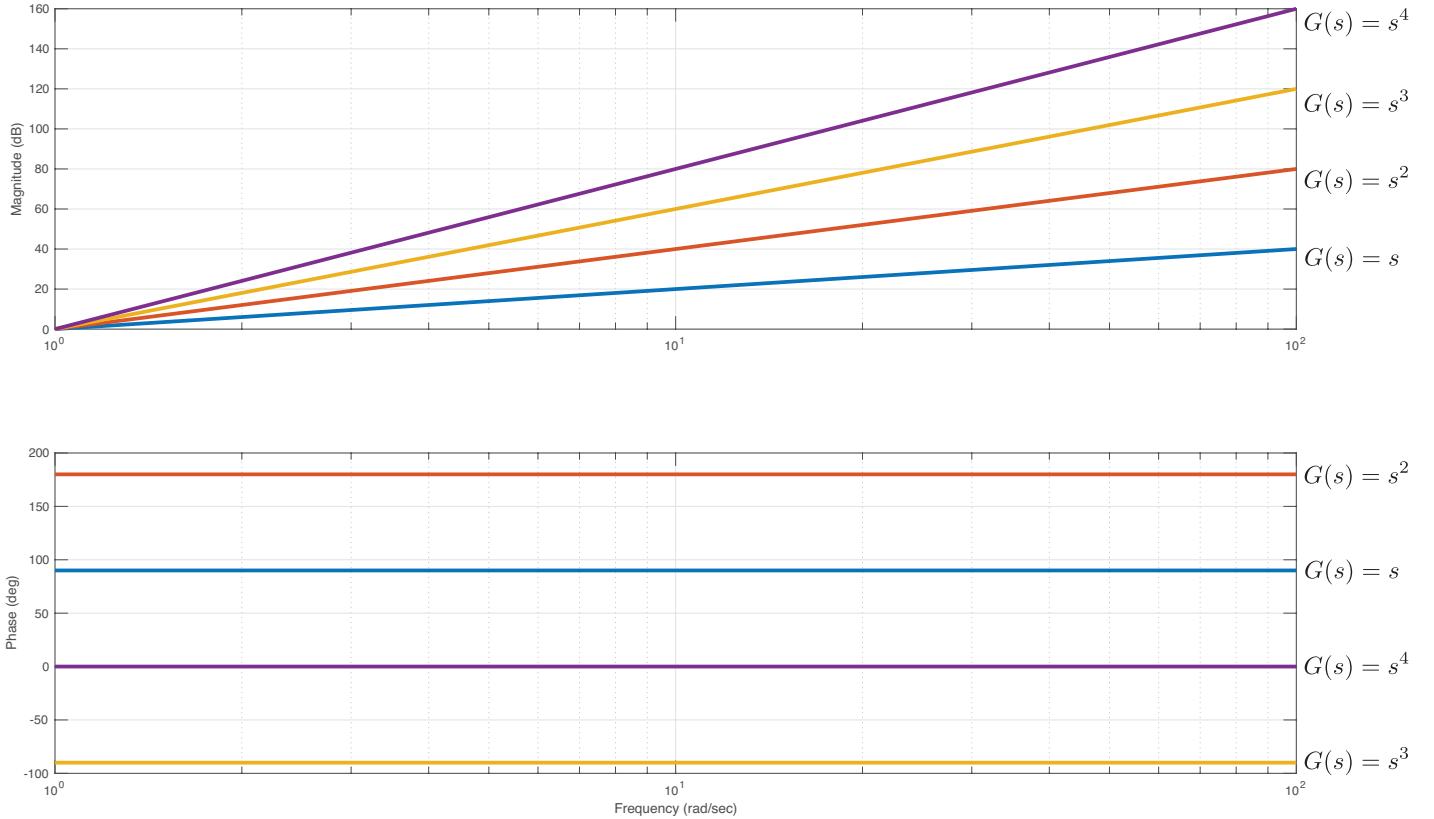


Figure 3.15: Bode plot of  $G(s) = s^n$  for  $n \in \{1, 2, 3, 4\}$ .

### Real zero

Consider  $G(s) = \tau s + 1$  with  $\tau > 0$ . Then  $G(j\omega) = j\tau\omega + 1$  and

$$20 \log |G(j\omega)| = 20 \log |j\tau\omega + 1|, \quad \angle(G(j\omega)) = \angle(j\tau\omega + 1) = \arctan2(1, \tau\omega).$$

We now make some key observations used to draw Bode plots by hand. The polar plot in Figure 3.7b is helpful for understanding these observations.

1. When  $\omega \ll \frac{1}{\tau}$  then:
  - (a)  $20 \log |j\tau\omega + 1| \approx 20 \log |1| = 0$ .
  - (b)  $\angle(j\tau\omega + 1) = \arctan2(1, \tau\omega) \approx \arctan2(1, 0) = 0^\circ$ .
2. When  $\omega \gg \frac{1}{\tau}$  then:
  - (a)  $20 \log |j\tau\omega + 1| \approx 20 \log |\omega|$ .
  - (b)  $\angle(j\tau\omega + 1) = \arctan2(1, \tau\omega) \approx \arctan2(0, \omega) = 90^\circ$ .
3. When  $\omega = \frac{1}{\tau}$  then:
  - (a)  $20 \log |j\tau\omega + 1| = 20 \log |\sqrt{2}| = 3\text{dB}$ .
  - (b)  $\angle(j\tau\omega + 1) = \arctan2(1, 1) = 45^\circ$ .

The frequency  $\frac{1}{\tau}$  is called the **break frequency** of  $\tau s + 1$ . Based on the above observations, we make the following simplifications for drawing the Bode plot of  $\tau s + 1$  by hand:

1. For  $\omega < \frac{1}{\tau}$ ,  $|j\tau\omega + 1| \approx 1$ .

2. For  $\omega > \frac{1}{\tau}$ ,  $|j\tau\omega + 1| \approx |\omega|$ .
3. For  $\omega < \frac{0.1}{\tau}$ ,  $\angle j\tau\omega + 1 \approx \angle 1 = 0^\circ$ .
4. For  $\omega > \frac{10}{\tau}$ ,  $\angle(j\tau\omega + 1) \approx \angle j\omega = 90^\circ$ .

The lines obtained using the above approximations are called **asymptotes** and the resulting approximation of the Bode plot is called an **asymptotic Bode plot**. Figure 3.16 shows the asymptotic Bode plot when  $\tau = 1$  as well as the exact Bode plot. The maximum error between the magnitude plots occurs at  $\omega = \frac{1}{\tau}$ .

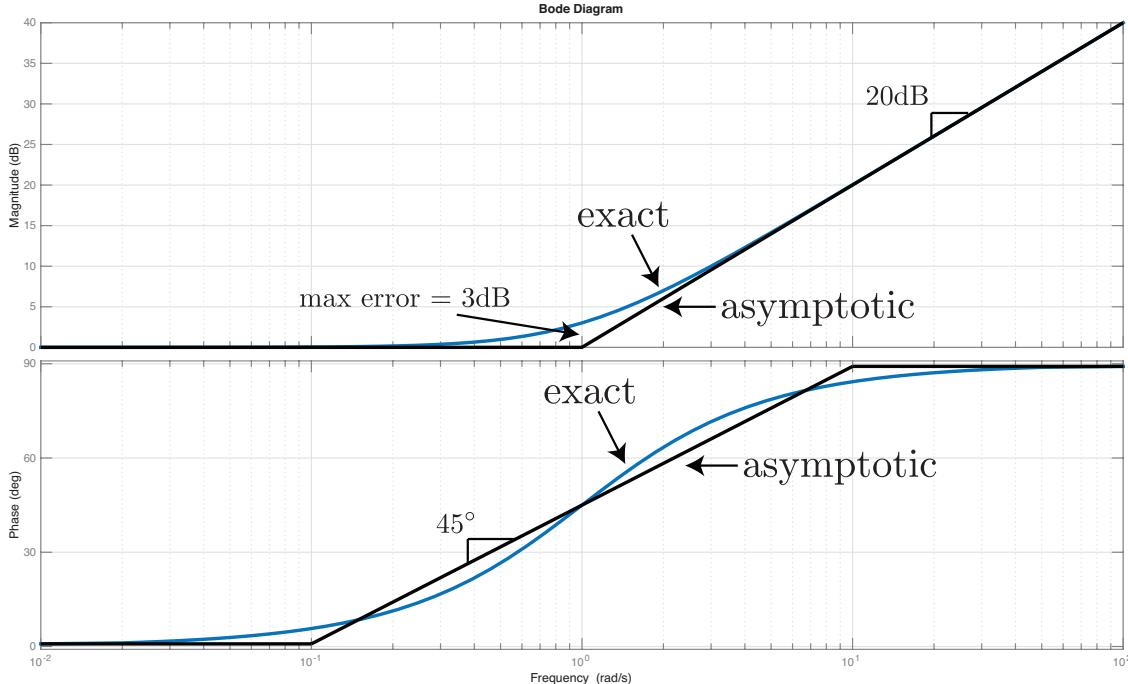


Figure 3.16: Exact and asymptotic Bode plots of  $\tau s + 1$  with  $\tau = 1$ .

Now consider  $G(s) = \tau s - 1$  with  $\tau > 0$ . Then  $G(j\omega) = j\tau\omega - 1$ . The magnitude plot is exactly as before. The only difference is in the phase plot. Once again it is instructive to observe the polar plot in Figure 3.7a. In this case we have that when  $\omega \ll \frac{1}{\tau}$ ,  $\angle G(j\omega) \approx 180^\circ$  and when  $\omega \gg \frac{1}{\tau}$ ,  $\angle G(j\omega) \approx 90^\circ$ . The asymptotic Bode plot is drawn in the same manner. The plots are shown in Figure 3.17.

### Complex conjugate zeros

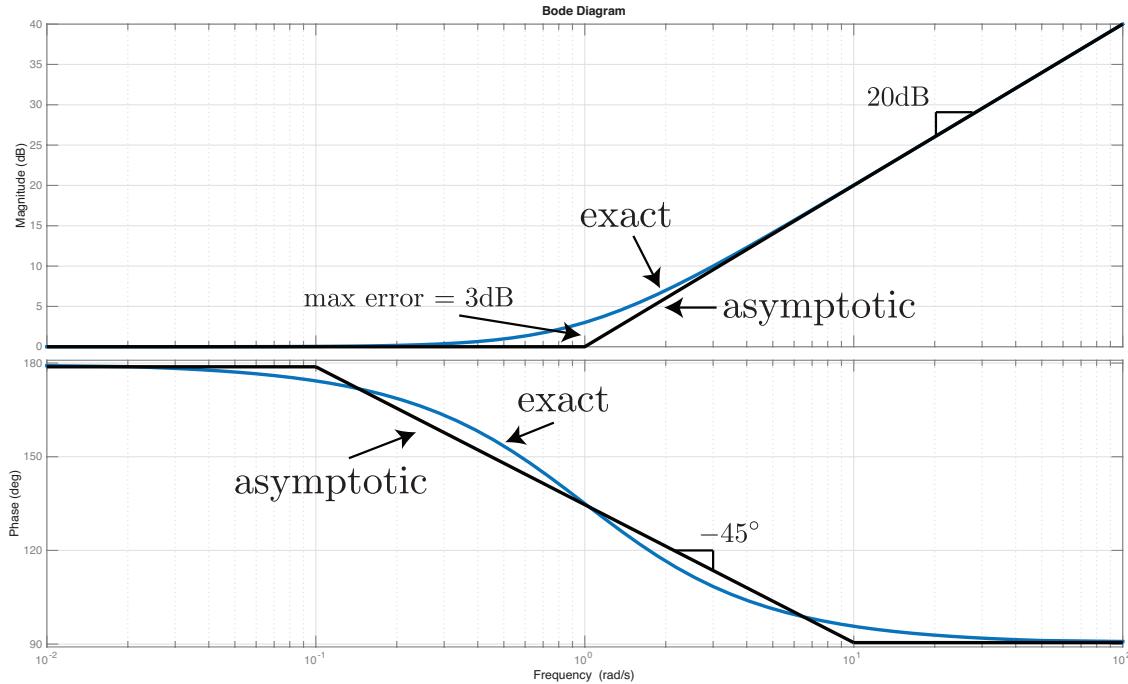
Lastly we consider the case when  $G(s)$  has complex conjugate zeros

$$G(s) = \frac{s^2}{\omega_n^2} + 2\frac{\zeta}{\omega_n}s + 1, \quad \zeta \in [0, 1), \quad \omega_n \neq 0.$$

For asymptotic Bode plots we make the approximation that  $\zeta = 1$ , i.e.,

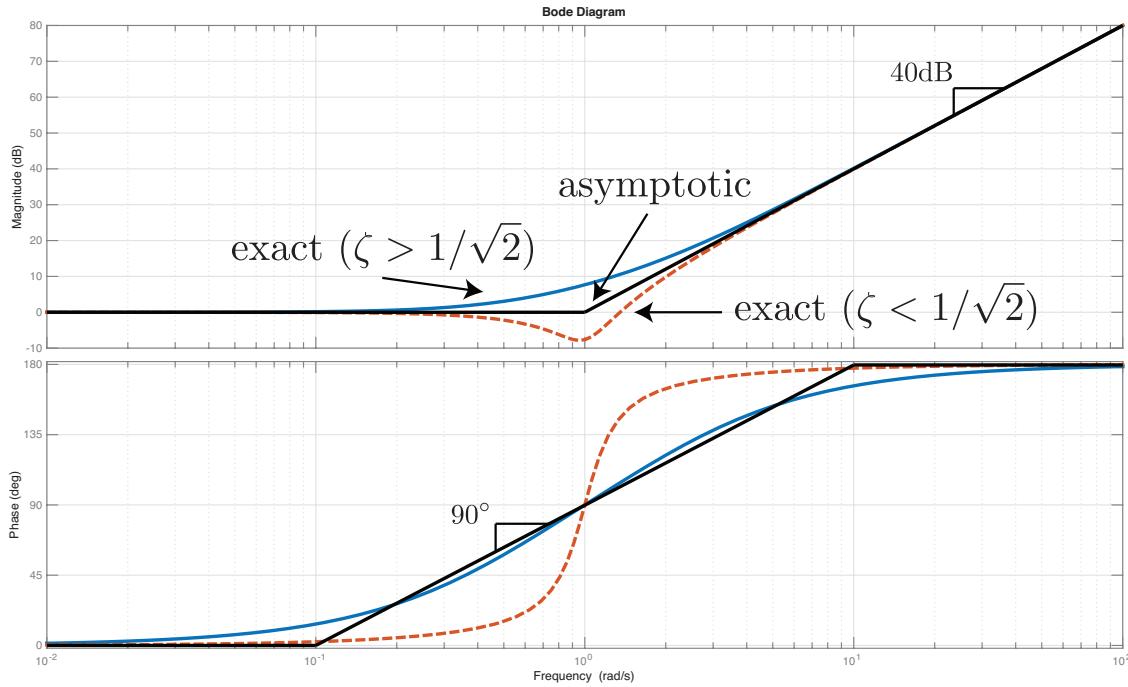
$$G(s) \approx \left( \frac{s^2}{\omega_n^2} + 2\frac{1}{\omega_n}s + 1 \right) = \left( \frac{s}{\omega_n} + 1 \right)^2$$

In this approximation we treat the complex conjugate zeros as two real, repeated zeros at  $s = -\omega_n$ . The advantage of doing this is that we can generate asymptotic Bode plots in the same fashion as the previous section using a break frequency of  $|\omega_n|$ . The disadvantage is that the approximation is quite crude and can lead to large errors between the asymptotic and exact Bode plots. The error is entirely dependent on the actual value

Figure 3.17: Exact and asymptotic Bode plots of  $\tau s - 1$  with  $\tau = 1$ .

of  $\zeta$ . The closer  $\zeta$  is to 1, the more accurate the plot. To make the asymptotic Bode plot more accurate you can correct it by computing the true value of  $|G(j\omega)|$  and  $\angle G(j\omega)$  near  $\omega = \omega_n$ . Most introductory textbooks have tables of correction factors for different values of  $\zeta$ .

Figure 3.18 shows a plot for complex conjugate zeros in  $\mathbb{C}^-$ , i.e.,  $\omega_n > 0$ ,  $\zeta \in (0, 1)$ . Figure 3.19 shows the plot for complex conjugate zeros in  $\mathbb{C}^+$ , i.e.,  $\omega_n < 0$ ,  $\zeta \in (0, 1)$ . Once again it is instructive to compare these Bode plots to the polar plots in Figure 3.7.

Figure 3.18: Exact and asymptotic Bode plots of  $\frac{s^2}{\omega_n^2} + 2\frac{\zeta}{\omega_n}s + 1$  with  $\omega_n = 1$ .

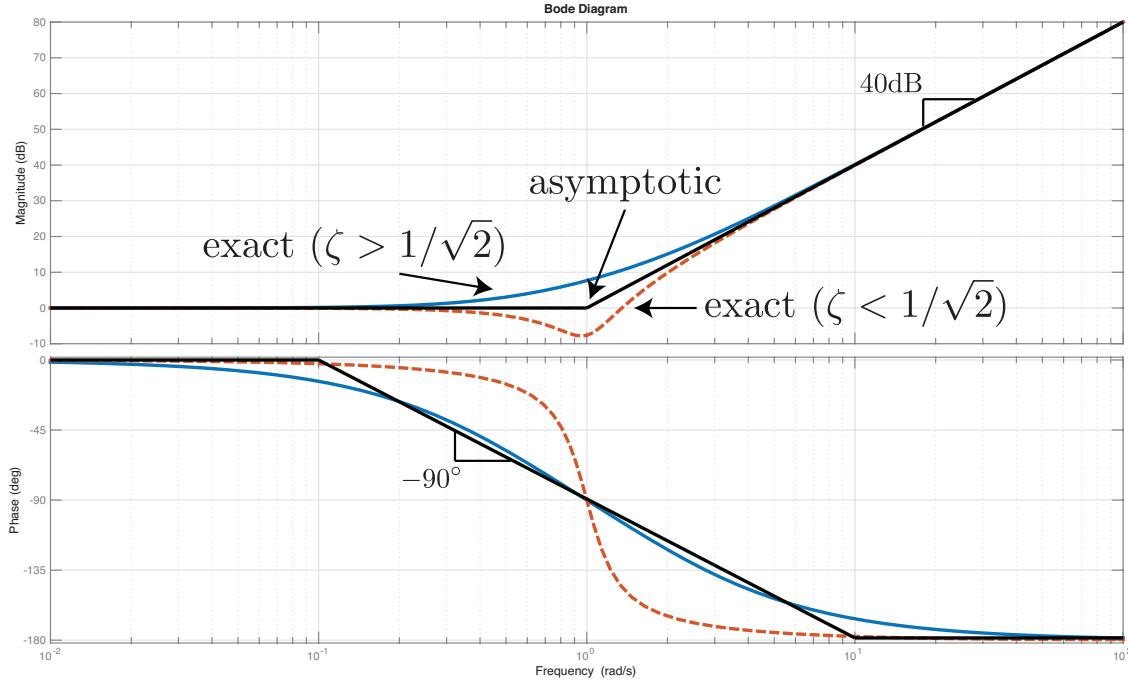


Figure 3.19: Exact and asymptotic Bode plots of  $\frac{s^2}{\omega_n^2} + 2\frac{\zeta}{\omega_n}s + 1$  with  $\omega_n = -1$ .

A special case is when  $\zeta = 0$ . In this case  $G(s)$  has imaginary roots at  $s = \pm j\omega_n$ . When  $\zeta = 0$  and  $\omega = \omega_n$  we have  $G(j\omega) = 0$ . Hence the magnitude bode plot approaches negative infinity as  $\omega \rightarrow \omega_n$ . This is illustrated in Figure 3.20. Compare with the polar plot in Figure 3.8. In such a case the asymptotic Bode plot has infinite error at the break frequency  $\omega = \omega_n$ .

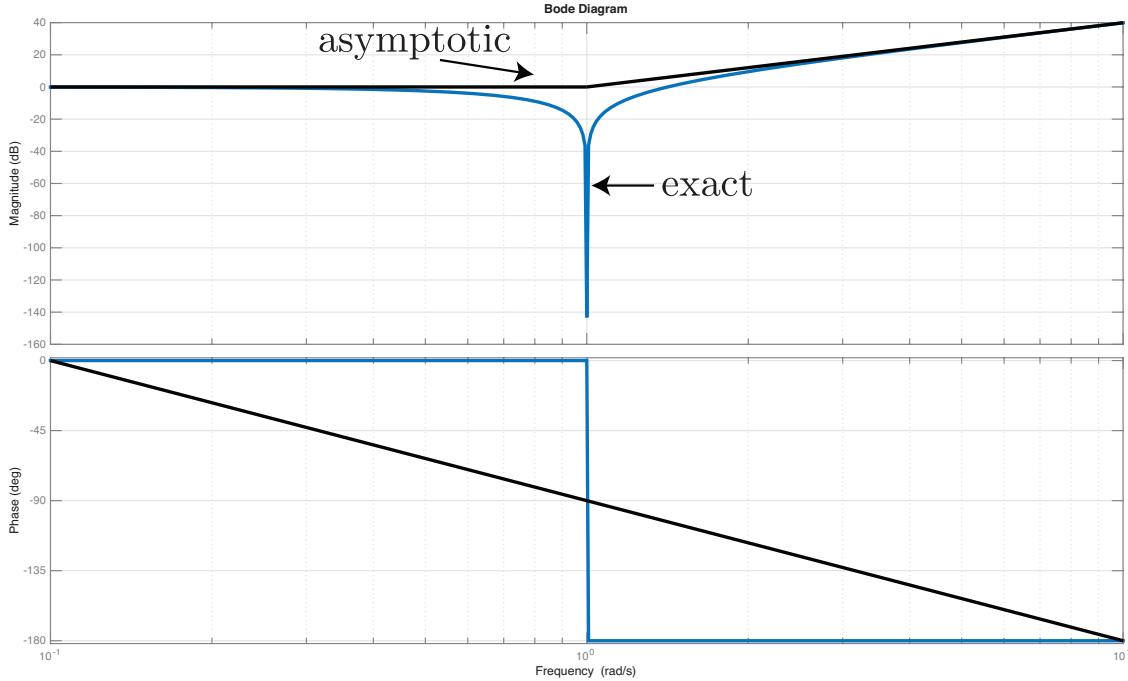


Figure 3.20: Exact and asymptotic Bode plots of  $\frac{s^2}{\omega_n^2} + 1$  with  $\omega_n^2 = 1$ .

**Exercise 3.6.** Using the principles developed so far, sketch the asymptotic Bode plot of

$$G(s) = \frac{40s^2(s-2)}{(s+5)(s^2+4s+100)}.$$

## Examples

**Example 3.8.9. (Minimum Phase)** A minimum phase TF

$$G_1(s) = \frac{1}{s+10} = 0.1 \frac{1}{0.1s+1}.$$

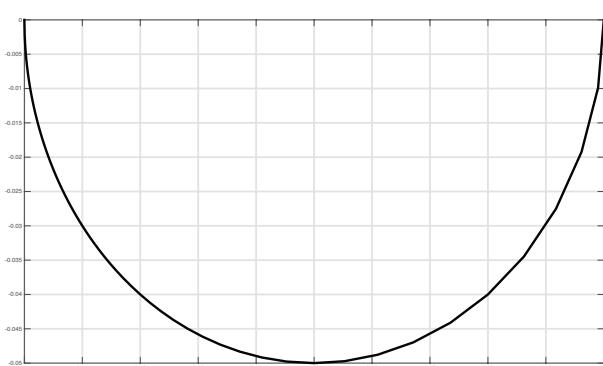
The frequency response is

$$G(j\omega) = 0.1 \frac{1}{j0.1\omega+1}.$$

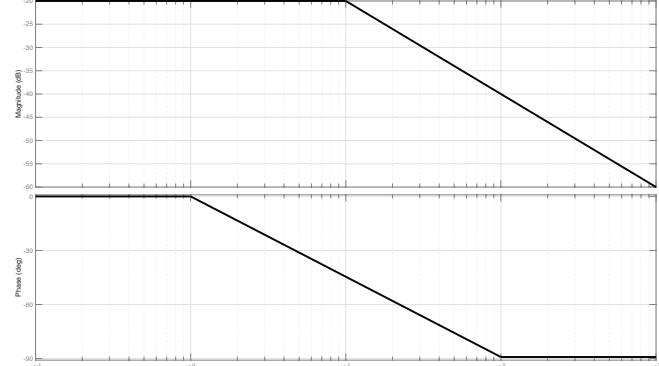
It follows that

$$\begin{aligned} 20 \log |G(j\omega)| &= 20 \log |0.1| - 20 \log |j0.1\omega + 1| = -20 - 20 \log |j0.1\omega + 1| \\ \angle(G(j\omega)) &= \angle 0.1 - \angle(j0.1\omega + 1) = -\angle(j0.1\omega + 1). \end{aligned}$$

Figure 3.21 shows the polar plot and asymptotic Bode plot for this frequency response. Make sure you understand the asymptotic Bode plot in Figure 3.21b and that you can plot it by hand. ▲



(a) Polar plot.



(b) Asymptotic Bode plot.

Figure 3.21: Polar and asymptotic Bode plots for Example 3.8.9.

**Example 3.8.10. (All-Pass)** An all-pass TF

$$G_2(s) = \frac{1-s}{1+s}.$$

The frequency response is

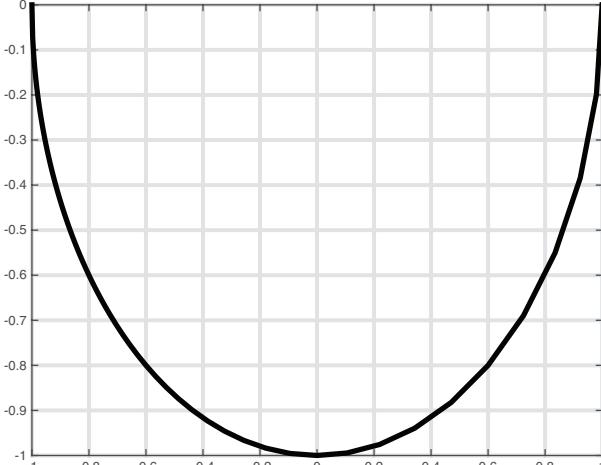
$$G_2(j\omega) = -1 \frac{j\omega - 1}{j\omega + 1}.$$

It follows that

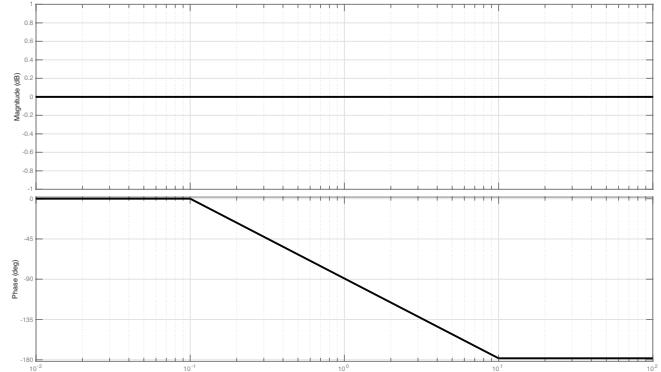
$$20 \log |G_2(j\omega)| = 20 \log |-1| + 20 \log |j\omega - 1| - 20 \log |j\omega + 1| = 0 \text{ dB}.$$

$$\angle(G_2(j\omega)) = \angle -1 + \angle(j\omega - 1) - \angle(j\omega + 1) = 180^\circ + \angle(j\omega - 1) - \angle(j\omega + 1).$$

Figure 3.22 shows the polar plot and asymptotic Bode plot for this frequency response. As before, make sure you understand the Bode plot in Figure 3.22b and know how to plot it by hand. ▲



(a) Polar plot.



(b) Asymptotic Bode plot.

Figure 3.22: Polar and asymptotic Bode plots for Example 3.8.10.

**Example 3.8.11. (Non-Minimum Phase)** A non-minimum phase TF

$$G_3(s) = G_2(s)G_1(s) = \frac{1-s}{1+s} \frac{1}{s+10}.$$

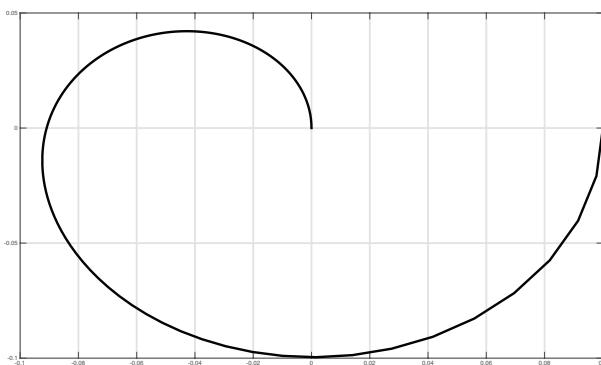
Using basic properties of complex numbers we have that

$$|G_3(j\omega)| = |G_2(j\omega)||G_1(j\omega)| = |G_1(j\omega)|$$

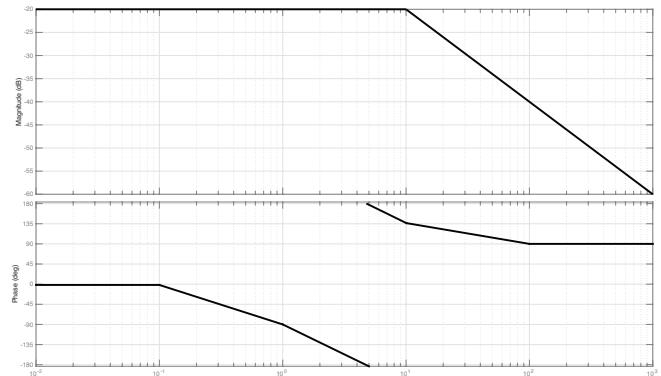
so that the magnitude response of  $G_3$  is that same as that of  $G_1$  from Figure 3.21b. Furthermore the frequency response is

$$\angle G_3(j\omega) = \angle G_2(j\omega) + \angle G_1(j\omega).$$

Figure 3.23 shows the polar plot and asymptotic Bode plot for this frequency response. As before, make sure you understand the Bode plot in Figure 3.23b and know how to plot it by hand. From the polar plot we again



(a) Polar plot.



(b) Asymptotic Bode plot.

Figure 3.23: Polar and asymptotic Bode plots for Example 3.8.11.

see (cf. Example 3.8.2) that the phase change for  $G_3$  is greater than the phase change for  $G_1$ . Of all TFs having the same magnitude response  $|G_1(j\omega)|$ ,  $G_1$  has the minimum phase change. Hence the terminology of Definition 3.8.2. ▲

**Remark 3.8.5.** This section has dealt with graphical methods done by hand on paper. Why did we bother? Why didn't we just ask MATLAB to draw the Bode plots? The answer is that by learning to draw the graphs,

you acquire understanding. In addition, there will come times when MATLAB will fail you. For example, poles on the imaginary axis: MATLAB isn't smart enough to properly draw Nyquist plots which we cover in Chapter 8. ♦

## 3.9 Summary

This chapter covered some of the basic results in linear system theory. Parts of this chapter should have been review from your signals and systems course. After reading this chapter you should know the following.

1. You should know how to compute the matrix exponential  $e^{At}$  using Laplace transforms for small problems ( $n \leq 3$ ).
2. Understand the structure of the total response of an LTI state-space model.
3. Know what it means for an LTI system to be asymptotically stable and how to test for asymptotic stability using eigenvalues.
4. Know the definition of BIBO stability and how to test for it by checking the poles of a TF.
5. Understand the relationship between asymptotic stability and BIBO stability of LTI systems.
6. Be able to apply the final-value theorem and compute the steady-state gain of a BIBO stable system.
7. Understand the physical interpretation of a system's frequency response.
8. Be able to plot asymptotic Bode plots by hand.

## 3.A Appendix: Complex numbers

We define  $\mathbb{C}$ , the **complex numbers**, to be a set of ordered pairs  $(x, y)$  where  $x$  and  $y$  are real and where addition and multiplication are defined by:

$$(x_1, y_1) + (x_2, y_2) := (x_1 + x_2, y_1 + y_2)$$

$$(x_1, y_1)(x_2, y_2) := (x_1 x_2 - y_1 y_2, x_1 y_2 + x_2 y_1).$$

We adopt the standard way of writing  $z = (x, y) \in \mathbb{C}$  as  $z = x + jy$  where  $j \cdot j = j^2 = -1$  and  $x, y \in \mathbb{R}$ . This is called the **Cartesian form** of  $z$ . With this convention, addition and multiplication of complex numbers can be done using the ordinary rules of algebra

$$(x_1 + jy_1) + (x_2 + jy_2) = (x_1 + x_2) + j(y_1 + y_2)$$

$$(x_1 + jy_1)(x_2 + jy_2) = (x_1 x_2 - y_1 y_2) + j(x_1 y_2 + x_2 y_1).$$

For the complex number  $z = x + jy$  we write  $\text{Re}(z) := x$  and  $\text{Im}(z) := y$  for the real part and imaginary part, respectively, of  $z$ . Note that *both* the real part *and* the imaginary part of a complex numbers are *real numbers*. The complex number  $-2 + j3$  has real part  $-2$  and imaginary part  $3$ . Complex numbers of the form  $z = x + j0$  are called **purely real** while complex numbers of the form  $z = 0 + jy$  are called **purely imaginary**. The **complex conjugate** of  $z = x + jy$  is defined to be the complex number  $\bar{z} = x - jy$ . It is easy to prove that

$$\text{Re}(z) = \frac{1}{2}(z + \bar{z}), \quad \text{Im}(z) = \frac{1}{2j}(z - \bar{z}).$$

By  $|z|$  we denote the **magnitude** or **modulus** of  $z$  and it equals  $\sqrt{x^2 + y^2}$ . You can verify that

$$|z| = |\bar{z}|, \quad |z|^2 = z\bar{z}, \quad \frac{1}{z} = \frac{\bar{z}}{|z|^2}.$$

For  $z, w \in \mathbb{C}$ , you can also verify the following basic properties of magnitudes

$$|zw| = |z||w|, \quad \left| \frac{z}{w} \right| = \frac{|z|}{|w|}.$$

By  $\angle z$  we denote the angle between the positive real axis and the line segment from 0 to  $z$  in the complex plane. This angle is called the **argument** of  $z$  and we write  $\angle z = \arg(z)$ . Since we can always add integer multiples of  $2\pi$ ,  $\arg(z)$  is not single-valued and therefore “arg” is not a function. To avoid confusion, we always use the principle value of the argument.

**Definition 3.A.1.** The **principle value of the argument** (or **principal argument**) of a complex number  $z$  is the value of  $\arg(z)$  greater than  $-\pi$  and less than or equal to  $\pi$ .

The principle argument of  $z = z + jy$  can be computed as  $\arctan2(x, y)$  where  $\arctan2 : \mathbb{R}^2 \setminus \{0\} \rightarrow (-\pi, \pi]$  is the four quadrant arctan. Unless otherwise stated, we always use the principle value of the argument in this course so that  $-\pi < \angle z \leq \pi$ .

**Example 3.A.1.** The argument of  $-1 - j$  is any number  $-3\pi/4 + 2k\pi$  where  $k$  is an arbitrary integer. The principle argument of  $-1 - j$  is  $-3\pi/4$ . ▲

Euler's equation states that for all  $x \in \mathbb{R}$ ,  $e^{jx} = \cos(x) + j \sin(x)$ . Notice that  $|e^{jx}| = 1$  and that we can always represent  $x \in \mathbb{R}$  as  $x = \theta + 2\pi k$  where  $\theta \in (-\pi, \pi]$  and  $k$  is an integer. Using this representation, the principle argument of  $e^{jx}$  equals  $\theta$  and we can write any non-zero complex number  $z = x + jy$  in **polar form**

$$z = |z|e^{j\angle z}.$$

It is common in engineering to drop the dependence on  $e$  and simply write  $|z|\angle z$ . This is called **phasor notation**. It is easy to show that

$$(r_1 e^{j\theta_1}) (r_2 e^{j\theta_2}) = r_1 r_2 e^{j(\theta_1 + \theta_2)}.$$

It follows that for any  $z, w \in \mathbb{C}$

$$\angle zw = \angle z + \angle w, \quad \angle \frac{z}{w} = \angle z - \angle w.$$

## 3.B Appendix: Time domain, s-domain and frequency domain

In this course we work in three domains: (1) the time-domain, (2) the *s*-plane and (3) the frequency domain. In each domain there is one mathematical object that captures the essence of system behaviour: (1) the impulse response  $g(t)$  (2) the transfer function  $G(s)$ , and (3) the frequency response  $G(j\omega)$ . Anything that we say about one of these three objects should be reflected in the others. We will move with ease between these three domains for the remainder of the course and you should understand the relationship between them. This relationship is illustrated, in the case of a BIBO stable system, in Figure 3.24.

We now review the justification for the various arrows in Figure 3.24. In Section 2.7 we discussed the conditions under which the LT  $\mathcal{L}$  of  $g(t)$  exists. Since  $g$  is assumed to be causal we can uniquely recover  $g(t)$  from  $G(s)$  without being told the ROC for  $G(s)$ . This justifies the arrows between  $g(t)$  and  $G(s)$ .

In Section 2.7.1 we discussed the conditions under which the FT  $\mathcal{F}$  of  $g(t)$  exists. Since the system in Figure 3.24 is assumed BIBO stable, it follows from Condition 2 of Theorem 3.5.4 that  $\mathcal{F}(g)$  exists (see also Section 3.7). This justifies the arrows between  $g(t)$  and  $G(j\omega)$ .

Since  $G(s)$  is BIBO stable, Condition 3 of Theorem 3.5.4 tells us that the ROC of  $G(s)$  includes the imaginary axis and the arrow from  $G(s)$  to  $G(j\omega)$  is justified (see Sections 2.7.2 and 3.7). We are left to justify the arrow from  $G(j\omega)$  to  $G(s)$ . Note that it is not obvious that one should be able to recover  $G(s)$  from  $G(j\omega)$ . After all, the frequency response function only gives us data on the imaginary axis.

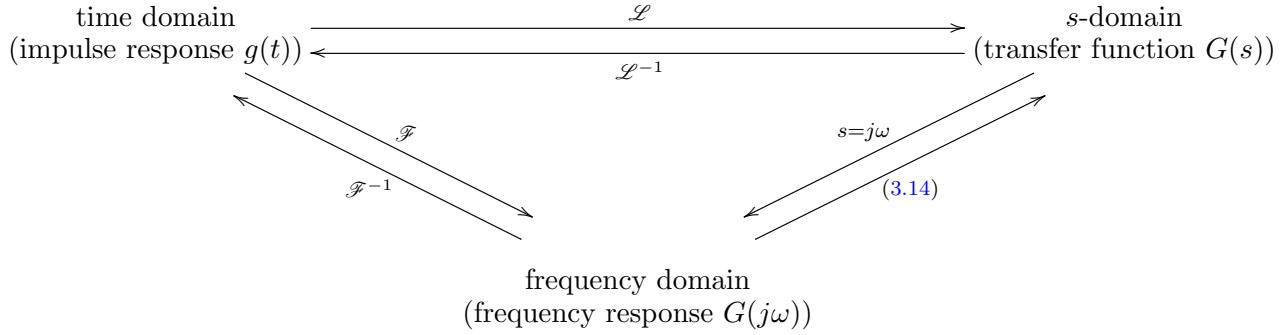


Figure 3.24: Relationship between impulse response, s-domain and frequency response of a BIBO stable LTI system.

**Proposition 3.B.1.** Let  $G(s) \in \mathbb{R}(s)$  be a strictly proper BIBO stable system with impulse response  $g(t)$  and frequency response  $G(j\omega)$ . Then, for all  $s \in \mathbb{C}^+$

$$G(s) = \frac{1}{2\pi} \int_{-\infty}^{\infty} \frac{G(j\omega)}{s - j\omega} d\omega. \quad (3.14)$$

*Proof.* As discussed above and in Section 3.7, under the assumption of BIBO stability, the frequency response of a system equals the Fourier transform of  $g(t)$  and therefore

$$g(t) = \mathcal{F}^{-1}\{G(j\omega)\} = \frac{1}{2\pi} \int_{-\infty}^{\infty} G(j\omega) e^{j\omega t} d\omega.$$

On the other hand,  $G(s)$  is the LT of  $g(t)$  and so for any  $s$  in its ROC, which by BIBO stability includes  $\mathbb{C}^+$ , we have

$$\begin{aligned} G(s) &= \int_0^\infty g(t) e^{-st} dt \\ &= \frac{1}{2\pi} \int_0^\infty \int_{-\infty}^\infty G(j\omega) e^{j\omega t} e^{-st} d\omega dt \\ &= \frac{1}{2\pi} \int_{-\infty}^\infty \int_0^\infty G(j\omega) e^{j\omega t} e^{-st} dt d\omega \\ &= \frac{1}{2\pi} \int_{-\infty}^\infty \frac{G(j\omega)}{s - j\omega} d\omega. \end{aligned}$$

□

**Remark 3.B.2.** State models fit into the above discussion via the impulse response (3.4) given by  $g(t) = Ce^{At}B\mathbf{1}(t) + D\delta(t)$  or via the transfer function  $G(s) = C(sI - A)^{-1}B + D$ . ♦

**Remark 3.B.3.** This discussion can be extended to unstable systems and systems with poles on the imaginary axis. The treatment in [Dullerud and Paganini, 2000] is somewhat advanced but rather complete. ♦

# Chapter 4

---

## First and second order systems

In the previous chapter we studied properties of linear time-invariant systems in a somewhat general manner. We showed that a single-input, single-output LTI system can be modelled by state equations

$$\dot{x} = Ax + Bu, \quad y = Cx + Du, \quad u, y \in \mathbb{R}$$

or by input-output models

$$y = g * u \quad \text{or} \quad Y(s) = G(s)U(s).$$

In this chapter we study the important special cases when  $G(s) \in \mathbb{R}(s)$  is given by

$$G(s) = \frac{K}{\tau s + 1} \quad (\text{prototype first order system}),$$
$$G(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \quad (\text{prototype second order system}).$$

First and second order systems are important because they are easy to analyze which makes them useful for developing intuition that can be applied to more complicated systems, e.g., higher order systems, systems with zeros. Also, the behaviour of these more complicated systems can sometimes be well approximated by first and second order models. Our main objective in this chapter is to relate the position of the poles of  $G(s)$  in the  $s$ -plane to the corresponding time-domain responses. This relationship is a key to the analysis and design of LTI feedback controllers.

## Contents

4.1 Prototype first order system . . . . .	93
4.2 Prototype second order system . . . . .	97
4.3 General characteristics of a step response . . . . .	104
4.4 The effect of adding poles and zeros . . . . .	108
4.5 Dominant poles and zeros . . . . .	111
4.6 Summary . . . . .	113

### 4.1 Prototype first order system

**Definition 4.1.1.** The **prototype first order system** with input  $u$  and output  $y$  is governed by the differential equation

$$\tau \dot{y} + y = Ku, \quad K, \tau \in \mathbb{R}. \quad (4.1)$$

The associated transfer function is

$$\frac{Y(s)}{U(s)} = \frac{K}{\tau s + 1}. \quad (4.2)$$

System (4.1) in Definition 4.1.1 can of course be given a state model. Define  $x := y$  to obtain (verify)

$$\dot{x} = ax + bu, \quad y = cx + du, \quad x, y, u \in \mathbb{R}$$

and

$$a = -\frac{1}{\tau}, \quad b = \frac{K}{\tau}, \quad c = 1, \quad d = 0.$$

For the remainder of this section we study how the values of the constants  $K$  and  $\tau$  affect the time-domain response of the prototype first order system.

**Example 4.1.1. (Heating a room)** Consider the problem of heating up a room as shown in Figure 4.1. Suppose that:

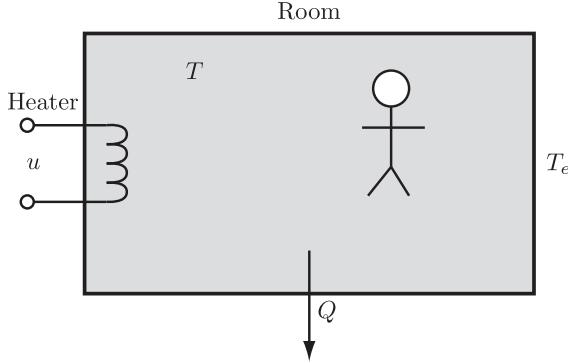


Figure 4.1: Heating a room.

- $u$  is the heat provided by the heater; the input.
- $T$  is the inside temperature. This is the variable we want to control; the output.
- $T_e$  is the exterior (outside) temperature. We assume that  $T_e$  is constant.
- $Q$  is the heat flow out of the room.
- $Q_a$  is the heat accumulating within the system.

By conservation of energy we have that

$$Q_a = u - Q.$$

If  $Q_a = \beta \frac{dT}{dt}$  and  $Q = \alpha(T - T_e)$ , where  $\beta$  and  $\alpha$  are real constants, then the system model is

$$\frac{dT}{dt} = \frac{u}{\beta} - \frac{\alpha}{\beta}(T - T_e). \quad (4.3)$$

This model is nonlinear (Do you see why? Hint: see Example 2.4.4.) with input  $u$  and output  $T$ . The governing equation (4.3) is first order so we only need one state variable. The natural choice is  $x := T$  so that the resulting nonlinear state model is

$$\begin{aligned} \dot{x} &= f(x, u) := -\frac{\alpha}{\beta}x + \frac{1}{\beta}u + \frac{\alpha T_e}{\beta} \\ y &= h(x, u) := x. \end{aligned}$$

Suppose that we want to control the temperature of the room to maintain an inside temperature of  $\bar{T} > T_e$ , i.e., the room should be  $\bar{T} - T_e$  degrees warmer than the outside.

**Exercise 4.1.** Show that  $(\bar{x}, \bar{u}) = (\bar{T}, \alpha(\bar{T} - T_e))$  is the unique equilibrium configuration corresponding to the desired temperature  $\bar{x} = \bar{T}$  (see Definition 2.5.1).

**Exercise 4.2.** Compute Jacobians  $a = \frac{\partial f}{\partial x}|_{(\bar{x}, \bar{u})}$ ,  $b = \frac{\partial f}{\partial u}|_{(\bar{x}, \bar{u})}$ ,  $c = \frac{\partial h}{\partial x}|_{(\bar{x}, \bar{u})}$ ,  $d = \frac{\partial h}{\partial u}|_{(\bar{x}, \bar{u})}$  and verify that  $a = -\alpha/\beta$ ,  $b = 1/\beta$ ,  $c = 1$ ,  $d = 0$ .

Finally, let  $\delta x := x - \bar{x} = T - \bar{T}$  and  $\delta u := u - \bar{u} = u - \alpha(\bar{T} - T_e)$  denote deviations from the equilibrium configuration to obtain the linearized model

$$\begin{aligned}\dot{\delta x} &= -\frac{\alpha}{\beta}\delta x + \frac{1}{\beta}\delta u \\ \delta y &= \delta x.\end{aligned}$$

Let  $Y(s) := \mathcal{L}\{\delta y\}$  and  $U(s) := \mathcal{L}\{\delta u\}$ . The resulting transfer function of the linearized model is

$$\frac{Y(s)}{U(s)} = \frac{\frac{1}{\beta}}{s + \frac{\alpha}{\beta}} = \frac{\frac{1}{\alpha}}{\frac{\beta}{\alpha}s + 1}.$$

This system has the form of a prototype first order system (4.2) with  $\tau = \frac{\beta}{\alpha}$  and  $K = \frac{1}{\alpha}$ . ▲

By Theorem 3.5.4 the prototype first order systems is BIBO stable if  $\tau > 0$  and unstable if  $\tau < 0$ .

**Exercise 4.3.** Is the prototype first order system (4.2) BIBO stable when  $\tau = 0$ ? What if  $K = 0$ ?

**Exercise 4.4.** Is the state-space model of the prototype first order system asymptotically stable when  $u = 0$  and  $\tau < 0$ ? Hint: Use Proposition 3.4.2.

Table 4.1 lists key properties of the prototype first order system (4.2).

Table 4.1: Properties of prototype first order system (4.2).

Pole location	Zero location	Steady-state gain	Bandwidth
$s = -\frac{1}{\tau}$	None	$K$	$\frac{1}{\tau}$ rad/s.

**Exercise 4.5.** Verify that Table 4.1 is correct using Definitions 2.8.2, 3.6.3 and 3.8.3.

Using the ideas of either Section 3.3 or Section 2.7 (Examples 2.7.7 and 2.7.9) it is easy to show that the impulse response of the prototype first order system is given by

$$g(t) = \frac{K}{\tau} e^{-\frac{t}{\tau}}, \quad t \geq 0. \quad (4.4)$$

**Exercise 4.6.** Verify that (4.4) is the impulse response of (4.2).

Suppose that we sample the impulse response (4.4) every  $\tau$  seconds starting from  $t = 0$  as shown in Figure 4.2a. Doing so we obtain the sampled values

$$g(0) = \frac{K}{\tau}, \quad g(\tau) = e^{-1}g(0) \approx 0.37g(0), \quad g(2\tau) = e^{-2}g(0) \approx 0.14g(0), \quad g(3\tau) = e^{-3}g(0) \approx 0.05g(0)$$

which shows that the smaller  $\tau > 0$  is, the faster the impulse response decays to zero. Next we obtain the step response by setting  $u(t) = 1(t)$  so that

$$Y(s) = G(s)U(s) = \frac{K}{1 + \tau s} \frac{1}{s} = \frac{K}{s} - \frac{K}{s + \frac{1}{\tau}}$$

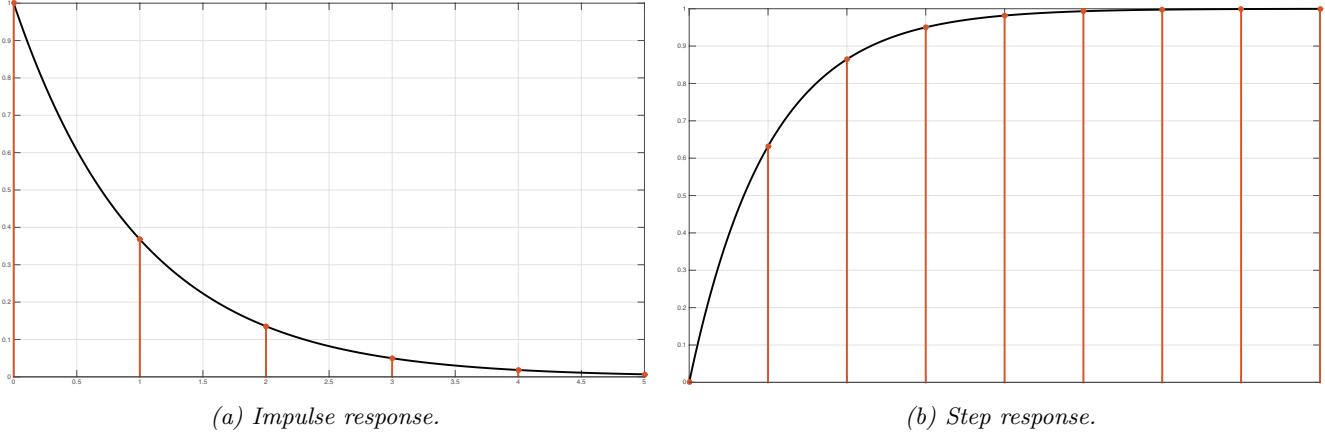


Figure 4.2: Responses of a prototype first order system with  $K = \tau = 1$  sampled every  $\tau$  seconds.

and, assuming zero initial conditions,

$$y(t) = \mathcal{L}^{-1}\{Y(s)\} = K \left(1 - e^{-\frac{t}{\tau}}\right), \quad t \geq 0. \quad (4.5)$$

Again, suppose that we sample the step response (4.5) every  $\tau$  seconds starting from  $t = 0$  as shown in Figure 4.2b. We obtain the sampled values

$$y(0) = 0, \quad y(\tau) = K(1 - e^{-1}) \approx 0.63y_{ss}, \quad y(2\tau) = K(1 - e^{-2}) \approx 0.86y_{ss}, \\ y(3\tau) = K(1 - e^{-3}) \approx 0.95y_{ss}, \quad y(4\tau) = K(1 - e^{-4}) \approx 0.98y_{ss}.$$

After  $4\tau$  seconds the step response has reached 98% of its steady-state value. The smaller  $\tau > 0$  is, the faster the step response converges to its steady-state value. The parameter  $\tau$  determines the rate at which  $e^{-t/\tau}$  decays to zero which motivates the subsequent definition.

**Definition 4.1.2.** The **time constant** of the prototype first order system is the parameter  $\tau$  in (4.2).

In summary, the time constant completely determines the transient response of the first order system. As  $\tau > 0$  approaches zero, the system pole moves towards  $-\infty$  along the real axis of the  $s$ -plane, the system's bandwidth approaches infinity and the system's time-domain response gets faster.

**Example 4.1.2.** In Example 2.9.1 proportional error feedback was applied to a prototype first order system. The resulting TF from the reference input to the plant output was found to be

$$\frac{Y(s)}{R(s)} = \frac{K_p K}{\tau s + 1 + K_p K}.$$

The time constant of the closed-loop system is  $\tau/(1+K_pK)$  and the closed-loop steady-state gain is  $K_pK/(1+K_pK)$ . The step response (4.5) is

$$y(t) = \frac{K_p K}{1 + K_p K} \left( 1 - e^{-\frac{t(1+K_p K)}{\tau}} \right), \quad t \geq 0.$$

By choosing large positive values of  $K_p$ , i.e., high gain control, the time constant of the closed-loop system can be made arbitrarily small and the steady-state gain arbitrarily close to one. However large values of  $K_p$  will result in large control signals and may not be physically implementable.  $\blacktriangleleft$

## 4.2 Prototype second order system

**Definition 4.2.1.** The **prototype second order system** with input  $u$  and output  $y$  is governed by the differential equation

$$\ddot{y} + 2\zeta\omega_n\dot{y} + \omega_n^2 y = K\omega_n^2 u, \quad K, \zeta, \omega_n \in \mathbb{R}. \quad (4.6)$$

The associated transfer function is

$$\frac{Y(s)}{U(s)} = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}. \quad (4.7)$$

To put (4.6) into state model form, let  $x := (y, \dot{y})$  to obtain (verify!)  $\dot{x} = Ax + Bu$ ,  $y = Cx + Du$  with

$$A = \begin{bmatrix} 0 & 1 \\ -\omega_n^2 & -2\zeta\omega_n \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ K\omega_n^2 \end{bmatrix}, \quad C = [1 \ 0], \quad D = 0.$$

The prototype second order system is very important because it is easy to analyse but it also has far richer dynamics than the first order system. This makes it a workhorse for obtaining simple models of many engineering systems.

**Example 4.2.1. (Mass-Spring-Damper)** The mass spring damper in Figure 4.3 from Examples 2.1.1 and 3.3.1 has transfer function

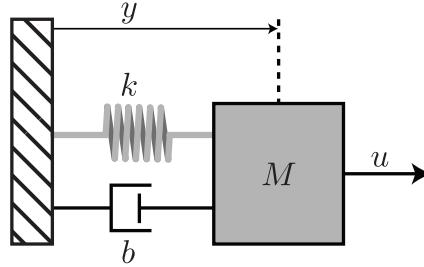


Figure 4.3: Mass-spring-damper system.

$$\frac{Y(s)}{U(s)} = \frac{1}{Ms^2 + bs + k}.$$

Comparing this transfer function to the prototype second order system (4.7) we deduce

$$\omega_n = \sqrt{\frac{k}{M}}, \quad K = \frac{1}{k}, \quad \zeta = \frac{b}{2\sqrt{kM}}.$$

**Example 4.2.2. (RLC Circuit)** Consider the series RLC circuit in Figure 4.4 (cf. Examples 2.3.5 and 2.4.5). The system TF is (verify!)

$$\frac{Y(s)}{U(s)} = G(s) = \frac{1}{s^2 LC + sRC + 1} = \frac{\frac{1}{LC}}{s^2 + \frac{R}{L}s + \frac{1}{LC}}.$$

Comparing this transfer function to the prototype second order system (4.7) we deduce

$$\omega_n = \frac{1}{\sqrt{LC}}, \quad K = 1, \quad \zeta = \frac{R}{2}\sqrt{\frac{C}{L}}.$$

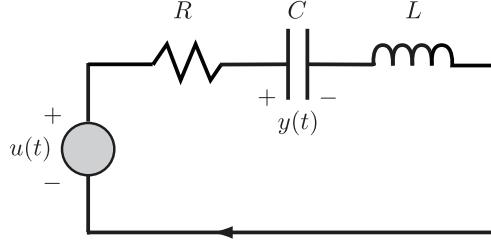


Figure 4.4: RLC circuit with voltage across capacitor taken as output.

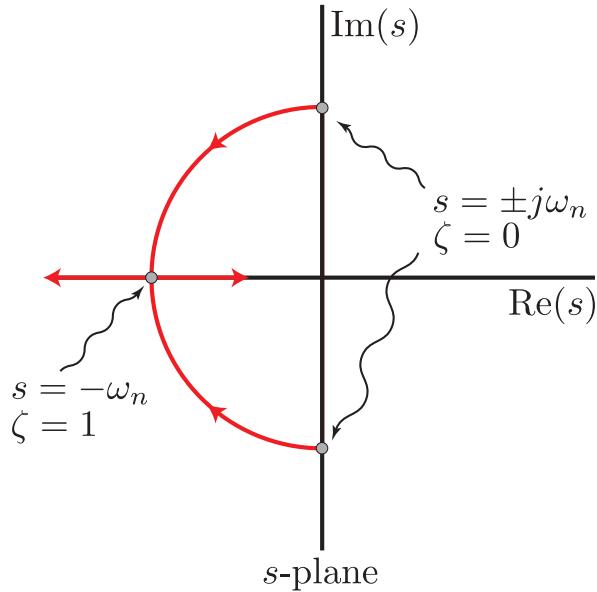
The prototype second order system's TF is real rational and it has no zeros so its poles are the roots of its denominator, i.e., the roots of  $s^2 + 2\zeta\omega_n s + \omega_n^2$ . These are easily computed using the quadratic formula

$$s = -\zeta\omega_n \pm \sqrt{(\zeta\omega_n)^2 - \omega_n^2} = \omega_n \left( -\zeta \pm \sqrt{\zeta^2 - 1} \right). \quad (4.8)$$

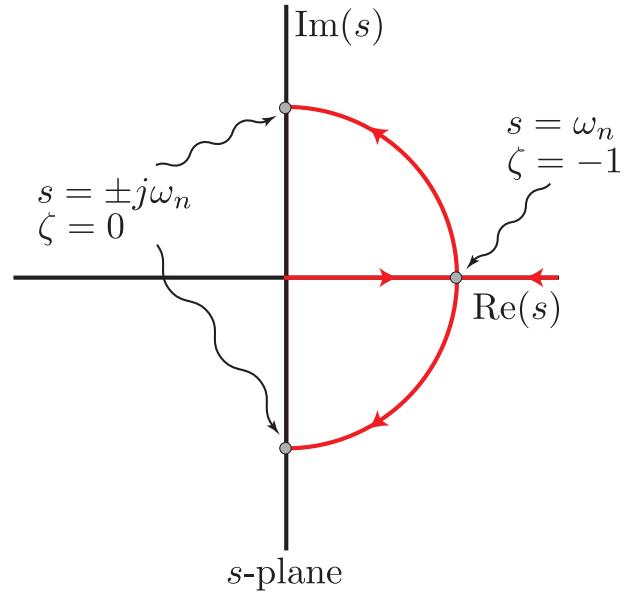
We observe from (4.8) that

- When  $\zeta > 1$  the system has two real, distinct, poles.
- When  $\zeta = 1$  the system has two real, repeated, poles.
- When  $0 < \zeta < 1$  the system has two complex conjugate poles.
- When  $\zeta \leq 0$  the system is unstable.

Figure 4.5 shows how the poles of the prototype second order system (4.7) vary on the  $s$ -plane as a function of  $\zeta$  for fixed  $\omega_n > 0$ .



(a)  $\zeta$  ranging from 0 to  $+\infty$ .



(b)  $\zeta$  ranging from  $-\infty$  to 0.

Figure 4.5: Pole locations of the prototype second order system as a function of  $\zeta$  with fixed  $\omega_n$ .

Just as the time constant characterizes the prototype first order system, the parameter  $\zeta$  is a useful way to categorize the prototype second order system and has its own name.

**Definition 4.2.2.** The **damping ratio** of the prototype second order system is the parameter  $\zeta$  in (4.7). The **natural undamped frequency** of the prototype second order system is the parameter  $\omega_n$  in (4.7). The prototype second order system is called:

- (i) **undamped** if  $\zeta = 0$ ,
- (ii) **underdamped** if  $0 < \zeta < 1$ ,
- (iii) **critically damped** if  $\zeta = 1$ ,
- (iv) **overdamped** if  $\zeta > 1$ .

The step responses for an underdamped, critically damped and over damped prototype second order system are shown in Figure 4.6.

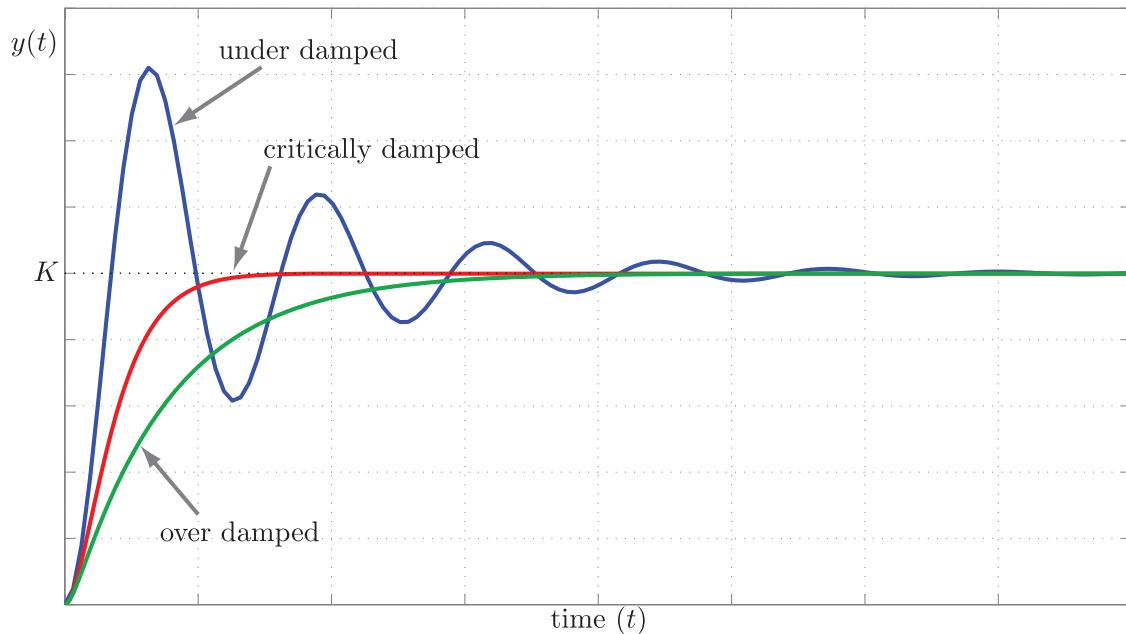


Figure 4.6: Representative step responses of the prototype second order system for different values of  $\zeta$  and fixed  $\omega_n$ .

**Exercise 4.7.** Simulate and plot the step response of an undamped prototype second order system.

Table 4.2 lists key properties of the prototype second order system (4.7). In the rest of this section we study

Table 4.2: Properties of prototype second order system (4.7).

Name	Poles	Zeros	Steady-state gain	Step response
Underdamped	$-\zeta\omega_n \pm j\omega_n\sqrt{1 - \zeta^2}$	None	$K$	Oscillatory
Critically damped	$-\omega_n$ (repeated)	None	$K$	Not oscillatory
Overdamped	$-\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1}$	None	$K$	Not oscillatory

in some detail the properties of underdamped, critically damped and overdamped second order systems.

### 4.2.1 Underdamped systems

**Example 4.2.3. (Mass-Spring-Damper)** Using the expression for the damping ratio from Example 4.2.1 we conclude that the mass-spring-damper system is underdamped if and only if the damper constant  $b$  satisfies  $0 < b < 2\sqrt{kM}$ .  $\blacktriangle$

**Exercise 4.8.** Suppose that  $L = 160 \mu\text{H}$  and  $C = 10 \mu\text{F}$  in the RLC circuit from Example 4.2.2. Find the range of resistance values so that the circuit is underdamped.

The poles of an underdamped system, written in Cartesian form, are  $s = -\zeta\omega_n \pm j\omega_n\sqrt{1 - \zeta^2}$ .

**Exercise 4.9.** Verify that

$$\left| -\zeta\omega_n \pm j\omega_n\sqrt{1 - \zeta^2} \right| = \omega_n, \quad \angle(-\zeta\omega_n \pm j\omega_n\sqrt{1 - \zeta^2}) = \pm(\pi - \arccos(\zeta)).$$

In light of Exercise 4.9, the polar form of the poles of an underdamped prototype second order system is

$$s = \omega_n e^{\pm j(\pi - \arccos(\zeta))}.$$

The angle of the poles is completely determined by the damping ratio  $\zeta$  while the magnitude is determined by the undamped natural frequency  $\omega_n$ , see Figure 4.7. We now analyse the frequency response of an underdamped

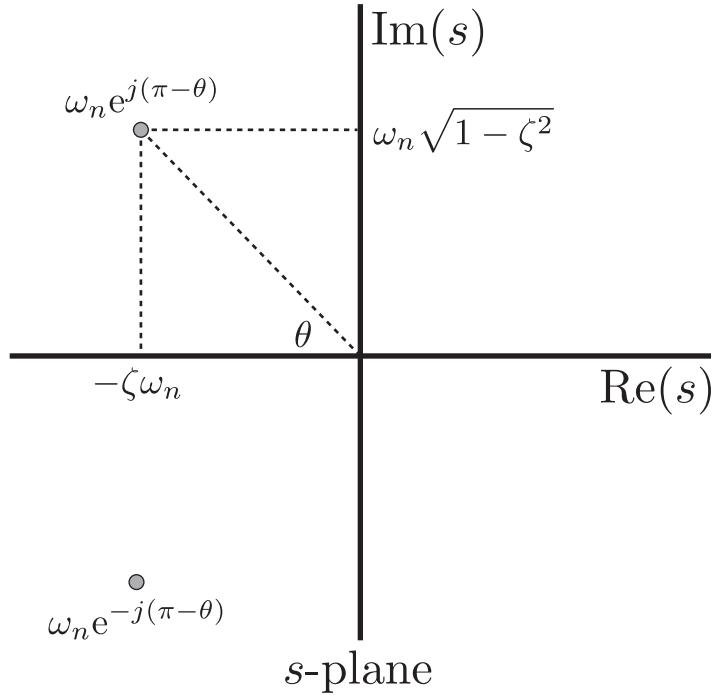


Figure 4.7: Pole location of an underdamped second order system. In this figure  $\theta = \arccos(\zeta)$ .

prototype second order system. If we assume that the steady-state gain is unity, i.e.,  $K = 1$ , then we can write the TF (4.7) as

$$G(s) = \frac{1}{\left(\frac{s}{\omega_n}\right)^2 + 2\zeta\left(\frac{s}{\omega_n}\right) + 1}.$$

Recall from Section 3.8.3 that this is exactly the standard form for drawing the Bode plot of a system with complex conjugate roots. The Bode plot for an underdamped system with  $K = \omega_n = 1$  is shown in Figure 4.8 for various different values of  $\zeta$ . We observe from Figure 4.8 that the system bandwidth is approximately  $\omega_n$  and

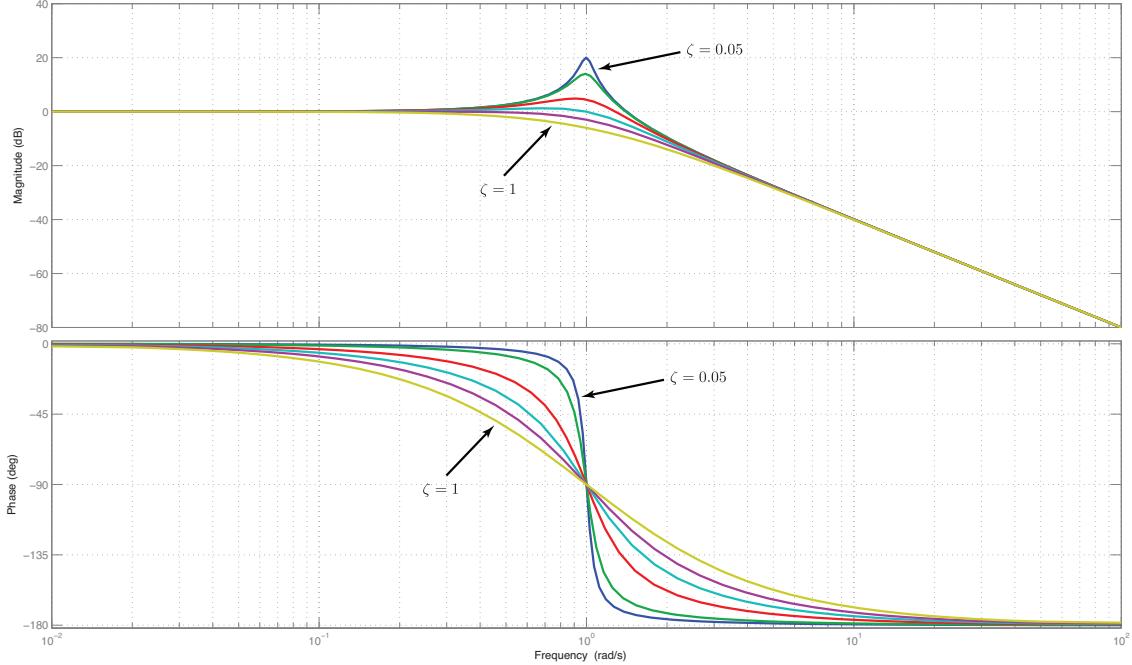


Figure 4.8: Bode plot of the prototype second order system with  $K = \omega_n = 1$  for  $\zeta \in \{0.05, 0.1, 0.3, 0.5, 1/\sqrt{2}, 1\}$ .

that, when  $0 < \zeta < 1/\sqrt{2}$ , the magnitude plot has a “peak.” It can be shown that the value of the magnitude plot at that peak is

$$20 \log \left| \frac{1}{2\zeta\sqrt{1-\zeta^2}} \right|$$

and that it occurs at  $\omega = \omega_n\sqrt{1-\zeta^2}$ .

Having identified the bandwidth of the underdamped system ( $\omega_{BW} \approx \omega_n$ ) we turn to the impulse and step responses to see how the damping ratio and undamped natural frequency affect them. We start with the impulse response. The impulse response of the underdamped second order system is found, using partial fraction expansions and doing some messy calculations, to be

$$g(t) = K \frac{\omega_n}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t} \sin(\omega_n\sqrt{1-\zeta^2}t), \quad t \geq 0. \quad (4.9)$$

Observe that the decay rate of the impulse response is determined by the exponential part  $e^{-\zeta\omega_n t}$  while the frequency of oscillation is  $\omega_n\sqrt{1-\zeta^2}$ . Therefore the decay rate depends on the real part of the poles (see Figure 4.7) while the oscillation frequency depends on the imaginary part of the poles (see Figure 4.7). Figure 4.9 shows the impulse response of various underdamped systems. In these figures, as the parameter  $\zeta$  gets closer and closer to zero (and the poles become purely imaginary), the response becomes more oscillatory.

To find the step response we can either integrate the impulse response or use partial fraction expansions on

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

to get

$$y(t) = K \left( 1 - \frac{1}{\sqrt{1-\zeta^2}} e^{-\zeta\omega_n t} \sin(\omega_n\sqrt{1-\zeta^2}t + \arccos(\zeta)) \right), \quad t \geq 0. \quad (4.10)$$

Figure 4.10 shows the step response of various underdamped systems.

Notice that in the equations for both the impulse response (4.9) and the step response (4.10) that whenever time  $t$  appears, it is multiplied by  $\omega_n$ . Therefore, for fixed  $\zeta$ , the parameter  $\omega_n$  acts as a time-scale factor and

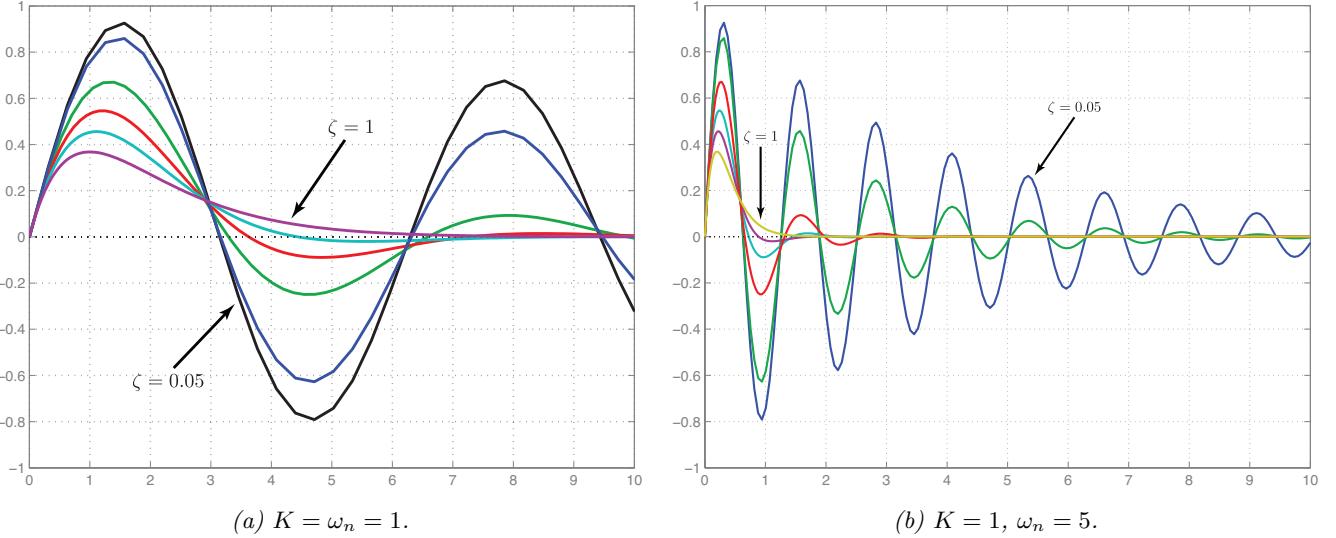


Figure 4.9: Impulse response of an underdamped system with  $\zeta \in \{0.05, 0.1, 0.3, 0.5, 1/\sqrt{2}, 1\}$ .

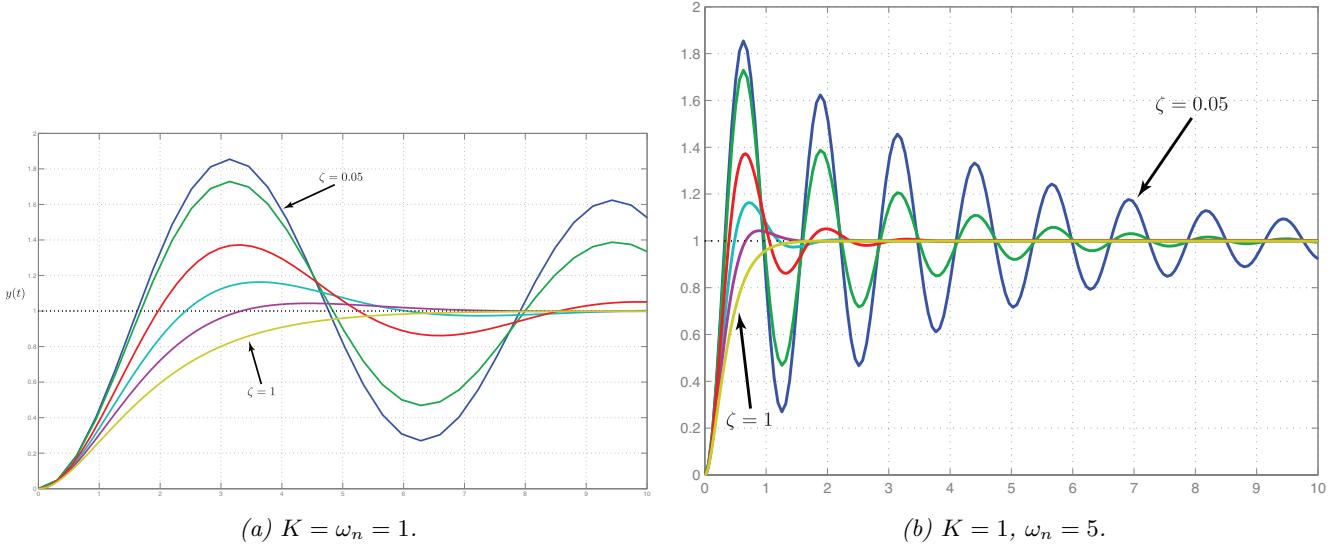


Figure 4.10: Step response of an underdamped system with  $\zeta \in \{0.05, 0.1, 0.3, 0.5, 1/\sqrt{2}, 1\}$ .

as  $\omega_n$  gets larger the response speeds up. Similarly, as  $\omega_n$  gets smaller, the response slows down. Therefore, we once again see that the higher the system bandwidth, the faster the response.

These plots also show that as  $\zeta$  gets closer to 1, equivalently as the complex conjugate poles get closer to the real axis, the response become less oscillatory. The undamped natural frequency  $\omega_n$  is the frequency of oscillation when  $\zeta = 0$  which provides some justification for its name.

Just as in the first order case, we see that the further to the left in  $\mathbb{C}^-$  that the poles are, the faster the response. Also, the closer the poles are to the real axis (for fixed  $\omega_n$ ), the less oscillatory the response. We will make much use of the above observations in analyzing and designing control systems.

## 4.2.2 Overdamped systems

The prototype second order system (4.7) with  $\zeta > 1$  has two distinct real poles located at

$$s = -\zeta\omega_n \pm \omega_n\sqrt{\zeta^2 - 1}. \quad (4.11)$$

By factoring the denominator of the TF (4.7) we can re-write the transfer function in the form

$$G(s) = \frac{Kab}{(s+a)(s+b)}$$

where  $a$  and  $b$  are the roots (4.11).

**Exercise 4.10.** Show that the bandwidth of the overdamped system approximately equals  $\omega_{\text{BW}} = \min \{|a|, |b|\}$ .

The impulse response of an overdamped system can be found using partial fraction expansions

$$\begin{aligned} g(t) &= \mathcal{L}^{-1}\{G(s)\} \\ &= K\mathcal{L}^{-1} \left\{ \frac{\frac{ab}{b-a}}{s+a} + \frac{\frac{ab}{a-b}}{s+b} \right\} \\ &= K \frac{ab}{b-a} (e^{-at} - e^{-bt}), \quad t \geq 0. \end{aligned} \tag{4.12}$$

The time derivative of the impulse response is

$$\frac{d}{dt}g(t) = K \frac{ab}{b-a} (be^{-bt} - ae^{-at}).$$

Setting this derivative equal to zero and solving for  $t$  we get

$$\bar{t} = \frac{\ln(a) - \ln(b)}{a - b}.$$

This means that  $g(\bar{t})$  is an extreme value of the impulse response: the absolute value of the the impulse peaks at that time. The step response can also be found using partial fraction expansions. If

$$Y(s) = \frac{Kab}{(s+a)(s+b)} \frac{1}{s} = \frac{K}{s} + \frac{K}{b-a} \left( \frac{a}{s+b} - \frac{b}{s+a} \right).$$

then

$$y(t) = K \left( 1 - \frac{1}{b-a} (ae^{-bt} - be^{-at}) \right), \quad t \geq 0. \tag{4.13}$$

The time derivative of the step response is

$$\frac{d}{dt}y(t) = K \frac{ab}{b-a} (e^{-at} - e^{-bt}).$$

This shows that  $\dot{y} = 0$  only when  $t = 0$ . In other words, there are no extreme values of  $y(t)$  for  $t \geq 0$  and the step response does not have “peaks.” Figure 4.11 shows typical impulse and step responses of an overdamped system.

**Exercise 4.11.** Find the pole locations of the system used to create the plots in Figure 4.11.

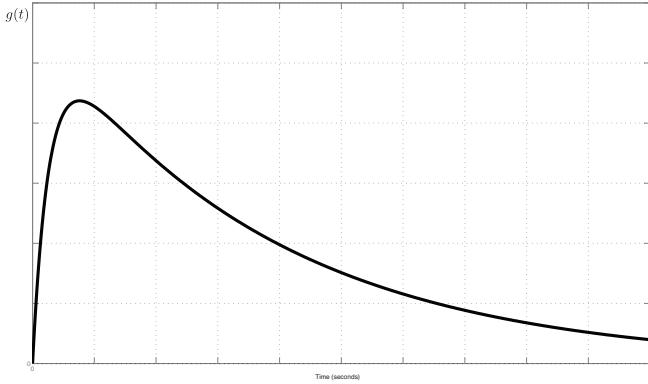
### 4.2.3 Critically damped systems

The prototype second order system (4.7) with  $\zeta = 1$  has two repeated real poles located at

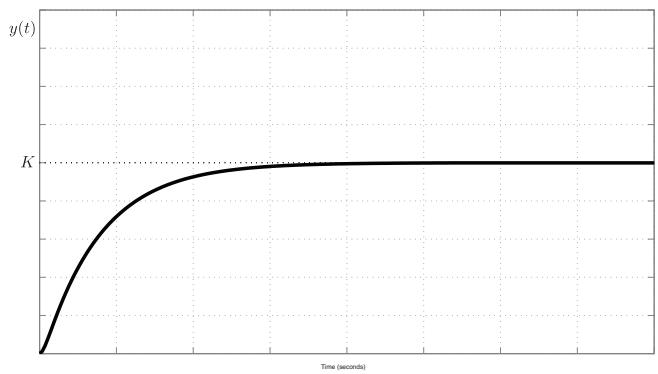
$$s = -\zeta\omega_n = -\omega_n. \tag{4.14}$$

Therefore, in the critically damped case, we can re-write the system’s TF as

$$G(s) = \frac{K\omega_n^2}{(s + \omega_n)^2}.$$



(a) Impulse response.



(b) Step response.

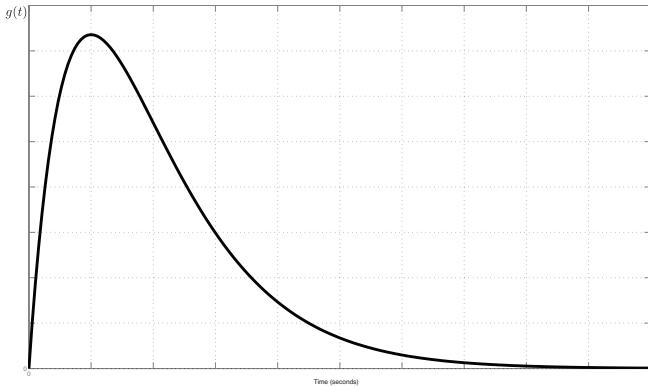
Figure 4.11: Time response of an overdamped system with  $\zeta = 2$ ,  $\omega_n = 1$ .

**Exercise 4.12.** Show that the bandwidth of a critically damped system is approximately  $\omega_{BW} = |\omega_n|$ .

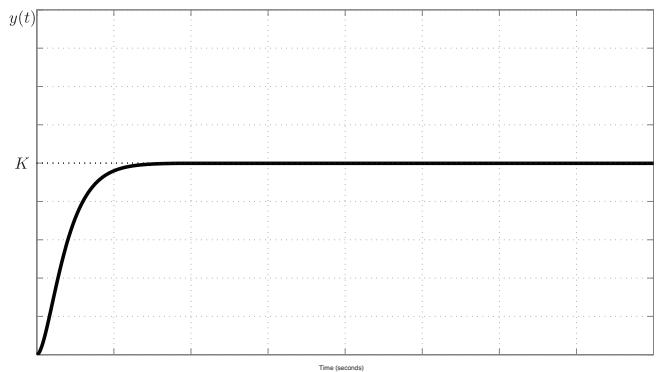
The unit step response can be found using the inverse LT and partial fraction expansions

$$y(t) = K \left( 1 - (1 + \omega_n t) e^{-\omega_n t} \right), \quad t \geq 0. \quad (4.15)$$

Using similar analysis as in the overdamped case, Section 4.2.2, you can show that the critically damped system has no “peaks” in its step response. Its response is slower than a underdamped system’s response but it does not “overshoot” its steady-state value during its transient period. Figure 4.12 shows the typical impulse and step responses of an overdamped systems.



(a) Impulse response.



(b) Step response.

Figure 4.12: Time response of a critically damped system with  $\zeta = 1$ ,  $\omega_n = 1$ .

## 4.3 General characteristics of a step response

In this section we look at common measures used to quantify the quality of a step response. These measures can be applied to *any* stable system, even if it isn’t first or second order. The key characteristics of a step response are illustrated in Figure 4.13. While the quantities in Figure 4.13 can be numerically or experimentally found for a system of any order, we will see that the prototype first and second order systems are useful because we can obtain closed-form expressions for the various characteristics. These closed-form expressions motivate many of our design equations.

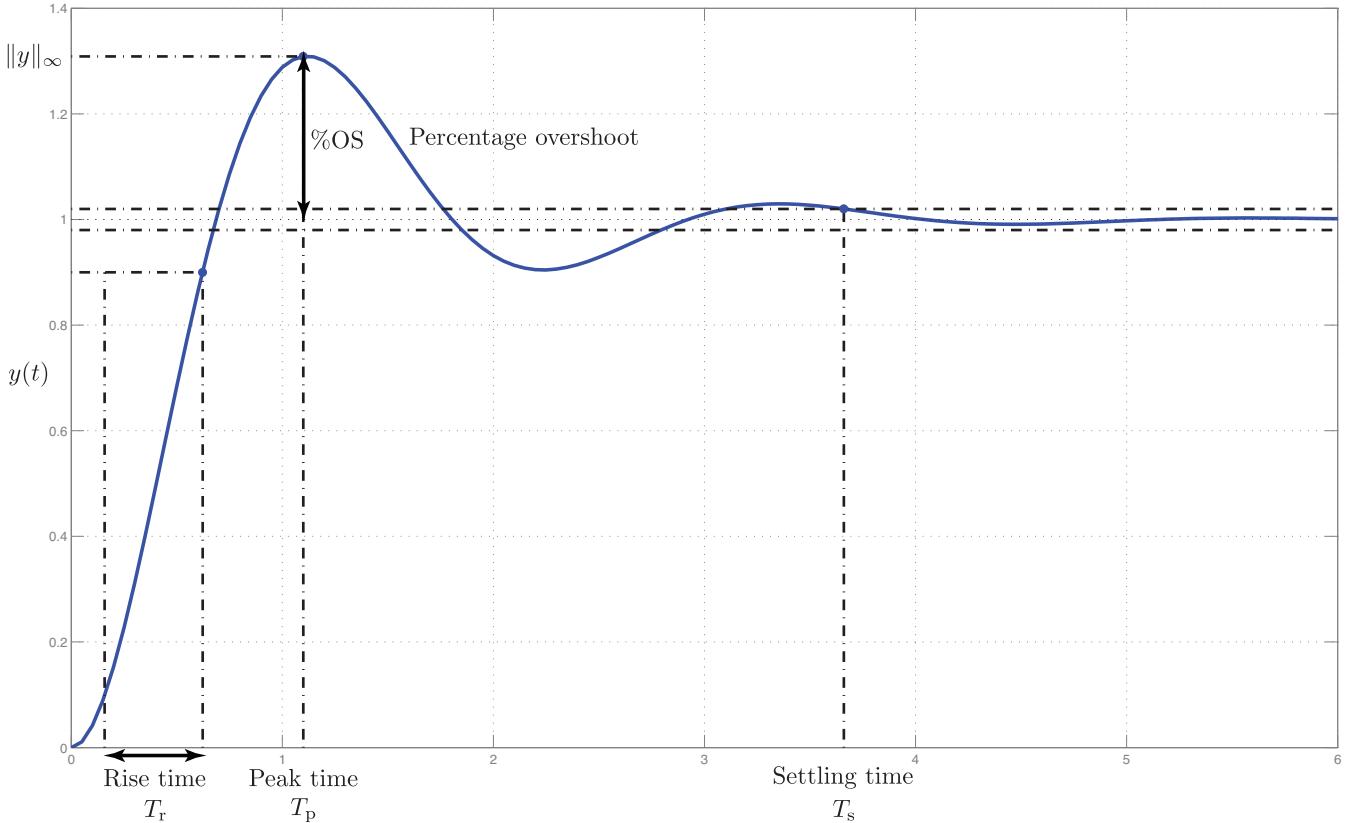


Figure 4.13: Measures of step response performance.

### 4.3.1 Overshoot

Informally, overshoot refers to the situation when the magnitude of a stable system's step response gets larger than the magnitude of its steady-state value. We have seen that stable underdamped systems always have overshoot while first order systems, critically damped and overdamped systems never do.

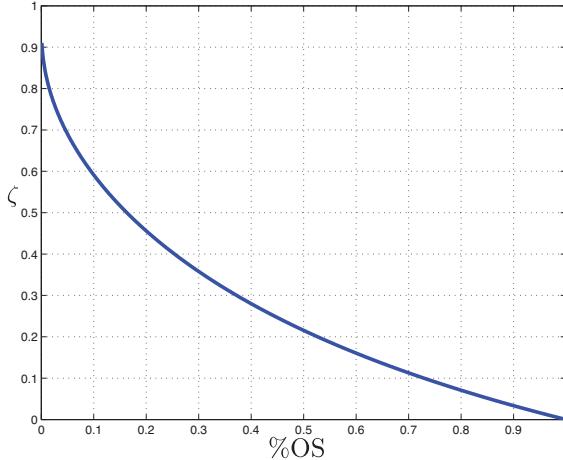
**Definition 4.3.1.** The **percentage overshoot** of a stable LTI system with scalar TF  $G(s)$  and unit step response  $y(t) = \mathcal{L}^{-1}(G(s)\frac{1}{s})$  is

$$\%OS := \frac{\|y\|_\infty - |G(0)|}{|G(0)|}.$$

To derive an expression for the overshoot of an underdamped prototype second order system (4.7), we differentiate the step response (4.10) and then solve for the smallest time  $\bar{t} > 0$  such that  $\dot{y}(\bar{t}) = 0$ . Then  $\|y\|_\infty = |y(\bar{t})|$  is the least upper bound of the step response (see Figure 4.13). Doing these calculations yields the following formula

$$\%OS = e^{\frac{-\zeta\pi}{\sqrt{1-\zeta^2}}} \quad (\text{underdamped prototype second order system}). \quad (4.16)$$

Notice that the percentage overshoot only depends on the damping ratio  $\zeta$  and not on  $\omega_n$ . We rearrange (4.16) to obtain the percentage overshoot as a function of the damping ratio



$$\zeta = -\frac{\ln(\%OS)}{\sqrt{\pi^2 + (\ln(\%OS))^2}}. \quad (4.17)$$

Figure 4.14: Damping ratio as a function of %OS (4.17).

Observe that the more damping (larger  $\zeta$ ) the less overshoot. We can use this observation to help us impose constraints on a system's damping ratio as follows. Let  $\%OS^{max}$  denote the maximum allowable overshoot. Then using (4.17) the specification  $\%OS^{max}$  is translated into a specification on the damping ratio

$$\%OS \leq \%OS^{max} \iff \zeta \geq -\frac{\ln(\%OS^{max})}{\sqrt{\pi^2 + (\ln(\%OS^{max}))^2}}.$$

Lastly, since  $\zeta = \cos(\theta)$  where  $\theta$  is the angle in Figure 4.7,

$$\%OS \leq \%OS^{max} \iff \theta \leq \arccos\left(-\frac{\ln(\%OS^{max})}{\sqrt{\pi^2 + (\ln(\%OS^{max}))^2}}\right) =: \theta^{max}.$$

Therefore, in order to meet an overshoot specification, the second order system must have all its poles in a “cone” in the left half of the  $s$ -plane

$$\{s \in \mathbb{C} : |\arg(s)| \geq \pi - \theta^{max}\}.$$

**Example 4.3.1.** Suppose that  $L = 160 \mu\text{H}$  and  $C = 10 \mu\text{F}$  in the RLC circuit from Example 4.2.2. Find the range of resistor values  $R > 0$  so that the system is (i) stable (ii) underdamped and (iii) its step response satisfies  $\%OS \leq 0.05$ .

From Exercise 4.8 we know that (i) and (ii) are satisfied if  $R < 8 \Omega$ . In this example  $\%OS^{max} = 0.05$  so that (iii) is satisfied if

$$\zeta \geq -\frac{\ln(0.05)}{\sqrt{\pi^2 + (\ln(0.05))^2}} = 0.69.$$

Using the expression for  $\zeta$  from Example 4.2.2 we obtain the constraint

$$\zeta = \frac{R}{2} \sqrt{\frac{C}{L}} = \frac{R}{8} \geq 0.69 \iff R \geq 5.52.$$

In summary, the design specifications are met if  $5.52 \leq R < 8$ . ▲

**Exercise 4.13.** Draw the region in the  $s$ -plane in which the poles of an under damped second order system must lie so that  $\%OS < 0.05$ .

### 4.3.2 Settling time

Settling time refers to how long it takes for the step response to lie within 2% of its final value.

**Definition 4.3.2.** The **2% settling time** of a stable LTI system with scalar TF  $G(s)$  and unit step response  $y(t) = \mathcal{L}^{-1}(G(s)\frac{1}{s})$  is the smallest  $T_s > 0$  such that

$$\text{for all } t \geq T_s \quad \frac{|G(0) - y(t)|}{|G(0)|} \leq 0.02.$$

For an underdamped second order system, based on the expression for its step response (4.10), we can crudely estimate the settling time by looking at the envelope  $e^{-\zeta\omega_n t}$  to get

$$e^{-\zeta\omega_n t} \leq 0.02.$$

Solving for  $t$  we get the approximate inequality  $t \geq 4/\zeta\omega_n$  to obtain<sup>1</sup>

$$T_s \approx \frac{4}{\zeta\omega_n} \quad (\text{underdamped prototype second order system}). \quad (4.18)$$

From this relationship we see that the larger the  $\omega_n$ , the faster the response. Note that  $T_s$  only depends on the real part of the poles of  $G(s)$ . This is consistent with our previous observations and motivates the following rule of thumb for underdamped second order systems: the higher the bandwidth (larger  $\omega_n$ ) the faster the response (smaller  $T_s$ ).

Suppose our design specification is that the settling time must be less than  $T_s^{\max}$ . This means that

$$T_s \leq T_s^{\max} \Leftrightarrow \zeta\omega_n \geq \frac{4}{T_s^{\max}}.$$

Therefore, in order to meet a settling time specification, the second order system must have all its poles sufficiently far to the left of the  $s$ -plane

$$\left\{ s \in \mathbb{C} : \operatorname{Re}(s) \leq -\frac{4}{T_s^{\max}} \right\}.$$

**Exercise 4.14.** Find the settling time for the system from Example 4.3.1 when  $R = 0.4\Omega$ .

**Exercise 4.15.** Draw the region in the  $s$ -plane in which the poles of an under damped second order system must lie so that  $T_s < 3$  seconds.

### 4.3.3 Time to peak

The peak time  $T_p$  is the amount of time it takes for the step response to reach its maximum value.

**Definition 4.3.3.** The **peak time** or **time to peak** of a stable LTI system with scalar TF  $G(s)$  and unit step response  $y(t) = \mathcal{L}^{-1}(G(s)\frac{1}{s})$  is the smallest  $T_p > 0$  such that

$$\|y\|_\infty = |y(T_p)|.$$

---

<sup>1</sup>It can be shown that the 2% settling time for a critically damped system is approximately given by  $T_s \approx \frac{5.8}{\omega_n}$ . Unless explicitly stated, we will use (4.18) for computing the settling time of second order systems.

To derive an expression for  $T_p$  for an underdamped prototype second order system (4.7), we (cf. Section 4.3.1) differentiate the step response (4.10) and then solve for the smallest time  $T_p > 0$  such that  $\dot{y}(T_p) = 0$ . Doing so yields

$$T_p = \frac{\pi}{\omega_n \sqrt{1 - \zeta^2}} \quad (\text{underdamped prototype second order system}). \quad (4.19)$$

Once again we observe that the larger the bandwidth ( $\omega_n$ ), the smaller  $T_p$  is and the faster the response becomes. Also note that  $T_p$  only depends on the imaginary part of the poles for a second order system.

**Exercise 4.16.** Draw the region in the  $s$ -plane in which the poles of an underdamped second order system must lie so that  $T_p < 3$  seconds.

#### 4.3.4 Rise time

The rise time  $T_r$  of a step response is the time it takes for it to go from 10% of its final value to 90% of its final value.

**Definition 4.3.4.** Let  $G(s)$  be the scalar TF of a stable LTI system with unit step response  $y(t) = \mathcal{L}^{-1}(G(s)\frac{1}{s})$ . Let  $t_1 > 0$  and  $t_2 > 0$  be the smallest times such that  $y(t_1) = 0.1y_{ss}$  and  $y(t_2) = 0.9y_{ss}$ . Then the **rise time** of  $G(s)$  is

$$T_r = t_2 - t_1.$$

The approximate relationship for a second order system is

$$T_r \approx \frac{2.16\zeta + 0.6}{\omega_n} \quad (0.3 < \zeta < 0.8). \quad (4.20)$$

Once again we observe that the larger the bandwidth (larger  $\omega_n$ ), the faster the response.

## 4.4 The effect of adding poles and zeros

We stated at the start of this chapter that second order systems are very useful in the analysis and design of more general LTI systems, i.e., higher order systems and systems with zeros. In this section we start to justify this statement by looking at what happens when additional poles and zeros are added to the prototype second order system.

### 4.4.1 Adding a stable pole

We start with the prototype second order system (4.7) but now we multiply it with a stable first order system with a steady-state gain of one, i.e.,

$$G_a(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \frac{1}{1 + \tau s}, \quad \tau > 0 \quad (\text{augmented plant}).$$

The pole that we've added is located at  $s = -1/\tau \in \mathbb{C}^-$ . As  $\tau \rightarrow +\infty$  the new pole approaches  $s = 0$ . As  $\tau \rightarrow 0$  the pole approaches  $-\infty$ . From the expression for  $G_a(s)$  we see that as the new pole approaches  $-\infty$  ( $\tau \rightarrow 0$ ) we recover<sup>2</sup> the TF of the second order system (4.7).

---

<sup>2</sup>In this case the effect of the additional pole is only noticeable when analyzing the high frequency response.

To see the effect on the system's step response when the pole approaches  $s = 0$  ( $\tau \rightarrow \infty$ ) response let's assume that  $K = 1$  and re-write  $G_a(s)$  using an appropriate partial fraction expansion

$$G_a(s) = \frac{a}{1 + \tau s} + \frac{bs + c}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

where

$$a = \frac{\tau^2\omega_n^2}{1 - 2\zeta\omega_n\tau + \omega_n^2\tau^2}, \quad b = -\frac{a}{\tau}, \quad c = \frac{\zeta\omega_n}{\tau} \left( \frac{1}{\tau\zeta\omega_n} - 2 \right) a.$$

From this we see that as  $\tau \rightarrow \infty$ , we get  $b, c \rightarrow 0$  and the augmented system's response approaches a first order response.

**Example 4.4.1.** Consider the underdamped second order system

$$G(s) = \frac{8}{s^2 + 2s + 4}.$$

Here  $\omega_n = 2$  and  $\zeta = \frac{1}{2}$ . Consider the augmented plant

$$G_a(s) = G(s) \frac{1}{1 + \tau s} = \frac{8}{s^2 + 2s + 4} \frac{1}{1 + \tau s}.$$

Figure 4.15 shows the Bode plots and step response of this system for  $\tau \in \{0, 0.01, 0.1, 0.2, 1, 2, 10\}$ .

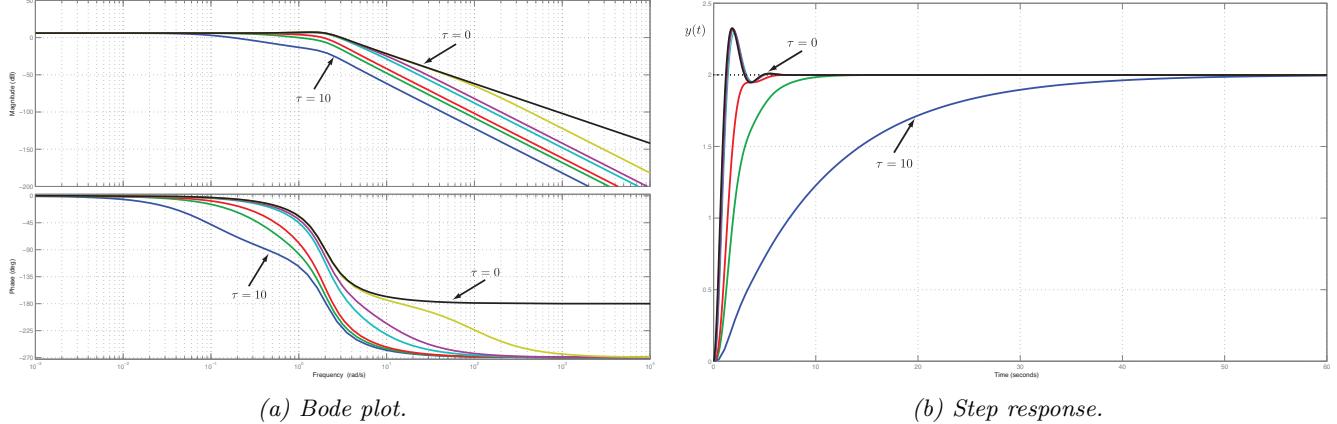


Figure 4.15: Effect of adding a stable pole to a second order system (Example 4.4.1).

We summarize our observations as follows: As the additional pole gets closer to the origin (i.e.,  $\tau$  increases) the bandwidth decreases, the phase is more negative and the step response is more sluggish. These effects become less prominent as the extra pole moves further into  $\mathbb{C}^-$  (i.e.,  $\tau$  decreases).

#### 4.4.2 Adding a left half-plane zero

We again start with the prototype second order system (4.7) but now we augment it with a minimum phase zero

$$G_a(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} (\tau s + 1), \quad \tau > 0.$$

The zero we've added is located at  $s = -1/\tau$ . As  $\tau \rightarrow +\infty$  the zero approaches  $s = 0$ . As  $\tau \rightarrow 0$  the zero approaches  $-\infty$ . To see the effect on the system response let's assume that  $K = 1$  and find the step response

$$\begin{aligned} y(t) &= \mathcal{L}^{-1} \left\{ G_a(s) \frac{1}{s} \right\} \\ &= \mathcal{L}^{-1} \left\{ \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} (\tau s + 1) \frac{1}{s} \right\} \\ &= \mathcal{L}^{-1} \left\{ \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \frac{1}{s} + \tau s \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \frac{1}{s} \right\} \\ &= \mathcal{L}^{-1} \left\{ \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \frac{1}{s} + \tau \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \right\} \\ &= \text{step response of } G(s) + \tau \times \text{impulse response of } G(s). \end{aligned}$$

From this we conclude that as the extra zero approaches  $-\infty$  (i.e.,  $\tau \rightarrow 0$ ) then  $y(t)$  approaches the standard second order step response. As the zero gets closer to  $s = 0$  ( $\tau \rightarrow +\infty$ ) the impulse response term becomes more noticeable in the step response.

**Example 4.4.2.** Consider, as in the previous example,

$$G(s) = \frac{8}{s^2 + 2s + 4}$$

and the augmented plant

$$G_a(s) = G(s)(\tau s + 1) = \frac{8}{s^2 + 2s + 4}(\tau s + 1).$$

Figure 4.16 shows the Bode plot and step response of this system for  $\tau \in \{0, 0.5, 1, 1.5, 2\}$ . ▲

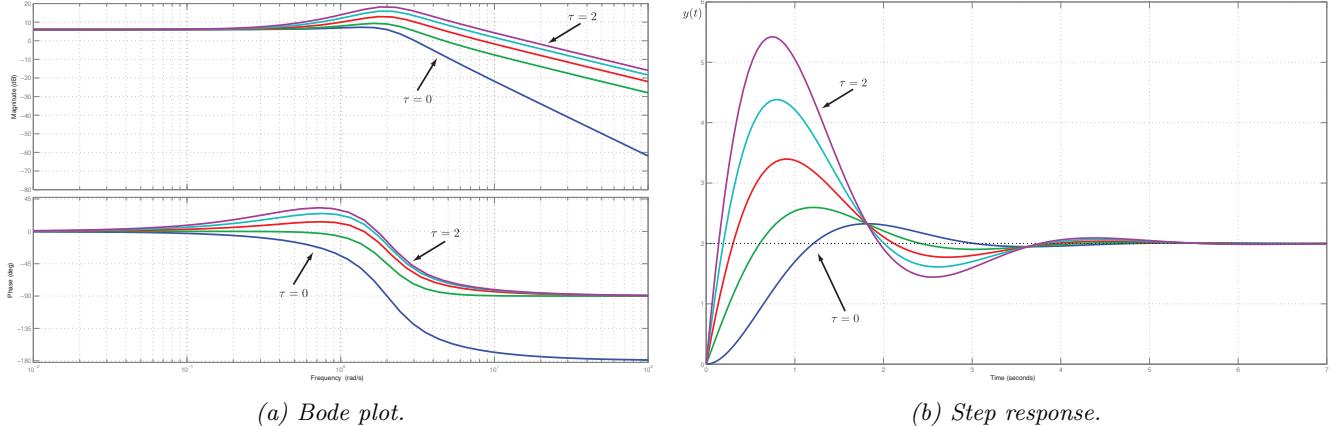


Figure 4.16: Effect of adding a minimum phase zero to a second order system (Example 4.4.2).

We summarize our observations as follows: As the minimum phase zero moves closer to the origin, i.e., as  $\tau$  increases, the bandwidth increases, the phase is more positive and the response gets faster but more oscillatory. As the zero moves closer to  $-\infty$ , i.e., as  $\tau$  decreases, the response approaches that of the prototype second order system. Because of these relationships we will generally try to keep zeros away from the origin.

#### 4.4.3 Adding a right half-plane zero

Zeros in the closed right half plane are called non-minimum phase (see Definition 3.8.2). Starting once again with the prototype second order system (4.7) we augment it with a right half-plane zero

$$G_a(s) = \frac{K\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} (1 - \tau s), \quad \tau > 0.$$

The added zero is located at  $s = 1/\tau$ . Generally, a non-minimum phase zero makes a system hard to control (see Section 9.6.2). It slows the system down and causes the system to go in the “wrong direction” at first. To see why, let’s assume that  $K = 1$  and find the step response

$$\begin{aligned} y(t) &= \mathcal{L}^{-1} \left\{ G_a(s) \frac{1}{s} \right\} \\ &= \mathcal{L}^{-1} \left\{ \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} (1 - \tau s) \frac{1}{s} \right\} \\ &= \mathcal{L}^{-1} \left\{ \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \frac{1}{s} - \tau s \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2} \frac{1}{s} \right\} \\ &= \text{step response of } G(s) - \tau \times \text{impulse response of } G(s). \end{aligned}$$

The negative sign in front of the impulse response in the above derivation is what causes the response to “go the wrong way.”

**Example 4.4.3.** Consider again

$$G(s) = \frac{8}{s^2 + 2s + 4}$$

and the augmented system

$$G_a(s) = G(s)(1 - \tau s) = \frac{8}{s^2 + 2s + 4}(1 - \tau s).$$

Figure 4.17 shows the Bode plots and step response of this system for  $\tau \in \{0.1, 0.5, 1, 1.5, 2\}$ . ▲

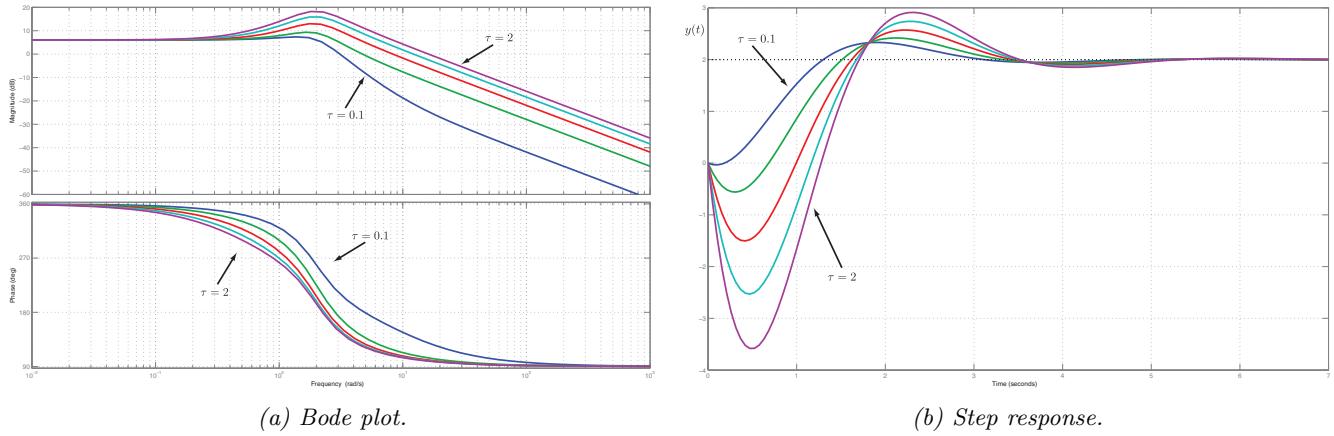


Figure 4.17: Effect of adding a non-minimum phase zero to a second order system (Example 4.4.3).

We summarize this section as follows: As the non-minimum phase zero moves closer to the origin, i.e., as  $\tau$  increases, the bandwidth increases, the phase is more negative. Furthermore, the step response goes further in the wrong direction near  $t = 0$ .

## 4.5 Dominant poles and zeros

The formulas we derived in Section 4.3 apply to underdamped second order systems. In general, it is very difficult to obtain closed-form formulas to evaluate the performance of higher order systems. However, in many cases, there are natural separations among the system poles and zeros. For example, some poles and zeros are much closer to the imaginary axis than others as illustrated in Figure 4.18.

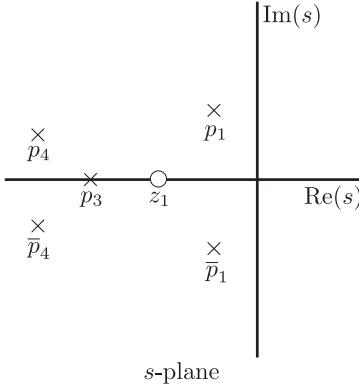


Figure 4.18: Pole-zero configuration with  $p_1, \bar{p}_1$  as dominant poles.

In Section 4.4 we showed that poles and zeros “far to the left” do not have a significant impact on the low frequency response<sup>3</sup> of a system. Hence good low-order approximations can be found by appropriately neglecting the less significant poles and zeros. Typically, we neglect poles and zeros that are at least 5 times further away from the imaginary axis. The poles and zeros close to the imaginary axis are called **dominant poles** and **dominant zeros**. The poles and zeros at least 5 times further away from the imaginary axis are called **non-dominant poles** and **non-dominant zeros**.

#### Example 4.5.1. (Model Reduction)

$$\begin{aligned} G(s) &= \frac{s+10}{(s+11)(s+12)(s^2+2s+2)} \\ &= \underbrace{\frac{s+10}{(s+11)(s+12)}}_{=:G_{\text{fast}}(s)} \underbrace{\frac{1}{s^2+2s+2}}_{=:G_{\text{slow}}(s)}. \end{aligned}$$

The portion of the time response due to  $G_{\text{fast}}(s)$  reaches steady-state much faster than the portion of the time response due to  $G_{\text{slow}}(s)$ . This motivates us to approximate  $G_{\text{fast}}(s)$  by its steady-state gain

$$G_{\text{fast}}(0) = \frac{10}{132}$$

and hence

$$G(s) \approx G_{\text{fast}}(0)G_{\text{slow}}(s) = \frac{10}{132} \frac{1}{s^2+2s+2} =: \hat{G}(s).$$

Figure 4.19a shows the frequency response of  $G(s)$  and  $\hat{G}(s)$ . As expected, at low frequencies the approximation is reasonably good. The step responses for the actual and approximated systems are shown in Figure 4.19b. ▲

When reducing the order of system model it is important to determine the frequency range in which the approximation should be valid.

**Example 4.5.2.** Consider the transfer function

$$G(s) = 100 \frac{(1 + \frac{1}{60}s)(1 + \frac{1}{900}s)}{(1 + \frac{1}{100}s)(1 + \frac{1}{500}s)(1 + \frac{1}{600}s)(1 + \frac{1}{1000}s)}.$$

We want to obtain a model that describes the process well in the frequency range  $\omega \leq 600$ . In this case

$$G_{\text{fast}}(s) = \frac{1 + \frac{1}{900}s}{1 + \frac{1}{1000}s}$$

<sup>3</sup>The Bode plot is affected at high frequencies where the gain is typically small.

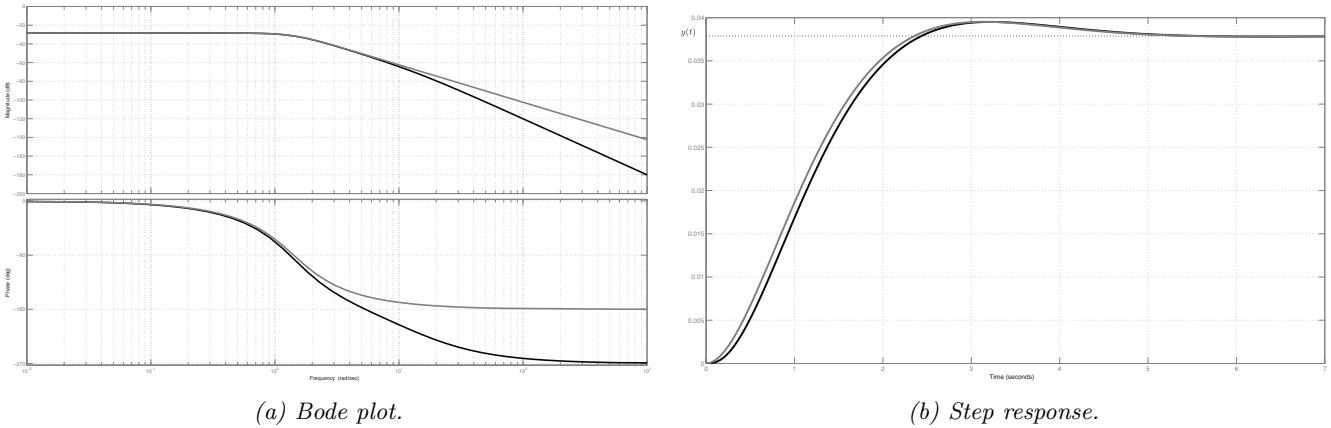


Figure 4.19: Bode and step responses of  $G(s)$  (black) and  $\hat{G}(s)$  (grey) from Example 4.5.1.

and the approximated transfer function is

$$\hat{G}(s) = 100 \frac{\left(1 + \frac{1}{60}s\right)}{\left(1 + \frac{1}{100}s\right) \left(1 + \frac{1}{500}s\right) \left(1 + \frac{1}{600}s\right)}.$$



## 4.6 Summary

The main purpose of this chapter is to build intuition about how pole locations affect the time-domain response and bandwidth of LTI systems. Upon finishing this chapter you should know the following.

1. The form of the TF for prototype first and second order systems.
2. Understand how changing the time constant  $\tau$  changes the time domain response and bandwidth of a prototype first order system.
3. The relationship between damping ratio  $\zeta$ , undamped natural frequency  $\omega_n$  and the pole locations of the prototype second order system (see Figures 4.5, 4.7).
4. The qualitative differences in the step responses for under-, over- and critically damped second order systems.
5. Perhaps most importantly, understand how changing the damping ratio and undamped natural frequency changes the step response of an underdamped prototype second order system.
6. Section 4.3 is very important and serves as a basis for many design techniques. You should know the common measures used to quantify the performance of a step response.
7. Compute the overshoot, rise time, settling and time to peak for an underdamped second order systems. You should also be able to translate specifications on these parameters into allowable pole locations for second order systems. You should be able to draw these allowable pole locations on the complex  $s$ -plane.
8. Qualitatively understand the effect that adding poles and zeros to a second order system has on its step response. You should be able to relate this effect to the location of the additional pole/zero.
9. Know what is meant by “dominant poles.”
10. Find a reduced order model of a system when there are dominant poles.

# Chapter 5

## Feedback control theory

In this chapter we start to develop the basic theory and tools for feedback control analysis and design in the frequency domain. “Analysis” means you already have a controller and you want to study how good it is; “design” of course means you want to design a controller to meet certain specifications. The most fundamental specification is stability. Typically, good performance requires high-gain controllers, yet typically the feedback loop will become unstable if the gain is too high.

### Contents

5.1	Closing the loop	114
5.2	Stability of feedback systems	118
5.3	The Routh-Hurwitz criterion	125
5.4	Steady-state performance	129
5.5	Summary	135

### 5.1 Closing the loop

We start with a series of examples to show how a typical control design might go. The first example deals with modelling.

#### Example 5.1.1. (Pendulum on a Cart - modelling)

A favourite toy control problem is to get a cart to automatically balance a pendulum<sup>1</sup> as shown in Figure 5.1. We follow the modelling procedure from Section 2.3.1.

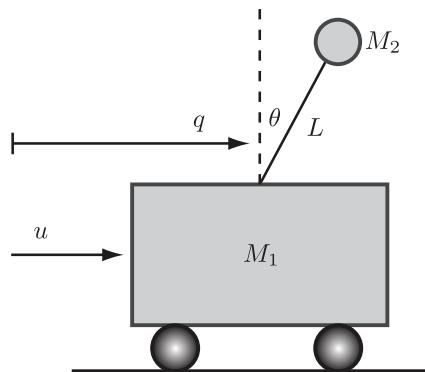


Figure 5.1: Pendulum on a cart.

<sup>1</sup>The pendulum in this example is a mathematical pendulum as opposed to the physical pendulum from Example 2.3.3. The difference is that in a mathematical pendulum the mass is assumed to be concentrated at the end and connected by a massless, rigid rod of length  $L$ .

1. Our reference frame is such that down is the positive vertical direction and right is the positive horizontal direction. Positive angular displacements correspond to clockwise rotations of the pendulum.
2. The coordinates are already suggested by the figure to be  $(q, \theta)$ .
3. The system has two mechanical components: the cart and the ball.
4. The free body diagrams are shown in Figure 5.2.

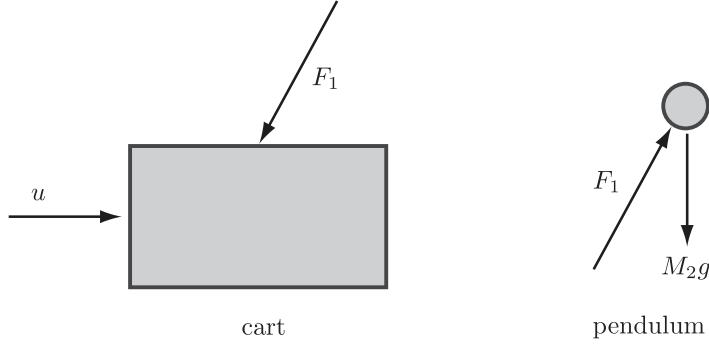


Figure 5.2: Free body diagram for the pendulum on a cart.

5. The position of the centre of mass of the cart is  $q$ . We ignore its vertical position since it has no motion in that direction. The position of the centre of mass of the ball is  $q + L \sin(\theta)$  (horizontal direction) and  $L - L \cos(\theta)$  (vertical direction). So when  $\theta = 0$  (upward vertical position), the ball is located at  $(q, 0)$ . When  $\theta = \pi$  (downward vertical position) the ball is located at  $(q, 2L)$ .
6. Newton's law for the cart is

$$M_1\ddot{q} = u - F_1 \sin \theta.$$

Newton's law for the ball in the horizontal direction is

$$M_2 \frac{d^2}{dt^2}(q + L \sin \theta) = F_1 \sin \theta$$

and in the vertical direction is

$$M_2 \frac{d^2}{dt^2}(L - L \cos \theta) = M_2 g - F_1 \cos \theta.$$

These are three differential equations in the four signals  $q, \theta, u, F_1$ . Use

$$\frac{d^2}{dt^2} \sin \theta = \ddot{\theta} \cos \theta - \dot{\theta}^2 \sin \theta, \quad \frac{d^2}{dt^2} \cos \theta = -\ddot{\theta} \sin \theta - \dot{\theta}^2 \cos \theta$$

to get

$$\begin{aligned} M_2 \ddot{q} + M_2 L \ddot{\theta} \cos \theta - M_2 L \dot{\theta}^2 \sin \theta &= F_1 \sin \theta \\ M_2 L \ddot{\theta} \sin \theta + M_2 L \dot{\theta}^2 \cos \theta &= M_2 g - F_1 \cos \theta \\ M_1 \ddot{q} &= u - F_1 \sin \theta. \end{aligned}$$

We can eliminate  $F_1$ : Add the first and the third to get

$$(M_1 + M_2)\ddot{q} + M_2 L \ddot{\theta} \cos \theta - M_2 L \dot{\theta}^2 \sin \theta = u;$$

multiply the first by  $\cos \theta$ , the second by  $\sin \theta$ , add, and cancel  $M_2$  to get

$$\ddot{q} \cos \theta + L \ddot{\theta} = g \sin \theta.$$

Solve the latter two equations for  $\ddot{q}$  and  $\ddot{\theta}$ :

$$\begin{bmatrix} M_1 + M_2 & M_2 L \cos \theta \\ \cos \theta & L \end{bmatrix} \begin{bmatrix} \ddot{q} \\ \ddot{\theta} \end{bmatrix} = \begin{bmatrix} u + M_2 L \dot{\theta}^2 \sin \theta \\ g \sin \theta \end{bmatrix}.$$

Thus

$$\begin{aligned} \ddot{q} &= \frac{u + M_2 L \dot{\theta}^2 \sin \theta - M_2 g \sin \theta \cos \theta}{M_1 + M_2 \sin^2 \theta} \\ \ddot{\theta} &= \frac{-u \cos \theta - M_2 L \dot{\theta}^2 \sin \theta \cos \theta + (M_1 + M_2) g \sin \theta}{L(M_1 + M_2 \sin^2 \theta)}. \end{aligned}$$

The natural state is (positions and velocities)  $x = (x_1, x_2, x_3, x_4) = (q, \theta, \dot{q}, \dot{\theta})$  so that in terms of state variables we have

$$\begin{aligned} \dot{x}_1 &= x_3 \\ \dot{x}_2 &= x_4 \\ \dot{x}_3 &= \frac{u + M_2 L x_4^2 \sin x_2 - M_2 g \sin x_2 \cos x_2}{M_1 + M_2 \sin^2 x_2} \\ \dot{x}_4 &= \frac{-u \cos x_2 - M_2 L x_4^2 \sin x_2 \cos x_2 + (M_1 + M_2) g \sin x_2}{L(M_1 + M_2 \sin^2 x_2)}. \end{aligned}$$

Again, these have the form

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

We might take the output to be the positions

$$y = \begin{bmatrix} q \\ \theta \end{bmatrix} = \begin{bmatrix} x_1 \\ x_2 \end{bmatrix} = h(x).$$

The system is highly nonlinear; as you would expect, it can be approximated by a linear system for  $|\theta|$  small enough, say less than  $5^\circ$ . ▲

In order to apply the analysis and design tools of this course we need an LTI plant. In the next example we linearize the pendulum on the cart.

**Example 5.1.2. (Pendulum on a Cart - linearization)** An equilibrium configuration (Definition 2.5.1)  $\bar{x} = (\bar{x}_1, \bar{x}_2, \bar{x}_3, \bar{x}_4)$ ,  $\bar{u}$  satisfies  $f(\bar{x}, \bar{u}) = 0$ , i.e.,

$$\bar{x}_3 = 0$$

$$\bar{x}_4 = 0$$

$$\bar{u} + M_2 L \bar{x}_4^2 \sin \bar{x}_2 - M_2 g \sin \bar{x}_2 \cos \bar{x}_2 = 0$$

$$-\bar{u} \cos \bar{x}_2 - M_2 L \bar{x}_4^2 \sin \bar{x}_2 \cos \bar{x}_2 + (M_1 + M_2) g \sin \bar{x}_2 = 0.$$

Multiply the third equation by  $\cos \bar{x}_2$  and add to the fourth: We get in sequence

$$-M_2 g \sin \bar{x}_2 \cos^2 \bar{x}_2 + (M_1 + M_2) g \sin \bar{x}_2 = 0$$

$$\sin \bar{x}_2 (M_1 + M_2 \sin^2 \bar{x}_2) = 0$$

$$\sin \bar{x}_2 = 0$$

$$\bar{x}_2 = 0 \text{ or } \pi.$$

Thus the equilibrium configurations are described by

$$\bar{x} = (\text{arbitrary}, 0 \text{ or } \pi, 0, 0), \quad \bar{u} = 0.$$

This makes good physical sense because the system is at rest when there are no applied forces; all the velocities are zero; the pendulum is pointing either up or down; and the position of the cart doesn't matter. We have to choose  $\bar{x}_2 = 0$  (pendulum up) or  $\bar{x}_2 = \pi$  (pendulum down). Let's take  $\bar{x}_2 = 0$ . Then the Jacobians compute to

$$A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -\frac{M_2}{M_1}g & 0 & 0 \\ 0 & \frac{M_1+M_2}{M_1} \frac{g}{L} & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ \frac{1}{M_1} \\ -\frac{1}{LM_1} \end{bmatrix}.$$

▲

**Remark 5.1.1.** Example 5.1.2 uses the general method of linearizing introduced in Section 2.5. In this particular example, there's a faster way, which is to approximate  $\sin \theta \approx \theta, \cos \theta \approx 1$  in the original equations:

$$\begin{aligned} M_1 \ddot{q} &= u - F_1 \theta \\ M_2 \frac{d^2}{dt^2}(q + L\theta) &= F_1 \theta \\ 0 &= M_2 g - F_1. \end{aligned}$$

These equations are already linear and lead to the above matrices  $A$  and  $B$ . ♦

Now we close the loop on the cart-pendulum system.

**Example 5.1.3. (Pendulum on a Cart - control design)** Consider the linearized cart-pendulum from the previous example. Take  $M_1 = 1 \text{ kg}$ ,  $M_2 = 2 \text{ kg}$ ,  $L = 1 \text{ m}$ ,  $g = 9.8 \text{ m/s}^2$ . Then the state-space model is

$$\dot{x} = Ax + Bu, \quad A = \begin{bmatrix} 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \\ 0 & -19.6 & 0 & 0 \\ 0 & 29.4 & 0 & 0 \end{bmatrix}, \quad B = \begin{bmatrix} 0 \\ 0 \\ 1 \\ -1 \end{bmatrix}.$$

Let's suppose we designate the cart position as the only output:  $y = x_1$ . Then

$$C = [1 \ 0 \ 0 \ 0].$$

The transfer function from  $u$  to  $y$  is (verify!)

$$\frac{Y(s)}{U(s)} = P(s) = \frac{s^2 - 9.8}{s^2(s^2 - 29.4)}.$$

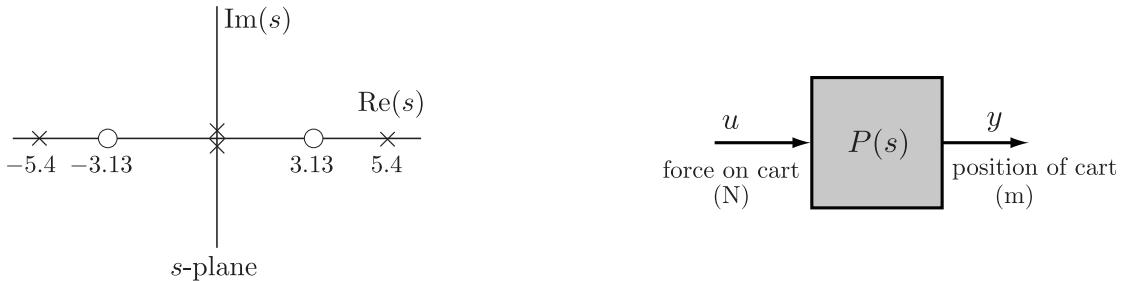


Figure 5.3: Linearized pendulum on a cart.

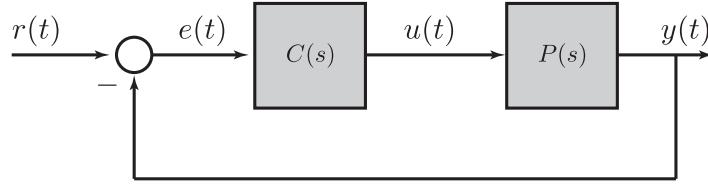


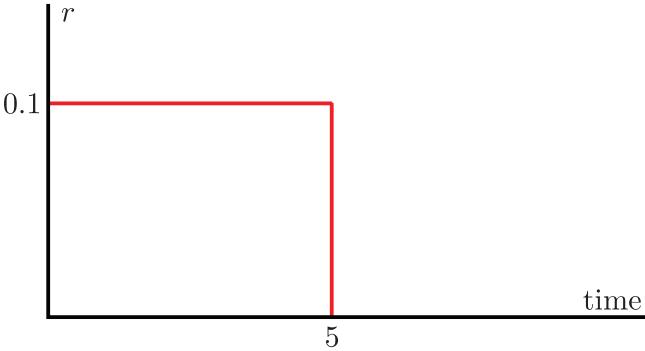
Figure 5.4: Control architecture for cart-pendulum example.

The poles and zeros of  $P(s)$  are shown in Figure 5.3a. Having three poles in  $\text{Re}(s) \geq 0$ , the plant is quite unstable. The right half-plane zero doesn't contribute to the degree of instability, but, as we shall see, it does make the plant quite difficult to control. The block diagram of the plant by itself is shown in Figure 5.3b.

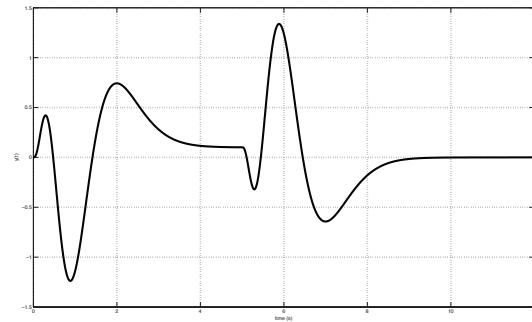
Let us try to stabilize the plant by feeding back the cart position,  $y$ , comparing it to a reference  $r$ , and setting the error  $r - y$  as the controller input as shown in Figure 5.4. Here  $C(s)$  is the transfer function of the controller to be designed. One controller that does in fact stabilize is

$$C(s) = \frac{10395s^3 + 54126s^2 - 13375s - 6687}{s^4 + 32s^3 + 477s^2 - 5870s - 22170}.$$

The controller itself,  $C(s)$ , is unstable, as is  $P(s)$ . But when the controller and plant are connected in feedback, the system is stable. If the pendulum starts to fall, the controller causes the cart to move, in the appropriate direction, to make the pendulum tend to come vertical again. You're invited to simulate the closed-loop system; for example, let  $r$  be the input shown in Figure 5.5a. This reference signal corresponds to a command that



(a) Reference signal (desired cart position).



(b) Output signal (cart position).

Figure 5.5: Closed-loop response of pendulum on a cart (Example 5.1.3).

the cart moves right 0.1 m for 5 seconds, then return to its original position. Figure 5.5b is a plot of the cart position  $x_1$  versus  $t$ . The cart moves rather wildly as it tries to balance the pendulum—it's not a *good* controller design—but it does stabilize.

We mentioned that our controller  $C(s)$  is open-loop unstable. It can be proved (it's beyond the scope of this course) that every controller that stabilizes this  $P(s)$  is itself unstable (see the notion of **strong stabilization** in, for instance, [Doyle et al., 1990]). ▲

## 5.2 Stability of feedback systems

Our objective in this section is to define what it means for the feedback system in Figure 5.6 to be stable. We will give two definitions. The two definitions look quite distinct at first, but they are exactly the same under mild assumptions.

Consider the setup in Figure 5.6. The signals and systems in this figure are:

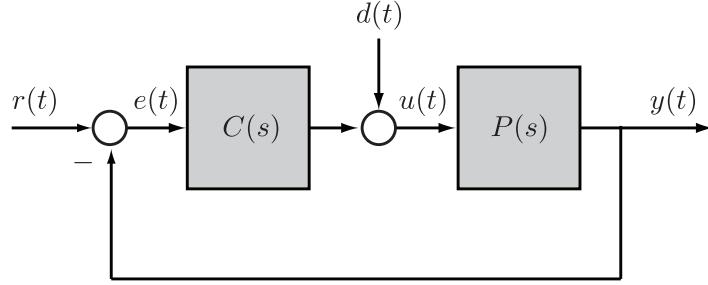


Figure 5.6: Unity feedback system.

systems     $P(s)$ , plant transfer function  
 $C(s)$ , controller transfer function

signals     $r$ , reference (or command) input  
 $e$ , tracking error  
 $d$ , disturbance  
 $u$ , plant input  
 $y$ , plant output.

The signals coming from the outside —  $r$  and  $d$  — are called **exogenous inputs**. The next example shows there is actually something to talk about in regards to stability of interconnected transfer functions and illustrates the difficulties that can arise due to even simple interconnections.

**Example 5.2.1. (Unstable Pole-Zero Cancellation)** Consider the series connection shown in Figure 5.7. Using the ideas of Section 2.9 we have

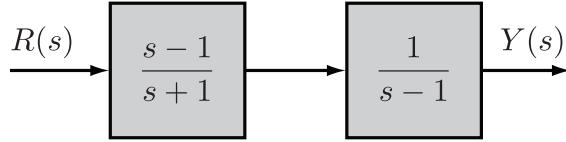


Figure 5.7: Cascade connection of Example 5.2.1.

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

which by Theorem 3.5.4 is BIBO stable. On these grounds we'd say that the system is BIBO stable, wash our hands of the stability question and walk away. In doing so we'd be too hasty. To see why this is so, suppose that the system admits some noise as in Figure 5.8.

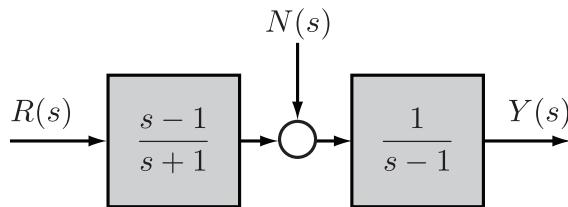


Figure 5.8: Noise at the interface of two systems.

The transfer function from  $n$  to  $y$  is

$$\frac{Y(s)}{N(s)} = \frac{1}{s-1}.$$

which by Theorem 3.5.4 is unstable. Thus *any* slight perturbation in the signal as it passes from the first system to the second will potentially be dangerously magnified in the output. ▲

**Definition 5.2.1.** The feedback system in Figure 5.6 is **well-posed** if  $1 + C(s)P(s)$  is not identically equal to zero. Otherwise, it is said to be **ill-posed**.

An ill-posed system is not meaningful, does not work properly, and should be avoided.

**Example 5.2.2. (Ill-Posed)** Consider the feedback system in Figure 5.6 with

$$P(s) = 1, \quad C(s) = -1.$$

In this case  $1 + C(s)P(s) \equiv 0$  and the system is ill-posed. The dependent signals  $e$ ,  $u$  and  $y$  cannot be uniquely determined from the independent signals  $r$  and  $d$ .  $\blacktriangle$

We make the following standing assumption throughout this section.

**Assumption 5.2.2.** The plant  $P(s)$  and controller  $C(s)$  are rational, the controller  $C(s)$  is proper, and the plant  $P(s)$  is strictly proper.  $\blacktriangleleft$

**Exercise 5.1.** Show that Assumption 5.2.2 implies that  $1 + PC$  is not identically zero.

Assumption 5.2.2 is also ensures that the closed-loop system is causal as illustrated in the next example

**Example 5.2.3.** Consider the feedback system in Figure 5.6 with

$$C(s)P(s) = \frac{1 + 2s + as^2}{s^2 + s + 1}$$

where  $a \in \mathbb{R}$  is unspecified. If  $a \neq 0$  then Assumption 5.2.2 is violated. The transfer function from  $r$  to  $y$  is (see Section 2.9)

$$\frac{Y(s)}{R(s)} = \frac{C(s)P(s)}{1 + C(s)P(s)} = \frac{1 + 2s + as^2}{(1 + a)s^2 + 3s + 2}.$$

If, and only if,  $a = -1$  the TF is improper and hence non-causal<sup>2</sup>.  $\blacktriangle$

### 5.2.1 Internal stability

For this concept we set the reference input  $r(t)$  and the disturbance input  $d(t)$  to zero, i.e.,  $r(t) = d(t) = 0$ . We bring in state-space models for the plant and controller as in Figure 5.9. The closed-loop equations are

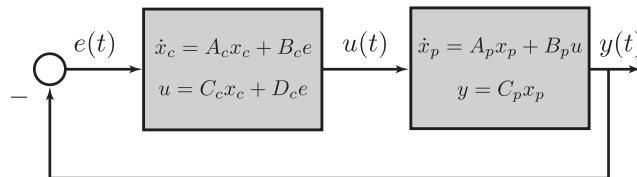


Figure 5.9: Internal stability.

$$\begin{aligned}\dot{x}_p &= A_p x_p + B_p u \\ \dot{x}_c &= A_c x_c + B_c e \\ u &= C_c x_c + D_c e \\ e &= -C_p x_p.\end{aligned}$$

<sup>2</sup>And, incidentally, also unstable by Theorem 3.5.5.

Let the closed-loop state be  $x_{cl} := (x_p, x_c)$  so that (verify!)

$$\dot{x}_{cl} = A_{cl}x_{cl} := \begin{bmatrix} A_p - B_p D_c C_p & B_p C_c \\ -B_c C_p & A_c \end{bmatrix} \begin{bmatrix} x_p \\ x_c \end{bmatrix}.$$

**Definition 5.2.3.** The feedback system in Figure 5.6 is **internally stable** if the state model  $\dot{x}_{cl} = A_{cl}x_{cl}$  is asymptotically stable.

The concept of internal stability means that with no exogenous inputs applied ( $r = d = 0$ ), the internal states  $x_p(t)$ ,  $x_c(t)$  decay to zero for every initial state  $x_p(0)$ ,  $x_c(0)$ . By Proposition 3.4.2 the feedback system is internally stable if and only if all the eigenvalues of the matrix  $A_{cl}$  have negative real part.

**Example 5.2.4.** Suppose the plant has state-space model

$$\begin{aligned}\dot{x}_p &= x_p + u \\ y &= x_p\end{aligned}$$

so that  $A_p = 1$ ,  $B_p = 1$  and  $C_p = 1$  and  $P(s) = 1/(s - 1)$ . The controller is given by

$$u = 2e$$

so that  $A_c = 0$ ,  $B_c = 0$ ,  $C_c = 0$ ,  $D_c = 2$  and  $C(s) = 2$ . Therefore

$$A_{cl} = A_p - B_p C_c C_p = -1.$$

The  $1 \times 1$  matrix  $A_{cl}$  has one eigenvalue at  $\lambda = -1$ . Therefore, the unstable plant  $1/(s - 1)$  is internally stabilized by unity feedback with a proportional (pure gain) controller 2.  $\blacktriangle$

**Example 5.2.5.** We re-visit Example 5.2.1 and show that we cannot internally stabilize the plant  $P(s) = 1/(s - 1)$  by cancelling the unstable pole. Consider the controller  $C(s) = (s - 1)/(s + 1)$  and the system in Figure 5.7. We saw in Example 5.2.1 that the transfer function from  $r$  to  $y$  is BIBO stable but that any noise at the interface of the systems will potentially be dangerously magnified in the output. We now show that while the instability is not apparent by only looking at input and output signals, it is apparent when we examine internal stability.

Figure 5.10 shows state models for each transfer function. We have, with  $r = 0$ ,

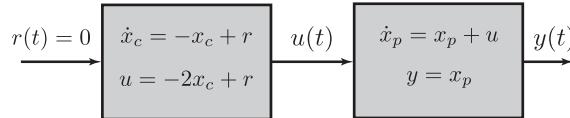


Figure 5.10: State-space model for Example 5.2.5.

$$\begin{aligned}\dot{x}_p &= x_p - 2x_c \\ \dot{x}_c &= -x_c\end{aligned}$$

and therefore

$$\begin{bmatrix} \dot{x}_p \\ \dot{x}_c \end{bmatrix} = \begin{bmatrix} 1 & -2 \\ 0 & -1 \end{bmatrix} \begin{bmatrix} x_p \\ x_c \end{bmatrix}.$$

The system is not internally stable since the eigenvalues of this matrix are  $\{1, -1\}$ .  $\blacktriangle$

### 5.2.2 Input-output stability

We now turn the second way of thinking about stability of the feedback loop in Figure 5.6. To do so, we first find all of the transfer functions from the independent signals  $r$  and  $d$  to the dependent signals  $e$ ,  $u$  and  $y$ . We use our systematic method of doing block diagram reduction. First, write the equations for the outputs of the summing junctions:

$$\begin{aligned} E &= R - PU \\ U &= D + CE. \end{aligned}$$

Assemble into a vector equation:

$$\begin{bmatrix} 1 & P \\ -C & 1 \end{bmatrix} \begin{bmatrix} E \\ U \end{bmatrix} = \begin{bmatrix} R \\ D \end{bmatrix}.$$

In view of our standing assumption ( $P$  strictly proper,  $C$  proper), the determinant of

$$\begin{bmatrix} 1 & P \\ -C & 1 \end{bmatrix}$$

is not identically zero. Thus we can uniquely solve for  $E, U$ :

$$\begin{bmatrix} E \\ U \end{bmatrix} = \begin{bmatrix} 1 & P \\ -C & 1 \end{bmatrix}^{-1} \begin{bmatrix} R \\ D \end{bmatrix} = \begin{bmatrix} \frac{1}{1+PC} & \frac{-P}{1+PC} \\ \frac{C}{1+PC} & \frac{1}{1+PC} \end{bmatrix} \begin{bmatrix} R \\ D \end{bmatrix}.$$

The output is given by

$$Y = PU = \frac{PC}{1+PC}R + \frac{P}{1+PC}D.$$

We just derived the following closed-loop transfer functions:

$$\begin{array}{lll} R \text{ to } E : \frac{1}{1+PC}, & R \text{ to } U : \frac{C}{1+PC}, & R \text{ to } Y : \frac{PC}{1+PC}, \\ D \text{ to } E : \frac{-P}{1+PC}, & D \text{ to } U : \frac{1}{1+PC}, & D \text{ to } Y : \frac{P}{1+PC}. \end{array}$$

**Definition 5.2.4.** The feedback system in Figure 5.6 is **input-output stable** provided  $e$ ,  $u$ , and  $y$  are bounded signals whenever  $r$  and  $d$  are bounded signals; briefly, the system from  $(r, d)$  to  $(e, u, y)$  is BIBO stable.

Input-output stability is equivalent to saying that the 6 transfer functions from  $(r, d)$  to  $(e, u, y)$  are BIBO stable, in the sense that all poles are in  $\text{Re}(s) < 0$ . Since whenever  $r$  and  $e$  are bounded, so is  $y = r - e$ , it suffices to look at the 4 transfer functions from  $(r, d)$  to  $(e, u)$ , namely,

$$\begin{bmatrix} E \\ U \end{bmatrix} = \begin{bmatrix} \frac{1}{1+PC} & \frac{-P}{1+PC} \\ \frac{C}{1+PC} & \frac{1}{1+PC} \end{bmatrix} \begin{bmatrix} R \\ D \end{bmatrix}.$$

**Remark 5.2.5.** An alternative definition for input-output stability of an interconnected system is: a feedback system is input-output stable if and only if *every* possible transfer function in the system (imagine introducing noise everywhere) is BIBO stable. ♦

**Example 5.2.6.**

$$P(s) = \frac{1}{s^2 - 1}, \quad C(s) = \frac{s - 1}{s + 1}$$

The four transfer functions are

$$\begin{bmatrix} E \\ U \end{bmatrix} = \begin{bmatrix} \frac{(s+1)^2}{s^2 + 2s + 2} & \frac{s+1}{(s-1)(s^2 + 2s + 2)} \\ \frac{(s+1)(s-1)}{s^2 + 2s + 2} & \frac{(s+1)^2}{s^2 + 2s + 2} \end{bmatrix} \begin{bmatrix} R \\ D \end{bmatrix}.$$

Three of these are stable; the one from  $D$  to  $E$  is not. Consequently, the feedback system is not input-output stable. This is in spite of the fact that a bounded  $r$  produces a bounded  $y$ . The problem here is that  $C$  cancels an unstable pole of  $P$ . That isn't allowed.  $\blacktriangle$

**Example 5.2.7.**

$$P(s) = \frac{1}{s-1}, \quad C(s) = K.$$

The feedback system is input-output stable if and only if  $K > 1$  (check). This again shows that a feedback system can be stable even if the individual components are unstable.  $\blacktriangle$

We now provide two ways to test input-output stability. Write

$$P = \frac{N_p}{D_p}, \quad C = \frac{N_c}{D_c}.$$

We assume the polynomials  $(N_p, D_p)$  are coprime, i.e., have no common factors, and  $(N_c, D_c)$  are coprime too.

**Definition 5.2.6.** The **characteristic polynomial** of the feedback system in Figure 5.6 is  $\pi(s) := N_p N_c + D_p D_c$ .

The characteristic polynomial of a feedback system is the least common multiple of the denominators in the four transfer functions from  $(r, d)$  to  $(e, u)$ , i.e.,

$$\begin{bmatrix} \frac{1}{1+PC} & \frac{-P}{1+PC} \\ \frac{C}{1+PC} & \frac{1}{1+PC} \end{bmatrix} = \frac{1}{N_p N_c + D_p D_c} \begin{bmatrix} D_p D_c & -N_p D_c \\ D_p N_c & D_p D_c \end{bmatrix}. \quad (5.1)$$

**Example 5.2.8.**

$$P(s) = \frac{1}{s^2 - 1}, \quad C(s) = \frac{s - 1}{s + 1}.$$

We have seen that with the plant and controller combination the system is not feedback stable because the transfer function from  $d$  to  $e$  is unstable. Notice the unstable pole-zero cancellation. The characteristic polynomial is

$$\pi(s) = s - 1 + (s^2 - 1)(s + 1) = (s - 1)(s^2 + 2s + 2).$$

This has a right half-plane root – precisely the root that the controller cancels.  $\blacktriangle$

**Theorem 5.2.7.** *The feedback system is input-output stable if and only if the characteristic polynomial has no roots with  $\text{Re}(s) \geq 0$ .*

*Proof.* ( $\Leftarrow$  Sufficiency) If  $N_p N_c + D_p D_c$  has no roots in  $\text{Re}(s) \geq 0$ , then the four transfer functions on the left-hand side of (5.1) have no poles in  $\text{Re}(s) \geq 0$ , and hence they are stable.

( $\Rightarrow$  Necessity) Conversely, assume the feedback system is stable, that is, the four transfer functions on the left-hand side of (5.1) are BIBO stable. To conclude that  $N_p N_c + D_p D_c$  has no roots in  $\text{Re}(s) \geq 0$ , we must show that the polynomial  $N_p N_c + D_p D_c$  does not have a common factor with all four numerators in (5.1), namely,  $D_p D_c$ ,  $N_p D_c$ ,  $N_c D_p$ . That is, we must show that the four polynomials

$$N_p N_c + D_p D_c, D_p D_c, N_p D_c, N_c D_p$$

are coprime. This part is left for you.  $\square$

**Exercise 5.2.** Show that the polynomials  $N_p N_c + D_p D_c$ ,  $D_p D_c$ ,  $N_p D_c$ ,  $N_c D_p$  do not have a common root. Hint: use the fact that the pair  $(N_p, D_p)$  is coprime and the pair  $(N_c, D_c)$  is coprime.

**Definition 5.2.8.** The plant  $P(s)$  and controller  $C(s)$  have a **pole-zero cancellation** at  $s = \lambda \in \mathbb{C}$  if

$$\begin{aligned} N_p(\lambda) = D_c(\lambda) &= 0 && (\text{controller pole cancels a plant zero}) \\ D_p(\lambda) = N_c(\lambda) &= 0 && (\text{controller zero cancels a plant pole}). \end{aligned}$$

It is called an **unstable pole-zero cancellation** if  $\text{Re}(\lambda) \geq 0$ .

**Corollary 5.2.9.** If there is an unstable pole-zero cancellation, then the feedback system is not input-output stable.

*Proof.* If there is an unstable pole-zero cancellation at  $\lambda \in \overline{\mathbb{C}}^+$ , then

$$\pi(\lambda) = N_p(\lambda)N_c(\lambda) + D_p(\lambda)D_c(\lambda) = 0 + 0.$$

Thus  $\pi$  has a root in  $\overline{\mathbb{C}}^+$  so that by Theorem 5.2.7 the system is not input-output stable.  $\square$

**Remark 5.2.10.** In practice, it is not possible to exactly cancel a plant zero or pole because of modelling errors. In Example 5.2.6 therefore the transfer function from  $r$  to  $y$  will also be unstable. However, it is important to stress that even in the ideal case with a perfect pole-zero cancellation, we still get an unstable system.  $\blacklozenge$

A second way to test feedback stability is as follows.

**Theorem 5.2.11.** The feedback system is input-output stable if, and only if,

1. The transfer function  $1 + PC$  has no zeros in  $\text{Re}(s) \geq 0$ , and
2. the product  $PC$  has no unstable pole-zero cancellations.

**Exercise 5.3.** Prove Theorem 5.2.11.

**Example 5.2.9.**

$$P(s) = \frac{1}{s^2 - 1}, \quad C(s) = \frac{s - 1}{s + 1}$$

Check that 1) holds but 2) does not.  $\blacktriangle$

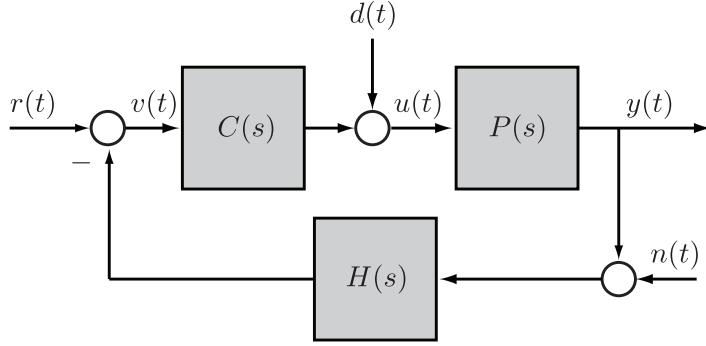


Figure 5.11: Non-unity feedback system.

**Remark 5.2.12.** Occasionally we will encounter non-unity feedback system as shown in Figure 5.11. This is the case when we model the dynamics of the sensor that provides the feedback using a transfer function  $H(s)$ . The exogenous signal  $n$  models sensor noise. If we write

$$P = \frac{N_p}{D_p}, \quad C = \frac{N_c}{D_c}, \quad H = \frac{N_h}{D_h}$$

then you can check that the characteristic polynomial becomes

$$\pi(s) = N_c N_p N_h + D_c D_p D_h.$$

In this case we have feedback stability if and only if all the roots of  $\pi(s)$  are in  $\mathbb{C}^-$ . ♦

**Exercise 5.4.** Find the 9 transfer functions from  $(r, d, n)$  to  $(v, u, y)$ .

### 5.2.3 Internal stability and input-output stability

We have given two definitions for what it means for the system in Figure 5.6 to be stable. Internal stability (Definition 5.2.3) dealt with the closed-loop state model whereas input-output stability (Definition 5.2.4) dealt with the various transfer functions in the loop. It can be shown that the roots of the characteristic polynomial  $\pi(s)$  are a subset of the eigenvalues of  $A_{cl}$ , i.e., the roots of  $\pi(s)$  are a subset of the roots of  $\det(sI - A_{cl})$ . Hence

$$\text{internal stability} \implies \text{input-output stability}.$$

Usually the roots of  $\pi(s)$  are identical to the eigenvalues of  $A_{cl}$ . In these cases, the two stability concepts are equivalent. More precisely, if the pair  $(A_p, B_p)$  is **stabilizable** and the pair  $(A_p, C_p)$  is **detectable** then internal stability is equivalent to input-output stability [Dullerud and Paganini, 2000]. The notions of stabilizability and detectability are beyond the scope of this course and are covered in ECE488.

In this course internal stability and input-output stability of a feedback system are assumed to be equivalent and hence when we talk about feedback stability there is no ambiguity. This is reasonable for most practical applications.

## 5.3 The Routh-Hurwitz criterion

In practice, one checks input-output stability using numerical algorithms to compute the roots of a polynomial. However, if some of the coefficients of the polynomial are not fixed, e.g., a controller gain, then we can't solve for the roots numerically. The Routh-Hurwitz criterion provides a test to check if the roots of  $\pi(s)$  are in  $\mathbb{C}^-$  without actually finding them.

Consider an  $n$ th order polynomial with real coefficients  $\pi \in \mathbb{R}[s]$ , i.e.,

$$\pi(s) = s^n + a_{n-1}s^{n-1} + \cdots + a_1s + a_0, \quad a_i \in \mathbb{R}, \quad n \neq 0. \quad (5.2)$$

**Definition 5.3.1.** A polynomial  $\pi \in \mathbb{R}[s]$  is **Hurwitz** if all its roots lie in  $\mathbb{C}^-$ .

The Routh-Hurwitz criterion is an algebraic test for a polynomial  $\pi(s)$  to be Hurwitz without having to calculate its roots [Routh, 1877].

### 5.3.1 A necessary condition for a polynomial to be Hurwitz

We make the observation that a necessary, but not sufficient, condition for  $\pi(s)$  to be Hurwitz is that all the coefficients  $a_i$  have the same sign.

*Proof.* Assume that  $\pi \in \mathbb{R}[s]$  is Hurwitz. Let  $\{\lambda_1, \dots, \lambda_r\}$  be the real roots of  $\pi$  and let  $\{\mu_1, \bar{\mu}_1, \dots, \mu_s, \bar{\mu}_s\}$  be the complex conjugate roots of  $\pi$ . Then  $n = r + 2s$ . Factor the polynomial as

$$\pi(s) = (s - \lambda_1) \cdots (s - \lambda_r)(s - \mu_1)(s - \bar{\mu}_1) \cdots (s - \mu_s)(s - \bar{\mu}_s).$$

For the real roots  $-\lambda_i > 0$  since  $\pi$  is Hurwitz by assumption. For the complex conjugate roots we expand

$$(s - \mu_i)(s - \bar{\mu}_i) = s^2 - (\mu_i + \bar{\mu}_i)s + \mu_i \bar{\mu}_i = s^2 - 2 \operatorname{Re}(\mu_i)s + |\mu_i|^2.$$

Note that  $\mu_i \bar{\mu}_i = |\mu_i|^2 > 0$  and  $-(\mu_i + \bar{\mu}_i) = -2 \operatorname{Re}(\mu_i) > 0$  since  $\pi$  is Hurwitz by assumption. Therefore, when we expand  $\pi(s)$ , every coefficient will be have the same sign as the coefficient multiplying  $s^n$ .  $\square$

#### Example 5.3.1.

$$s^4 + 3s^3 - 2s^2 + 5s + 6 \quad (\text{a bad root})$$

$$s^3 + 4s + 6 \quad (\text{a bad root})$$

$$s^3 + 5s^2 + 9s + 1 \quad (\text{don't know}).$$

### 5.3.2 Routh's algorithm

One of the most popular algorithms to determine whether or not a polynomial is Hurwitz is Routh's algorithm. We present it here without proof and refer the interested student to [Qiu and Zhou, 2010, Section 3.2] for a proof. For the method of Routh, we construct an array involving the coefficients of the polynomial in question. The array is constructed inductively, starting with the first two rows.

We again consider a polynomial  $\pi \in \mathbb{R}[s]$  of degree  $n$  like (5.2). We construct Table 5.1. In Table 5.1 the first two rows are populated using the coefficients of  $\pi$ . The third row is computed from the first two

$$r_{2,0} = \frac{a_{n-1}a_{n-2} - a_{n-3}}{a_{n-1}}, \quad r_{2,1} = \frac{a_{n-1}a_{n-4} - a_{n-5}}{a_{n-1}}, \quad r_{2,2} = \frac{a_{n-1}a_{n-6} - a_{n-7}}{a_{n-1}}, \dots$$

The remaining rows are computed recursively using the same pattern

$$r_{i,j} = \frac{r_{i-1,0}r_{i-2,j+1} - r_{i-2,0}r_{i-1,j+1}}{r_{i-1,0}}, \quad i \in \{2, \dots, n\}, \quad j \in \left\{0, \dots, \left\lfloor \frac{n-i}{2} \right\rfloor\right\}. \quad (5.3)$$

The construction terminates if we end up with a zero in the first column. There are at most  $n + 1$  rows in the final table.

Table 5.1: Routh array.

$s^n$	$r_{0,0} = 1$	$r_{0,1} = a_{n-2}$	$r_{0,2} = a_{n-4}$	$r_{0,3} = a_{n-6}$	$\dots$
$s^{n-1}$	$r_{1,0} = a_{n-1}$	$r_{1,1} = a_{n-3}$	$r_{1,2} = a_{n-5}$	$r_{1,3} = a_{n-7}$	$\dots$
$s^{n-2}$	$r_{2,0}$	$r_{2,1}$	$r_{2,2}$	$r_{2,3}$	$\dots$
$s^{n-3}$	$r_{3,0}$	$r_{3,1}$	$r_{3,2}$	$r_{3,3}$	$\dots$
$\vdots$	$\vdots$	$\vdots$	$\vdots$		
$s^2$	$r_{n-2,0}$	$r_{n-2,1}$	0		$\dots$
$s^1$	$r_{n-1,0}$	0	0		$\dots$
$s^0$	$r_{n,0}$	0	0		$\dots$

**Remark 5.3.2.** The definition of the coefficients (5.3) can be expressed using determinants

$$r_{i,j} = -\frac{1}{r_{i-1,0}} \det \begin{bmatrix} r_{i-2,0} & r_{i-2,j+1} \\ r_{i-1,0} & r_{i-1,j+1} \end{bmatrix}, \quad i \in \{2, \dots, n\}, \quad j \in \left\{0, \dots, \left\lfloor \frac{n-i}{2} \right\rfloor\right\}.$$

◆

**Theorem 5.3.3** (Routh-Hurwitz Stability Criterion). Consider a polynomial  $\pi \in \mathbb{R}[s]$  given by (5.2) and its associated Routh array in Table 5.1. The number of roots of  $\pi$  with real part greater than or equal to zero equals the number of sign changes in the first column of the array.

**Example 5.3.2.**

$$\pi(s) = a_2 s^2 + a_1 s + a_0, \quad a_2 \neq 0.$$

$s^2$	$a_2$	$a_0$	0
$s^1$	$a_1$	0	0
$s^0$	$\frac{a_1 a_0}{a_2} = a_0$	0	0

We conclude that all the roots of  $\pi$  are in  $\mathbb{C}^-$  if and only if  $a_0, a_1, a_2$  have the same sign. ▲

**Example 5.3.3.**

$$\pi(s) = 2s^4 + s^3 + 3s^2 + 5s + 10.$$

$s^4$	2	3	10	0
$s^3$	1	5	0	0
$s^2$	$\frac{3-10}{1} = -7$	10	0	0
$s^1$	$\frac{-35-10}{-7} = \frac{45}{7}$	0	0	0
$s^0$	10	0	0	0

There are two sign changes in the first column so  $\pi(s)$  has two roots in  $\mathbb{C}^+$ . ▲

▲

The next example illustrates the utility of the Routh-Hurwitz criterion.

**Example 5.3.4.** Consider the unity feedback system in Figure 5.12 with

$$P(s) = \frac{1}{s^4 + 6s^3 + 11s^2 + 6s}, \quad C(s) = K_p.$$

Find all  $K_p \in \mathbb{R}$  such that the feedback system is stable. From Section 5.2.2 we know that the system is feedback stable if and only if all the roots of  $\pi(s)$  are in  $\mathbb{C}^-$ . In this case

$$\pi(s) = N_p N_c + D_p D_c = s^4 + 6s^3 + 11s^2 + 6s + K_p.$$

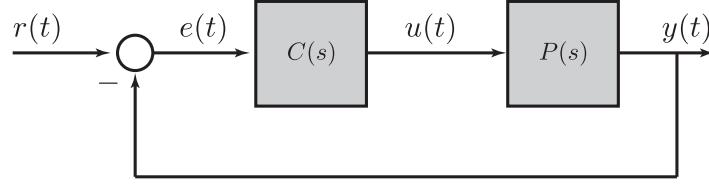


Figure 5.12: System for Example 5.3.4.

We now apply the Routh-Hurwitz stability criterion

$s^4$	1	11	$K_p$	0
$s^3$	6	6	0	0
$s^2$	10	$K_p$	0	0
$s^1$	$\frac{3}{5}(10 - K_p)$	0	0	0
$s^0$	$K_p$	0	0	0

We need all entries in the first column to be positive. From the  $s^1$  row we get the constraint  $K_p < 10$ . From the  $s^0$  row we get the constraint  $K_p > 0$ . Therefore the system is feedback stable provided  $K_p \in (0, 10)$ .  $\blacktriangle$

### Special cases

In using Routh's array there are sometimes special cases that require extra steps to be taken. For example, we see that in building the array, we cannot proceed when one of the elements in the first column is zero. In this case we can immediately conclude that the polynomial is unstable.

**Example 5.3.5.** Consider  $\pi(s) = s^5 + 3s^4 + 2s^3 + 6s^2 + 3s + 3$ . We then have

$s^5$	1	2	3	
$s^4$	3	6	3	
$s^3$	0	2		
$s^2$	$r_{3,0}$			
$s^1$				
$s^0$				

Since  $r_{2,0} = 0$ , the next element  $r_{3,0}$  cannot be computed. We can stop at this point and conclude that  $\pi(s)$  is unstable since the elements in the first column aren't all positive.  $\blacktriangle$

The second special case we consider is when an entire row of Routh's array is zero. If row  $s^{n-k}$  is all zero it means that the polynomial  $\pi_a(s) = r_{k-1,0}s^{n-k+1} + r_{k-1,1}s^{n-k-1} + r_{k-1,2}s^{n-k-3} + r_{k-1,3}s^{n-k-5} + \dots$  divides the original polynomial  $\pi$ . The polynomial  $\pi_a$  is sometimes called an **auxiliary polynomial**. It consists of only even or odd powers of  $s$ , where the coefficients come from the row above the row of zeros. By the result from Section 5.3.1 we conclude that  $\pi_a$  is not Hurwitz and therefore neither is  $\pi$ .

**Example 5.3.6.** Consider  $\pi(s) = s^6 + 5s^5 + 2s^4 + 5s^3 + 4s^2 + 15s + 3$ . The associated Routh table is

$s^6$	1	2	4	3	
$s^5$	5	5	15	0	
$s^4$	1	1	3		
$s^3$	0	0			
$s^2$					
$s^1$					
$s^0$					

Thus  $\pi_a(s) = s^4 + s^2 + 3$  and  $\pi(s) = \pi_a(s)(s^2 + 5s + 1)$ . The roots of  $\pi_a(s)$  are  $s = \pm 0.785 \pm j1.06$ .  $\blacktriangle$

## 5.4 Steady-state performance

For any control system, stability is mandatory; good performance is desirable. Two types of performance measures are:

1. Transient performance: depends in a complicated way on the location of the closed-loop poles and zeros. If the closed-loop system is dominated by a first or second order system, then we can use the results of Chapter 4.
2. Steady-state performance: This refers to specifications like tracking reference signals and rejecting disturbance signals.

In this section we discuss steady-state performance.

### 5.4.1 Tracking reference signals

Cruise control in a car regulates the speed to a prescribed set point. What is the principle underlying its operation? The answer lies in the final-value theorem (Theorem 3.6.1).

**Example 5.4.1.** Consider the unity feedback system in Figure 5.13 with

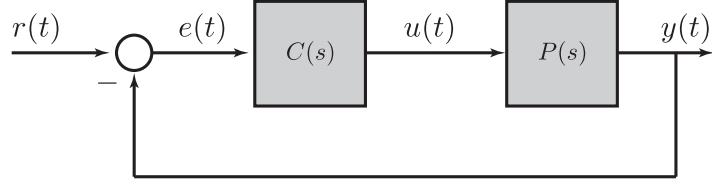


Figure 5.13: System for Example 5.4.1.

$$P(s) = \frac{1}{s+1}, \quad C(s) = \frac{1}{s}.$$

Let  $r$  be a step input,  $r(t) = r_0 \mathbf{1}(t)$ . Then we have

$$\begin{aligned} E(s) &= \frac{1}{1 + P(s)C(s)} R(s) \\ &= \frac{s(s+1)}{s^2 + s + 1} \frac{r_0}{s} \\ &= \frac{s+1}{s^2 + s + 1} r_0. \end{aligned}$$

The FVT applies to  $E(s)$ , and  $e(t) \rightarrow 0$  as  $t \rightarrow \infty$ . Thus the feedback system provides asymptotic tracking of step reference signals with zero steady-state error.

How it works:  $C(s)$  contains an internal model of  $R(s)$  (i.e., an integrator); closing the loop creates a zero from  $R(s)$  to  $E(s)$  to exactly cancel the unstable pole of  $R(s)$ . (This isn't an illegal pole-zero cancellation.)  $\blacktriangleleft$

**Remark 5.4.1.** Tracking error is always defined by  $e := r - y$  where  $r$  is the command or reference input and  $y$  is the plant output. In a unity feedback control system, as in Figure 5.6, the tracking error is the input to the controller. If the sensor TF in the feedback path is not 1, as in Figure 5.11, then the signal coming out of the negative feedback summing junction is *not* the tracking error.  $\blacklozenge$

## Step inputs

Let's generalize the preceding example. Consider the unity feedback control system in Figure 5.6 and write

$$\frac{E(s)}{R(s)} = \frac{1}{1 + CP} = \frac{D_p D_c}{N_p N_c + D_p D_c}. \quad (5.4)$$

Suppose that  $R(s) = r_0/s$ , i.e.,  $r(t) = r_0 \mathbf{1}(t)$  for some constant  $r_0 \in \mathbb{R}$ . If we *assume* that the controller  $C(s)$  stabilizes the closed-loop system, then  $\pi(s) = N_p N_c + D_p D_c$  has all its roots in  $\mathbb{C}^-$ , the TF (5.4) is BIBO stable, and the final-value theorem applies. Application of the FVT to the tracking error yields

$$e_{ss} = \lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} sE(s) = \lim_{s \rightarrow 0} s \frac{1}{1 + C(s)P(s)} R(s) = \lim_{s \rightarrow 0} \frac{s}{1 + C(s)P(s)} \frac{r_0}{s} = \lim_{s \rightarrow 0} \frac{r_0}{1 + C(s)P(s)}.$$

Therefore  $e_{ss} = 0$  if and only if  $P(0)C(0) = \infty$ . This means that  $P(s)C(s)$  has at least one pole at  $s = 0$ , i.e., at least one integrator.

If  $P(s)$  does not have a pole at  $s = 0$  and we want perfect step tracking, then we must choose

$$C(s) = \frac{1}{s} C_1(s)$$

so that  $C(s)P(s)$  has an integrator. Then  $C_1(s)$  is designed to provide feedback stability. The integrator adds more phase lag though and this can make the overall system harder to stabilize.

## Ramp inputs

Now let's try a non-constant reference input.

**Example 5.4.2.** This time take

$$P(s) = \frac{2s+1}{s(s+1)}, \quad C(s) = \frac{1}{s}$$

and take  $r$  to be a ramp,  $r(t) = r_0 t$ ,  $t \geq 0$  for some constant  $r_0 \in \mathbb{R}$ . Then  $R(s) = r_0/s^2$  and so (verify!)

$$E(s) = \frac{s+1}{s^3 + s^2 + 2s + 1} r_0.$$

You can check using, say, the Routh-Hurwitz stability criterion, that the poles of  $E(s)$  are in  $\text{Re}(s) < 0$  which means that the FVT can be applied to get  $\lim_{t \rightarrow \infty} e(t)$ . Again  $e(t) \rightarrow 0$ ; perfect tracking of a ramp. Here  $C(s)$  and  $P(s)$  together provide the internal model, a double integrator.  $\blacktriangleleft$

Generalizing the preceding example, suppose that  $R(s) = r_0/s^2$  (ramp reference  $r(t) = r_0 t$ ,  $t \geq 0$ ,  $r_0 \in \mathbb{R}$ ). If we *assume* that the controller  $C(s)$  provides input-output stability then  $\pi(s)$  has all its roots in  $\mathbb{C}^-$  and the TF (5.4) is BIBO stable. Let's try and apply the final-value theorem

$$e_{ss} = \lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} sE(s) = \lim_{s \rightarrow 0} \frac{s}{1 + C(s)P(s)} \frac{r_0}{s^2} = \left. \frac{r_0}{sC(s)P(s)} \right|_{s=0}.$$

We observe:

- (i) If  $C(s)P(s)$  has no poles at  $s = 0$  then we can't even apply the FVT since  $sE(s)$  has bad poles. It turns out that  $e_{ss}$  is unbounded in this case. This doesn't contradict the assumption that  $C(s)$  provides input-output stability because the input is unbounded.
- (ii) If  $C(s)P(s)$  has one pole at  $s = 0$  then  $e_{ss} = \frac{r_0}{C(0)P(0)}$  is finite and non-zero.
- (iii) If  $C(s)P(s)$  has two or more poles at  $s = 0$  then  $e_{ss}$  equals zero.

### Internal model principle

It is possible to generalize the above discussion to arbitrary reference signals.

**Theorem 5.4.2** (Internal model principle). *Assume that  $P(s)$  is strictly proper,  $C(s)$  is proper and the feedback system is stable. If  $C(s)P(s)$  contains an internal model of the unstable part of  $R(s)$ , then perfect asymptotic tracking occurs.*

The internal model principle of Theorem 5.4.2 can be understood as follows. Suppose that the Laplace transform of the reference signal  $r(t)$  is  $R(s) = \mathcal{L}\{r(t)\}$ . Write

$$R(s) = \frac{N_r(s)}{D_r(s)} = \frac{N_r(s)}{D_r^+(s)D_r^-(s)}$$

where the roots of the polynomial  $D_r^-(s)$  are in  $\mathbb{C}^-$  ( $\text{Re}(s) < 0$ ) while the roots of the polynomial  $D_r^+(s)$  are in  $\overline{\mathbb{C}}^+$  ( $\text{Re}(s) \geq 0$ ). The internal model principle says that in order for the plant output  $y(t)$  to asymptotically converge to  $r(t)$ , the product  $C(s)P(s)$  should have the form

$$C(s)P(s) = \frac{N(s)}{D(s)D_r^+(s)}$$

where  $N(s), D(s) \in \mathbb{R}[s]$  are polynomials. That is, the product  $C(s)P(s)$  must have a copy of  $D_r^+(s)$  in its denominator.

**Example 5.4.3.** Let

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

We want the closed-loop system to be stable and to track the reference signal

$$r(t) = r_0 \sin(t).$$

Then

$$R(s) = \frac{r_0}{s^2 + 1}.$$

In this case  $N_r(s) = r_0$  and  $D_r^+(s) = s^2 + 1$ ,  $D_r^-(s) = 1$ . Since the plant doesn't contain a copy of  $D_r^+(s)$ , the internal model principle suggests that we should pick a controller of the form

$$C(s) = \frac{1}{D_r^+(s)} C_1(s) = \frac{1}{s^2 + 1} C_1(s)$$

where  $C_1(s) \in \mathbb{R}(s)$  is a TF yet to be determined. That is, we embed an internal model of the unstable part of  $R(s)$  in  $C(s)$  and allow an extra factor  $C_1(s)$  to achieve feedback stability. You can check that  $C_1(s) = s$  works. ▲

**Remark 5.4.3.** The internal model principle of Theorem 5.4.2 is only concerned with the unstable part of the Laplace transform  $R(s)$  of the reference signal  $r(t)$ . For example, if  $r(t) = e^{-t} + \sin(t)$  then

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

In this case  $D_r^-(s) = s+1$  and  $D_r^+(s) = s^2 + 1$ . The internal model needs to include the poles at  $s = \pm j$  (roots of  $D_r^+(s)$ ) but not the pole at  $s = -1$ . This is because the part of  $r(t)$  corresponding to stable pole decays to zero in steady-state and hence there is nothing to asymptotically track. ◆

To help understanding, it is useful to interpret the internal model principle in the frequency domain.

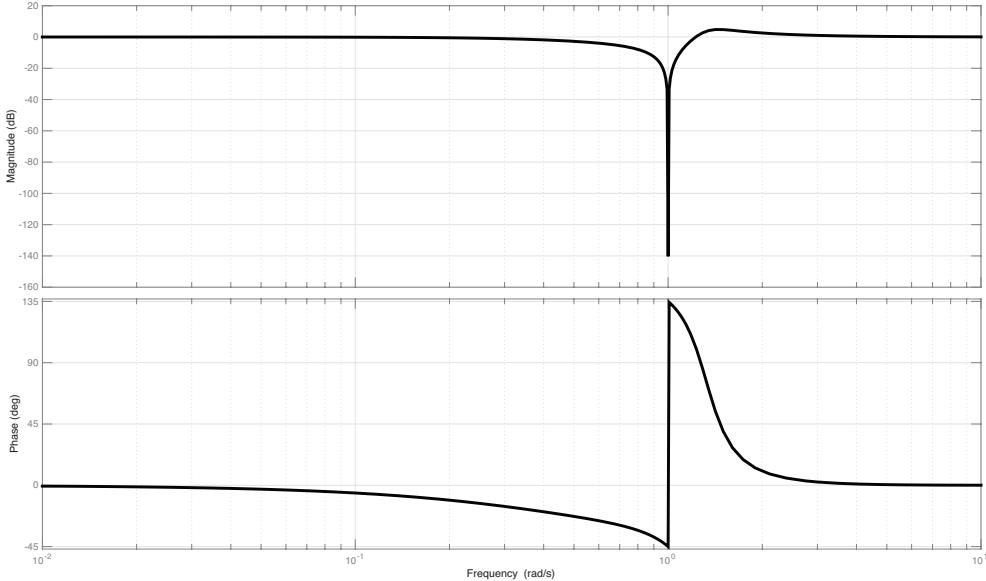


Figure 5.14: Bode plot of  $E/R$  from Example 5.4.3.

**Example 5.4.4.** Returning to Example 5.4.3 we have that the TF from  $r$  to  $e$  is

$$\frac{E(s)}{R(s)} = \frac{1}{1 + C(s)P(s)} = \frac{s^3 + s^2 + s + 1}{s^3 + s^2 + 2s + 1} =: S(s).$$

Figure 5.14 shows the Bode plot of the frequency response  $S(j\omega)$ . Notice that the system heavily attenuates signals at the frequency  $10^0 = 1$  rad/s. Indeed, the internal model introduces zeros at  $s = \pm j$  in  $S$  which creates a notch filter. Any reference signals at this frequency result in zero steady-state error as we saw using the FVT in the previous example. ▲

**Exercise 5.5.** At what frequencies do you expect the Bode plot of  $E/R$  from Example 5.4.1 to heavily attenuate signals? Confirm your response by plotting the Bode plot.

In summary, we have the following facts:

- If the closed-loop system is unstable, we can't even talk about steady-state tracking performance since there will likely not be a steady-state response.
- Even if the closed-loop system is stable, we have to be careful applying the FVT to compute the steady-state tracking error because, depending on the reference signal,  $sE(s)$  may have poles in  $\text{Re}(s) \geq 0$ .
- If the closed-loop system is stable and  $P(s)C(s)$  contains an internal model of the unstable part of  $R(s)$ , then perfect asymptotic tracking occurs.
- It does not matter if  $P(s)$  or  $C(s)$  provide the internal model. Perfect asymptotic tracking occurs when the product  $C(s)P(s)$  has an internal model of the unstable part of  $R(s)$ .
- If  $C(s)P(s)$  does not contain an internal model, then increasing the gain  $|C(j\omega)P(j\omega)|$  over the range of frequencies that  $r(t)$  contains decreases the steady-state tracking error. This is consistent with the idea that high-gain leads to good performance.

## 5.4.2 Steady-state disturbance rejection

A disturbance is an exogenous input over which we have no control, e.g., turbulence acting on a plane, load torque on a car due to hills. Also signal noise, which in practice every signal in a control system has, can be

modelled as a disturbance. Consider a system with an input disturbance  $d_i(t)$ , an output disturbance  $d_o(t)$  and sensor noise  $n(t)$  illustrated in Figure 5.15. We can once again use the final-value theorem to check for the

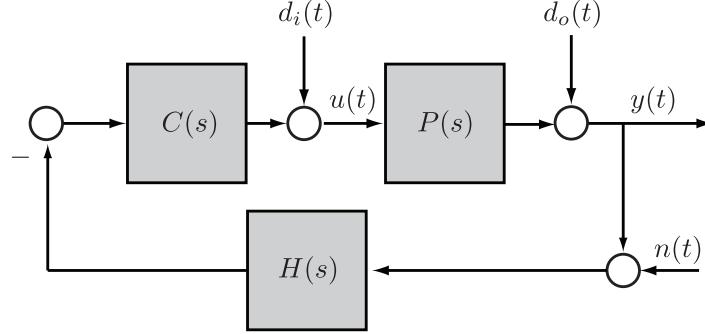


Figure 5.15: Feedback system with various disturbances and noise.

steady-state effect of certain types of disturbances and noise.

**Example 5.4.5.** Suppose that  $d_i = n = 0$  in Figure 5.15 and that  $d_o(t) = \mathbf{1}(t)$ . Find conditions on  $P(s)$  and  $C(s)$  so that the steady-state effect of this disturbance on the output  $y$  is zero. The transfer function from  $d_o$  to  $y$  is

$$\frac{Y(s)}{D_o(s)} = \frac{1}{1 + C(s)P(s)H(s)}.$$

If we have feedback stability, then the above TF is stable and the steady-state effect of  $d_o$  is

$$\lim_{t \rightarrow \infty} y(t) = \lim_{s \rightarrow 0} sY(s) = \frac{1}{1 + C(0)P(0)H(0)}.$$

Therefore the disturbance will have no steady-state effect on the output if and only if  $C(0)P(0)H(0) = \infty$ , i.e.,  $C(s)P(s)H(s)$  has at least one pole at the origin. We conclude that the output disturbance  $d_o(t) = \mathbf{1}(t)$  has no effect on the steady-state output of this system as long as (i) the system is feedback stable and (ii)  $C(s)P(s)H(s)$  has at least one pole at the origin.  $\blacktriangle$

As seen in Figure 5.15, disturbances can appear at the input to a plant and/or the output to a plant. In this section we adopt a unified description that encompasses both types of disturbances. Specifically, for a nominal plant model  $P(s)$  with input  $U(s)$  and output  $Y(s)$ , we'll assume that a disturbance  $D(s)$  acts on the plant at some intermediate point, i.e., we model the plant as follows:

$$Y(s) = P_2(s)(D(s) + P_1(s)U(s)) = P_2(s)D(s) + P_2(s)P_1(s)U(s). \quad (5.5)$$

where  $P(s) = P_2(s)P_1(s)$ . The generalized model of plant disturbances in a feedback loop are shown in Figure 5.16. If  $P_1(s) = 1$ , then  $d(t)$  is an input disturbance. If  $P_2(s) = 1$ , then  $d(t)$  is an output disturbance. Note that Figure 5.16 does not include measurement noise, i.e., it doesn't include the signal  $n(t)$  from Figure 5.15.

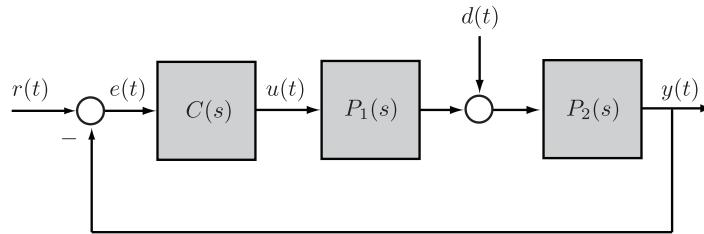


Figure 5.16: Feedback system with generalized model of plant disturbances.

**Example 5.4.6.** The generalized disturbance model allows us to handle more than just input and output disturbances. Consider a plant with state-space model

$$\begin{aligned}\dot{x}(t) &= \begin{bmatrix} 0 & 1 \\ 0 & 0 \end{bmatrix} x(t) + \begin{bmatrix} 0 \\ 1 \end{bmatrix} u(t) + \begin{bmatrix} 0.3 \\ 0 \end{bmatrix} d(t) \\ y(t) &= [1 \ 0] x(t).\end{aligned}$$

Here  $u$  is the plant input while  $d$  is a disturbance. Find the transfer functions  $P_1(s)$  and  $P_2(s)$  so that the input-output model of this plant has the structure (5.5).

The TF from  $U(s)$  to  $Y(s)$  is

$$\frac{Y(s)}{U(s)} = [1 \ 0] \begin{bmatrix} s & -1 \\ 0 & s \end{bmatrix}^{-1} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{s^2} [1 \ 0] \begin{bmatrix} s & 1 \\ 0 & s \end{bmatrix} \begin{bmatrix} 0 \\ 1 \end{bmatrix} = \frac{1}{s^2}.$$

The TF from  $D(s)$  to  $Y(s)$  is

$$\frac{Y(s)}{D(s)} = [1 \ 0] \begin{bmatrix} s & -1 \\ 0 & s \end{bmatrix}^{-1} \begin{bmatrix} 0.3 \\ 0 \end{bmatrix} = \frac{1}{s^2} [1 \ 0] \begin{bmatrix} s & 1 \\ 0 & s \end{bmatrix} \begin{bmatrix} 0.3 \\ 0 \end{bmatrix} = \frac{0.3}{s}.$$

Therefore

$$P_1(s) = \frac{10}{3} \frac{1}{s}, \quad P_2(s) = \frac{0.3}{s}.$$

▲

We are interested in understanding when a closed-loop system is able to asymptotically reject disturbances in steady-state. Assume throughout that the controller  $C(s)$  provides input-output stability. The TF from  $D(s)$  to  $Y(s)$  for the system in Figure 5.16 is

$$\frac{Y(s)}{D(s)} = \frac{P_2(s)}{1 + C(s)P_2(s)P_1(s)}.$$

Suppose that  $D(s) = \mathcal{L}\{d(t)\}$  is the Laplace transform of the disturbance signal and write

$$D(s) = \frac{N_d(s)}{D_d(s)} = \frac{N_d(s)}{D_d^+(s)D_d^-(s)}$$

where the roots of the polynomial  $D_d^-(s)$  are in  $\mathbb{C}^-$  ( $\text{Re}(s) < 0$ ) while the roots of the polynomial  $D_d^+(s)$  are in  $\overline{\mathbb{C}}^+$  ( $\text{Re}(s) \geq 0$ ). We'd like to use the FVT to determine the steady-state effect of the disturbance on the output so let's determine conditions under which  $sY(s)$  has no poles in  $\overline{\mathbb{C}}^+$ . We have

$$\begin{aligned}sY(s) &= \frac{P_2(s)}{1 + C(s)P_2(s)P_1(s)} D(s) \\ &= s \frac{P_2(s)}{1 + C(s)P_2(s)P_1(s)} \frac{N_d(s)}{D_d^+(s)D_d^-(s)}.\end{aligned}$$

Now write  $C(s) = N_c(s)/D_c(s)$ ,  $P_1(s) = N_{p_1}(s)/D_{p_1}(s)$ ,  $P_2(s) = N_{p_2}(s)/D_{p_2}(s)$  so that

$$sY(s) = s \frac{N_{p_2}(s)D_c(s)D_{p_1}(s)}{D_c(s)D_{p_1}(s)D_{p_2}(s) + N_c(s)D_{p_2}(s)D_{p_1}(s)} \frac{N_d(s)}{D_d^+(s)D_d^-(s)}.$$

Since the controller provides input-output stability, the characteristic polynomial  $\pi(s) = D_c(s)D_{p_1}(s)D_{p_2}(s) + N_c(s)D_{p_2}(s)D_{p_1}(s)$  is Hurwitz. Therefore,  $sY(s)$  has no poles with  $\text{Re}(s) \geq 0$  if the product  $N_{p_2}(s)D_c(s)D_{p_1}(s)$  contains a copy of the polynomial  $D_d^+(s)$ . In summary, we've shown that if  $C(s)$  provides input-output stability

and the product  $N_{p_2}(s)D_c(s)D_{p_1}(s)$  contains a copy of the polynomial  $D_d^+(s)$ , then the final-value theorem applies and

$$\lim_{t \rightarrow \infty} y(t) = \lim_{s \rightarrow 0} sY(s) = \lim_{s \rightarrow 0} s \frac{N_{p_2}(s)D_c(s)D_{p_1}(s)}{D_c(s)D_{p_1}(s)D_{p_2}(s) + N_c(s)D_{p_2}(s)D_{p_1}(s)} \frac{N_d(s)}{D_d^+(s)D_d^-(s)} = 0.$$

This suggests controllers of the form

$$C(s) = \frac{1}{D_d^+(s)} C_1(s)$$

where the term  $1/D_d^+(s)$  provides the internal model of the disturbance while the term  $C_1(s)$  is used to stabilize the loop.

**Remark 5.4.4.** The derivations of this section show that in order to simultaneously reject input and output disturbances, the *controller*  $C(s)$ , not the product  $C(s)P(s)$ , must contain an internal model of the disturbance.



Sometimes we can't track and reject disturbances simultaneously as the next example illustrates.

**Example 5.4.7. (Tracking steps while rejecting measurement error)** Consider the feedback system in Figure 5.17. Suppose that we want the closed-loop system to asymptotically track constant reference signals

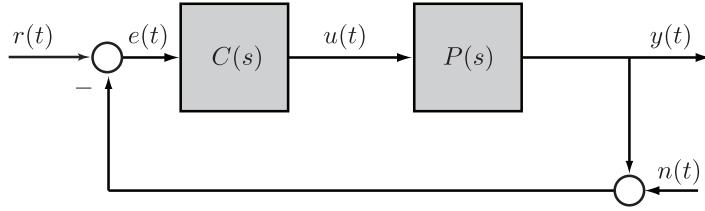


Figure 5.17: Tracking steps while rejecting bias in measurements

$r(t) = r_0 \mathbf{1}(t)$  while rejecting measurement error modelled as a constant bias  $n(t) = n_0 \mathbf{1}(t)$ . Measurement noise was not considered in our derivations of disturbance rejecting controllers so let's apply first principles to see if we can reject the error. The TF from  $N(s) = \mathcal{L}\{n(t)\}$  to  $Y(s)$  is

$$\frac{Y(s)}{N(s)} = \frac{-C(s)P(s)}{1 + C(s)P(s)}.$$

If  $N(s) = n_0/s$  and  $C(s)$  provides feedback stability, then by the FVT

$$\lim_{t \rightarrow \infty} y(t) = n_0 \frac{-C(0)P(0)}{1 + C(0)P(0)}.$$

This means that the steady-state error due to measurement error is zero only if  $C(s)P(s)$  has a zero at the origin. However, if  $C(0)P(0) = 0$ , then  $y$  can never track a reference of the form  $r_0 \mathbf{1}(t)$  because the controller can't provide an internal model of  $R(s)$  without causing an unstable pole-zero cancellation. This shows that the feedback system is not capable of rejecting measurement error of the form  $n_0 \mathbf{1}(t)$  if it is required to track signals of the same form. This makes a lot of intuitive sense. If we wish to track a step and there is a step measurement error, then the controller can never know the true value of the output and cannot take appropriate actions.

## 5.5 Summary

This chapter contains some of the most important ideas in this course. The following concepts should be understood and gleamed after reading this chapter.

1. We have to be careful when we start to interconnect LTI systems because stability of the interconnected system does not follow from stability of the individual components.
2. We gave two definitions for what it means for the system in Figure 5.6 to be stable. Internal stability (Definition 5.2.3) dealt with the closed-loop state model whereas input-output stability (Definition 5.2.4) dealt with the various transfer functions in the loop.
3. You must know how to test for both internal stability and input-output stability.
4. You should understand why internal stability implies input-output stability.
5. You should know what the characteristic polynomial is for a feedback system and how to find it for unity feedback systems and non-unity feedback systems.
6. Section 5.3 covered the Routh-Hurwitz criterion. You must be able to use these tests to check for stability.
7. You should understand the internal model principle (Theorem 5.4.2) and be able to determine the steady-state tracking error and disturbance response for a stable feedback system using the final-value theorem.

# Chapter 6

## Root-locus method

The poles of a system tell us a lot about the behaviour of the system. They determine stability (directly) and transient performance (in a complicated way). In this chapter we study the roots of polynomials when the coefficients depend linearly on a parameter. In control systems, the parameter is typically the gain of a feedback loop, and our interest is in choosing the gain so that the closed-loop system is internally stable.

The manner of studying such problems in this chapter is to understand how the roots move in the complex plane as functions of the parameter. That is to say, we look at the locus of all roots of the polynomial as the parameter varies, hence the name “root-locus.” The root-locus shows us, graphically, how the closed-loop poles of a control system move around the  $s$ -plane as a single parameter is varied. The root-locus can be very easily and quickly sketched by hand which makes it an efficient tool for analysis and, to some extent, design<sup>1</sup>.

## Contents

6.1 Basic root-locus construction . . . . .	137
6.2 Examples . . . . .	141
6.3 Non-standard problems . . . . .	146
6.4 Summary . . . . .	149

### 6.1 Basic root-locus construction

Consider a unity feedback system as shown in Figure 6.1. The characteristic polynomial of this system (see

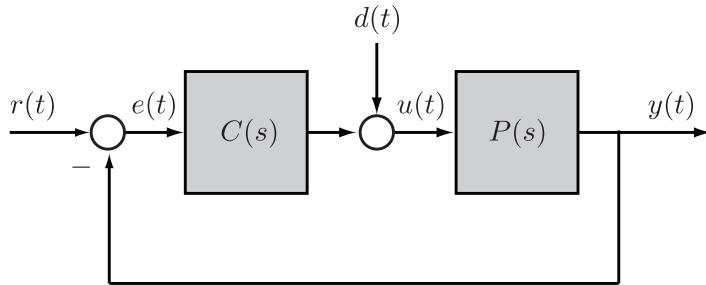


Figure 6.1: Unity feedback system.

Definition 5.2.6) is  $\pi(s) = N_p(s)N_c(s) + D_p(s)D_c(s)$ . The closed-loop system is internally stable if and only if all the roots of  $\pi$  are in  $\mathbb{C}^-$ , see Theorem 5.2.7.

<sup>1</sup>The root-locus method was put forth by Walter Evans [Evans, 1948], [Evans, 1950].

**Example 6.1.1.** To get a feeling for what a root-locus diagram looks we first consider a simple example in which we can compute the closed-loop poles explicitly. Suppose that

$$C(s) = K, \quad P(s) = \frac{1}{s(s+2)}.$$

Then  $\pi(s) = s^2 + 2s + K$ . Using the quadratic formula from high school the roots of the characteristic polynomial are  $s = -1 \pm \sqrt{1-K}$ . From this equation we make the following observations.

- If  $K \leq 0$ , the closed-loop system is unstable.
- If  $K = 0$ , the roots of  $\pi(s)$  equal the poles of  $P(s)$ .
- If  $K \in (0, 1]$ , the closed-loop system has two real poles in  $\mathbb{C}^-$  and is stable.
- If  $K \in (1, +\infty)$ , the closed-loop system has complex conjugate poles with real part  $-1$  and hence it is stable.
- As  $K \rightarrow +\infty$  the imaginary part of the poles goes to  $\pm\infty$ .

From these observations we can sketch a plot that shows how the closed-loop poles vary as  $K$  goes from 0 to  $+\infty$ . The sketch is shown in Figure 6.2.

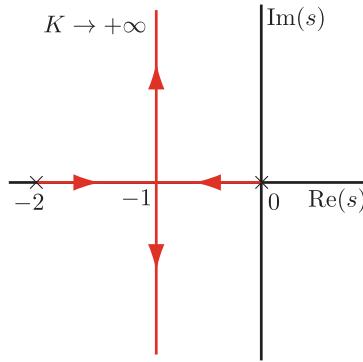


Figure 6.2: Locus of roots for  $s^2 + 2s + K$  as  $K$  goes from 0 to  $+\infty$ . The roots when  $K = 0$  are denoted by the symbol  $\times$ .

This plot shows us that the system is stable for all  $K > 0$  (infinite gain margin, see Chapter 8). The same information could have been obtained using the Routh-Hurwitz criterion but this figure contains much more information. We see that as  $K \rightarrow +\infty$  the closed-loop system becomes less damped, the step response has more overshoot and its frequency of oscillation increases. This suggests that for a good step response we shouldn't choose  $K$  too large for this system. ▲

We now show how to draw root-locus diagrams for higher order polynomials. Consider the system in Figure 6.3 where we have pulled a gain  $K$  out of the controller.

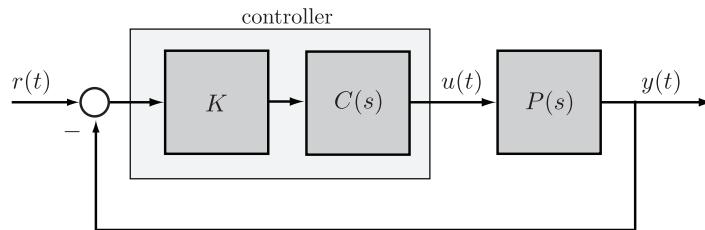


Figure 6.3: System considered in basic root-locus.

The characteristic polynomial of this system is  $\pi(s) = D(s) + KN(s)$  where  $N(s) := N_p N_c$  and  $D(s) := D_p D_c$ . The roots of the characteristic polynomial are of course

$$\{s \in \mathbb{C} : \pi(s) = 0\} = \{s \in \mathbb{C} : D(s) + KN(s) = 0\}. \quad (6.1)$$

The **root-locus** is a drawing of how the roots of  $\pi(s) = D(s) + KN(s)$  vary as a function of  $K \in \mathbb{R}$ . For now we make the following assumption.

**Assumption 6.1.1.** Let  $D, N \in \mathbb{R}[s]$  and let  $n := \deg D(s)$  and  $m := \deg N(s)$ .

1.  $C(s)P(s)$  is proper, i.e.,  $n \geq m$ .

2.  $K \geq 0$ .

3.  $N(s)$  and  $D(s)$  are monic, i.e., their leading coefficients are 1. ◀

In Section 6.3 we'll see how to handle cases where items 1 and 2 in Assumption 6.1.1 don't hold. As for item 3, that  $D$  should be taken as monic seems natural from (6.1). As for  $N$ , it can be made monic by redefining  $K$  if necessary.

**Remark 6.1.2.** As with Bode and Nyquist plots, the root-locus of a polynomial can be generated using software. The MATLAB graphical user interfaces started by the commands `rltool` or `sisotool` are very powerful. We discuss how to do these plots by hand because, in most cases, it is more important to quickly determine the *trend* of the root loci rather than the exact root-locus plot. ◆

### Construction rules

**Rule 1** The root-locus of  $D(s) + KN(s)$  is symmetric about the real axis.

This rule follows from the fact that  $\pi \in \mathbb{R}[s]$  so that its complex roots always appear as conjugate pairs.

**Rule 2** There are  $n$  branches of the root-locus.

This rule follows from the fact that, by Assumption 6.1.1, the polynomial  $D(s) + KN(s)$  has degree  $n$ .

**Rule 3** The root-locus of  $D(s) + KN(s)$  is a continuous function of  $K$ .

This follows from the fact that the roots of a polynomial in  $\mathbb{R}[s]$  are a continuous function of its coefficients.

**Rule 4** When  $K = 0$  the roots of  $D(s) + KN(s)$  equal the roots of  $D(s)$ .

**Rule 5** As  $K \rightarrow +\infty$  there are  $m$  branches of the root-locus that approach the roots of  $N(s)$ .

This follows from re-writing  $D(s) + KN(s) = 0$  in the form

$$\frac{1}{K}D(s) + N(s) = 0.$$

As  $K \rightarrow +\infty$ , the  $s$  with finite magnitude for which this equation holds approach the roots of  $N$ .

**Rule 6** The  $n - m$  branches of the root-locus that do not approach the zeros of  $N(s)$  asymptotically tend towards infinity following asymptotes that originate from the point

$$\sigma := \frac{\sum \text{roots of } D(s) - \sum \text{roots of } N(s)}{n - m} \quad (\text{centroid of } \pi). \quad (6.2)$$

The angles of the **asymptotes** are  $\phi_1, \dots, \phi_{n-m}$  where

$$\phi_k := \frac{(2k - 1)\pi}{n - m}, \quad k \in \{1, \dots, n - m\}. \quad (6.3)$$

Figure 6.4 shows the asymptote patterns for different values of  $n - m$ .

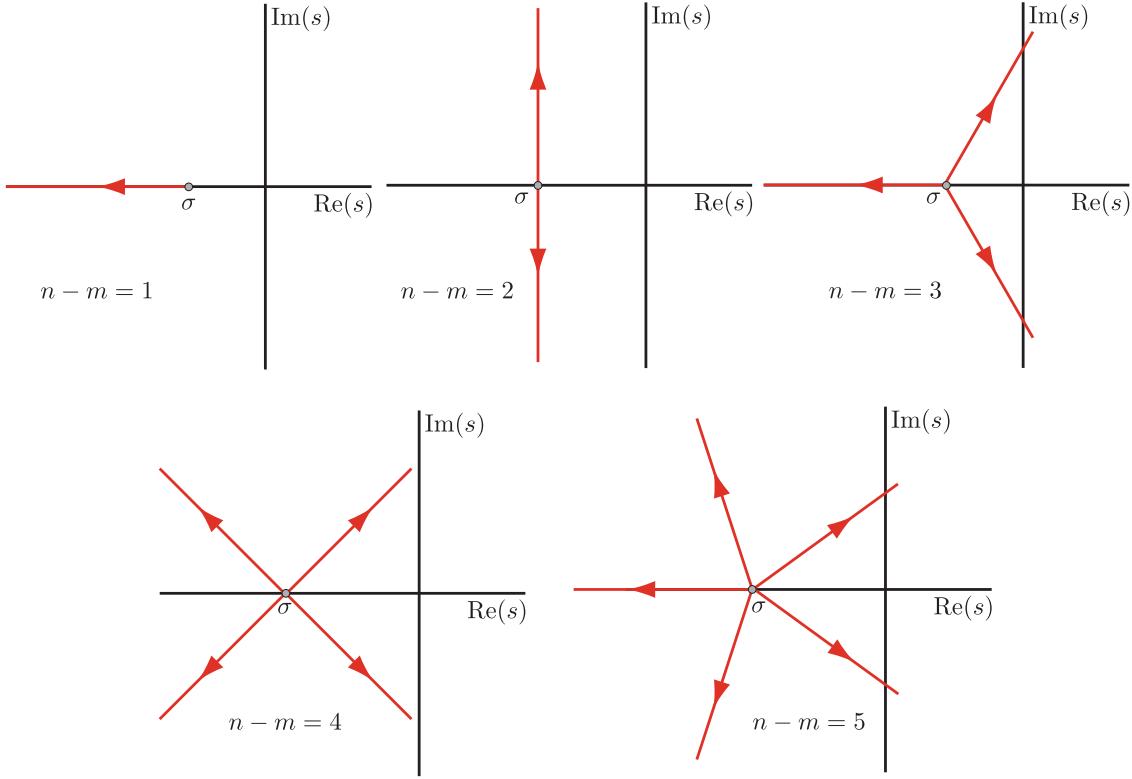


Figure 6.4: Asymptote patterns generated by Rule 6.

**Rule 7** (“no–yes–no rule”) A point  $s_0$  on the real axis is part of the root-locus if, and only if,  $s_0$  is to the left of an odd number of poles and zeros of  $P(s)C(s) = N(s)/D(s)$ .

This rule follows from (6.1) and the angular constraint each point  $s_0$  on the root-locus must satisfy

$$\angle N(s_0) - \angle(D(s_0)) = \angle\left(-\frac{1}{K}\right) = \pi \quad (\text{for } K > 0). \quad (6.4)$$

**Rule 8** (angles of arrival and departure) As  $K$  varies from 0 to  $+\infty$  the angle that a branch of the root-locus makes when leaving a root of  $D(s)$  (**departure angle**) and the angles of branches that arrive at roots of  $N(s)$  (**arrival angle**) can be computed using (6.4).

To find these angles note that at each point on the root-locus we have

$$\angle(s - z_1) + \dots + \angle(s - z_m) - \angle(s - p_1) - \dots - \angle(s - p_n) = \pi \quad (\text{for } K > 0)$$

where  $z_i \in \mathbb{C}$  are the roots of  $N(s)$  and  $p_i \in \mathbb{C}$  are the roots of  $D(s)$ .

For example, let  $\theta_{p_i}$  denote the angle of departure from  $p_i$ . Then  $\theta_{p_i}$  satisfies

$$\theta_{p_i} = \sum_{j=1}^m \angle(p_i - z_j) - \sum_{\substack{j=1 \\ j \neq i}}^n \angle(p_i - p_j) - \pi.$$

Similarly, the angle of arrival  $\theta_{z_i}$  at  $z_i$  is

$$\theta_{z_i} = \sum_{j=1}^n \angle(z_i - p_j) - \sum_{\substack{j=1 \\ j \neq i}}^m \angle(z_i - z_j) + \pi.$$

**Rule 9** Given a point  $s \in \mathbb{C}$  on the root locus, the corresponding value of  $K$  is given by

$$K = -\frac{D(s)}{N(s)}.$$

The value of  $K$  above is real if and only if  $s$  is on the root locus.

**Rule 10** The values of  $K$  for which the root-locus crosses the imaginary axis can be computed using the Routh array. See Examples 5.3.6 and 7.2.2.

### Procedure for plotting a root-locus

Given a polynomial  $D(s) + KN(s)$  we can plot its root-locus for  $K \geq 0$  as follows.

1. Compute the roots  $\{z_1, \dots, z_m\}$  of  $N(s)$  and place a  $\circlearrowleft$  at the location of each root in  $\mathbb{C}$ .
2. Compute the roots  $\{p_1, \dots, p_n\}$  of  $D(s)$  and place a  $\times$  at the location of each root in  $\mathbb{C}$ .
3. Use the “no–yes–no” rule (Rule 7) to fill in the portion of the real axis that are on the root-locus.
4. Compute the centroid  $\sigma$  using (6.2).
5. Draw the  $n - m$  asymptotes using (6.3).
6. Use Rule 8 to compute the departure and arrival angles. This is usually only needed for complex conjugate pairs and, occasionally, repeated real roots.
7. If you are lucky, you can give a reasonable guess as to how the root-locus behaves.

The last step is in some sense the most crucial. It is possible that one can do the steps preceding it, and still get the root-locus wrong. Some experience is involved in knowing how a “typical” root-locus diagram looks, and then extrapolating this to a given example.

## 6.2 Examples

**Example 6.2.1.** Consider the system in Figure 6.3 with

$$P(s) = \frac{1}{s^2 + 2s + 5}, \quad KC(s) = K \left( 1 + \frac{1}{0.25s} \right) \quad (\text{PI controller}).$$

Draw the root-locus of the closed-loop system for  $K \geq 0$ .

1. In this case  $N(s) = (s + 4)$  so  $m = 1$  and  $z_1 = -4$ .
2. We have  $D(s) = s(s^2 + 2s + 5)$  so  $n = 3$  and  $\{p_1, p_2, p_3\} = \{0, -1 + j2, -1 - j2\}$ .
3. By the no–yes–no rule only  $(-4, 0) \subset \mathbb{R}$  is part of the root-locus.
4. The centroid is

$$\sigma = \frac{0 + (-1 - j2) + (-1 + j2) - (-4)}{3 - 1} = 1$$

5. The asymptotes are

$$\phi_1 = \frac{\pi}{2}, \quad \phi_2 = \frac{3\pi}{2} \sim -\frac{\pi}{2}.$$

6. The departure angle from  $p_2$  is given by

$$\begin{aligned}\theta_{p_2} &= \angle(-1 + j2 - (-4)) - \angle(-1 + j2 - 0) - \angle(-1 + j2 - (-1 - j2)) - \pi \\ &= \arctan2(3, 2) - \arctan2(-1, 2) - \arctan2(0, 4) - \pi \\ &= 33.7^\circ - 116^\circ - 90^\circ - 180^\circ \\ &= -352.9^\circ \sim 7.125^\circ.\end{aligned}$$

By Rule 1 this calculation also yields the departure angle of  $p_3$ . By Rules 4 and 5 we can easily deduce the departure / arrival angles of real poles / zeros.

7. For the sake of completeness, we'll compute the imaginary axis crossings though at this point we can roughly plot the root-locus with the information we've already computed. The Routh table for  $\pi(s) = s(s^2 + 2s + 5) + K(s + 4) = s^3 + 2s^2 + (K + 5)s + 4K$  is

$s^3$	1	$K + 5$
$s^2$	2	$4K$
$s^1$	$5 - K$	0
$s^0$	$4K$	

When  $K = 5$  we get a row of zeros in  $s^1$ . The auxiliary polynomial is  $\pi_a(s) = 2s^2 + 20$  whose roots are  $s = \pm j\sqrt{10}$ . Thus the root-locus plot cross  $j\mathbb{R}$  at these points when  $K = 5$ .

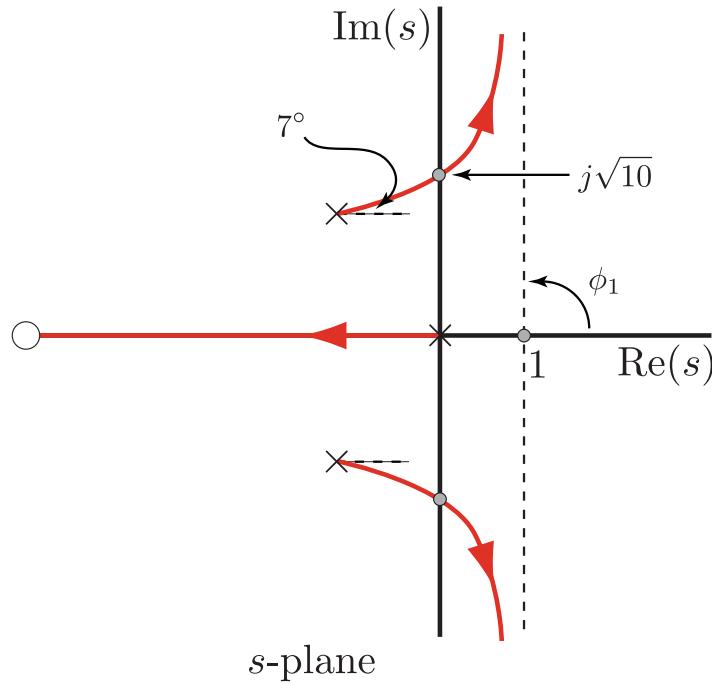


Figure 6.5: Root-locus for Example 6.2.1.

The resulting root-locus plot is shown in Figure 6.5. ▲

The previous example was done in great detail but most of the time a root-locus can be very quickly sketched. This makes root-loci great for quick system analysis.

**Example 6.2.2.** Draw the root-locus for  $\pi(s) = D(s) + KN(s)$  where  $D(s) = s^3(s + 4)$ ,  $N(s) = (s + 1)$ . Here  $n = 4$ ,  $m = 1$ . The centroid and asymptotes are

$$\sigma = \frac{-4 - (-1)}{4 - 1} = -1, \quad \phi_1 = \frac{\pi}{3}, \quad \phi_2 = \pi, \quad \phi_3 = -\frac{\pi}{3}.$$

With this information along with Rule 1 and the “no–yes–no” rule we can obtain the root-locus as shown in Figure 6.6. ▲

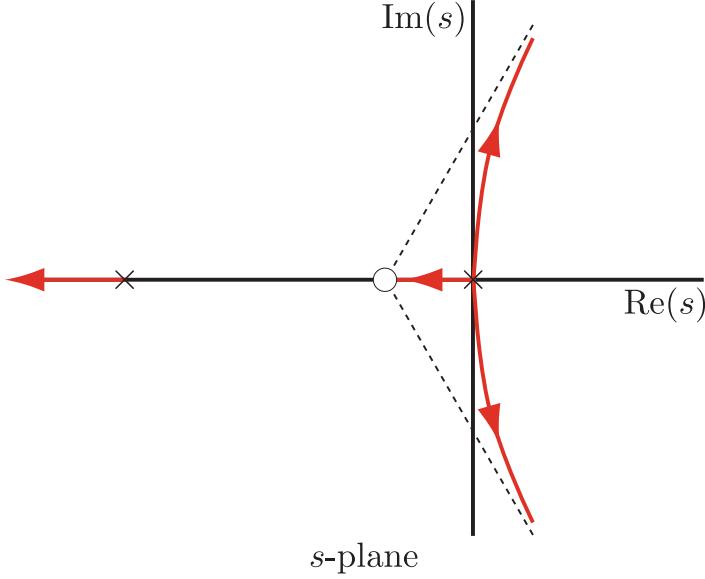


Figure 6.6: Root-locus for Example 6.2.2.

**Exercise 6.1.** Re-create the root-locus in Figure 7.3 using the procedure outlined in this chapter.

Root-locus diagrams are useful for deciding on the *structure* of the controller as the next four examples demonstrate.

**Example 6.2.3. (Inverted pendulum on a cart and proportional control)** Consider the inverted pendulum on a cart from Section 5.1 where the angle of the pendulum is taken as the output

$$P(s) = \frac{1}{M_1 L s^2 - g(M_1 + M_2)}.$$

Can a proportional controller stabilize this system in unity feedback? With  $C(s) = K_p$  the characteristic polynomial of the system is  $\pi(s) = D(s) + K N(s)$  with

$$D(s) = s^2 - \frac{g}{M_1 L} (M_1 + M_2), \quad N(s) = 1, \quad K = \frac{K_p}{M_1 L}.$$

Here  $n = 2$ ,  $m = 0$  and the root-locus is drawn in Figure 6.7. We conclude that a proportional controller cannot render the closed-loop system internally stable. ▲

**Example 6.2.4. (Inverted pendulum on a cart and proportional-derivative control)** Continuing with the inverted pendulum on a cart, determine if an (ideal) PD controller can render the feedback system stable. With

$$C(s) = K_p (1 + T_d s)$$

the characteristic polynomial of the system is  $\pi(s) = D(s) + K N(s)$  with

$$D(s) = s^2 - \frac{g}{M_1 L} (M_1 + M_2), \quad N(s) = s + \frac{1}{T_d}, \quad K = \frac{K_p T_d}{M_1 L}.$$

Our root-locus rules were derived for the case where there is only one undetermined parameter in  $\pi(s)$ . In this example we have two free parameters – the proportional gain  $K_p$  and the derivative time constant  $T_d$ . We break the root-locus into three cases determined by the location of  $N(s)$ .

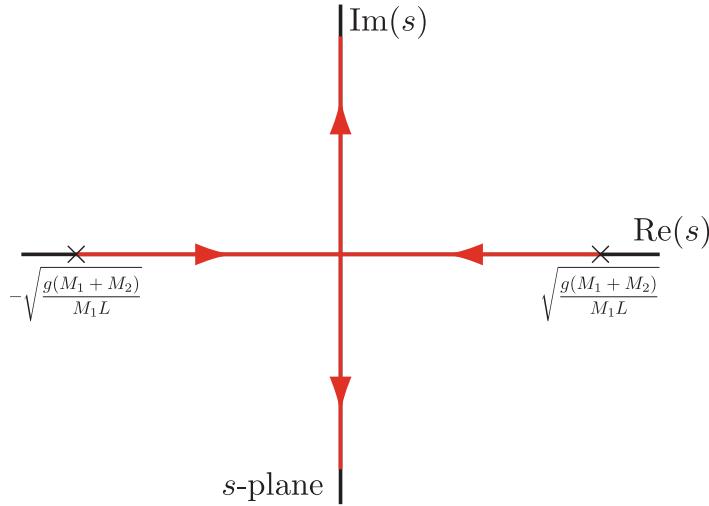


Figure 6.7: Root-locus for Example 6.2.3.

**Case 1** ( $T_d < 0$ ) In this case the root of  $N(s)$  is in the right half-plane. This root is a zero of the closed-loop system. Placing a zero in  $\mathbb{C}^+$  means that, by Rules 5 and 7, we will not be able to stabilize the system. Furthermore, right half-plane zeros limit the achievable performance of a closed-loop system (Section 4.4 and Section 9.6.2) so we avoid them if at all possible.

**Case 2** ( $T_d > \sqrt{(M_1 L)/(g(M_1 + M_2))}$ ) In this case the root of  $N(s)$  is in  $\mathbb{C}^-$  between the left root of  $D(s)$  and the imaginary axis. We have  $n = 2$ ,  $m = 1$ . The root locus is shown in Figure 6.8a. In this case we see that the PD controller can stabilize the closed-loop system but placing the zero between the poles limits how fast we can make the response. The dominant pole can never move to the left of the zero no matter how high we make the proportional gain  $K_p$ .

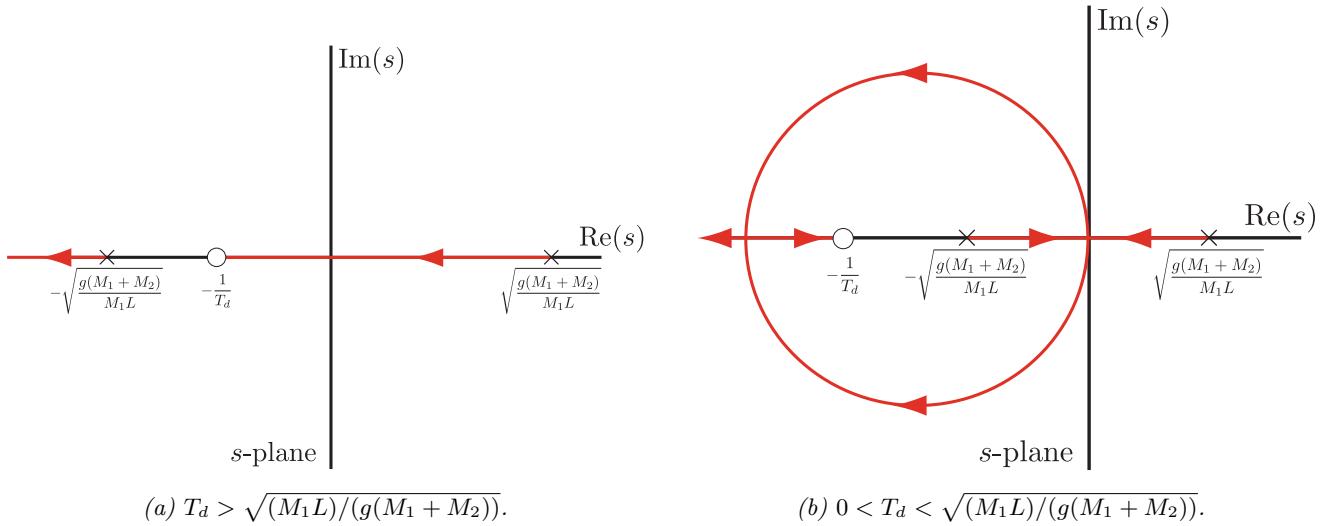


Figure 6.8: Root-loci for the PD controllers of Example 6.2.4.

**Case 3** ( $0 < T_d < \sqrt{(M_1 L)/(g(M_1 + M_2))}$ ) In this case the root of  $N(s)$  is to the left of both roots of  $D(s)$ . We still have  $n = 2$ ,  $m = 1$ . The root-locus is shown in Figure 6.8b. In this case we again see that the PD controller can stabilize the closed-loop system. The root-locus shows that we can get both poles far to left which means a high bandwidth and fast response. On the other hand, the control effort will be high.

**Exercise 6.2.** Draw the root-locus for Case 1 of the previous example.

**Example 6.2.5. (Inverted pendulum on a cart and proportional-integral control)** Once again the plant is the inverted pendulum on a cart. Determine if a PI controller can render the feedback system stable. With

$$C(s) = K_p \left( 1 + \frac{1}{T_i s} \right)$$

the characteristic polynomial of the system is  $\pi(s) = D(s) + KN(s)$  with

$$D(s) = s \left( s^2 - \frac{g}{M_1 L} (M_1 + M_2) \right), \quad N(s) = s + \frac{1}{T_i}, \quad K = \frac{K_p}{M_1 L}.$$

In this case  $n = 3$  and  $m = 1$ . The choice of integral time constant  $T_i$  determines the location of the root of  $N(s)$ . We consider two cases.

**Case 1** ( $T_i > \sqrt{(M_1 L)/(g(M_1 + M_2))}$ ) In this case the zero (root of  $N(s)$ ) is in  $\mathbb{C}^-$  between the pole at  $s = 0$  and the pole at  $s = -\sqrt{g(M_1 + M_2)/(M_1 L)}$ . Since  $n - m = 2$  there are two asymptotes. The centroid is given by

$$\sigma = \frac{-\sqrt{g(M_1 + M_2)/(M_1 L)} + \sqrt{g(M_1 + M_2)/(M_1 L)} + 0 - 1/T_i}{3 - 1} = \frac{1}{2T_i}$$

which means it is to the right of imaginary axis and to left of the plant's unstable pole. The root-locus is shown in Figure 6.9a from which we conclude that this configuration for the PI controller cannot stabilize the pendulum on a cart system.

**Case 2** ( $0 < T_i < \sqrt{(M_1 L)/(g(M_1 + M_2))}$ ) The root of  $N(s)$  is now to the left of both plant poles. The analysis is the same as in Case 1 but now if  $T_i < 0.5\sqrt{(M_1 L)/(g(M_1 + M_2))}$ , then the centroid is to the right of the plant's positive pole. The root-locus is shown in Figure 6.9b

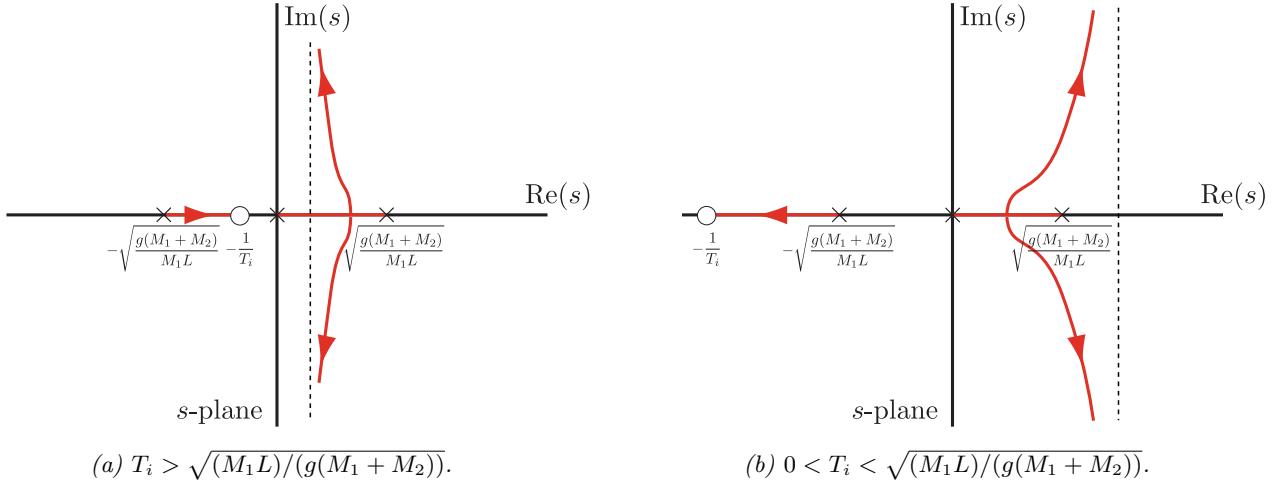


Figure 6.9: Root-loci for the PI controllers of Example 6.2.5.

We conclude that a PI controller will not stabilize this system. ▲

**Exercise 6.3.** Draw the root-locus for the previous example when  $T_i < 0$ .

**Exercise 6.4.** Re-create the root-locus in Figure 7.6 using the procedure outlined in this chapter.

**Example 6.2.6. (Inverted pendulum on a cart and proportional-integral-derivative control)** For our last example consider the pendulum on a cart controlled by an (ideal) PID controller

$$C(s) = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right).$$

the characteristic polynomial of the system is  $\pi(s) = D(s) + KN(s)$  with

$$D(s) = s \left( s^2 - \frac{g}{M_1 L} (M_1 + M_2) \right), \quad N(s) = s^2 + \frac{1}{T_d} s + \frac{1}{T_d T_i}, \quad K = \frac{K_p T_d}{T_i M_1 L}.$$

The zero locations, i.e., roots of  $N(s)$ , can be placed anywhere in the complex plane by appropriately choosing the values of  $T_i, T_d$ . By breaking things down into cases based on the zero locations, we can analyse the closed-loop system in the same manner as the previous examples. It turns out that the PID controller can stabilize the feedback system and it provides perfect step tracking to boot! This is hardly surprising given the discussion in Section 7.3. The diagram of the root-locus for this system is left as an exercise. ▲

**Exercise 6.5.** Draw the root-locus for  $K \geq 0$  for Example 6.2.6 when the roots of  $N(s)$  are complex conjugates and in  $\mathbb{C}^-$ .

**Exercise 6.6.** Draw the root-locus for  $K \geq 0$  for Example 6.2.6 when the roots of  $N(s)$  are real, distinct and to the left of both plant poles.

## 6.3 Non-standard problems

In this section we learn how to handle cases where Assumption 6.1.1 fails to hold as well as transfer functions in the feedback path.

### Non-unity feedback systems

Suppose that instead of system 6.3 we have the system shown in Figure 6.10.

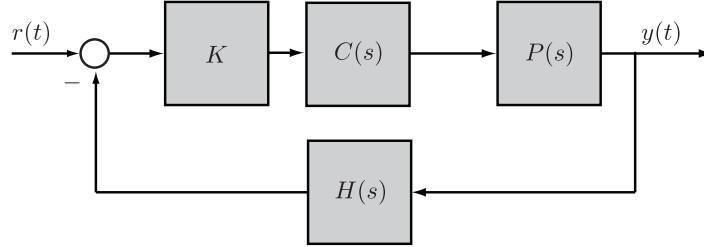


Figure 6.10: A non-unity feedback system.

In this case the loop gain is  $L(s) = KC(s)P(s)H(s)$  and the closed-loop poles satisfy  $1 + L(s) = 0$ . The characteristic polynomial is given by  $\pi(s) = D_c D_p D_h + K N_c N_p N_h =: D(s) + KN(s)$ . This polynomial has the same form as before so the root-locus construction proceeds in the exact same way. The unity-feedback situation becomes a special case in which  $N_h = D_h = 1$ .

### Controller is not a linear function of its gain

Figures 6.3 and 6.10 implicitly require that the controller be a linear function of  $K$  which allows us to pull it out of the controller block. This is not needed to draw a root locus. The requirement for drawing a root-locus is that the polynomial's coefficients be linear in the unknown parameter. That is, we only require that  $\pi(s)$  have the form  $D(s) + KN(s)$ . If  $K$  can't be pulled out of the controller, then  $\pi$  may still have the desired form except that, in general,  $N(s) \neq N_c N_p$  and  $D(s) \neq D_c D_p$ .

**Example 6.3.1.** Consider the system in Figure 6.1 with

$$P(s) = \frac{1}{s(s+2)}, \quad C(s) = 10(1 + T_d s) \quad (\text{ideal PD controller}).$$

In this case  $C(s)$  cannot be written in the form  $K C_1(s)$ . The characteristic polynomial of the system is

$$\pi(s) = s(s+2) + 10(1 + T_d s) = (s^2 + 2s + 10) + (10T_d)s =: D(s) + KN(s).$$

Notice that the characteristic polynomial's coefficients are linear in the unknown parameter  $K$  but that  $D(s) = s^2 + 2s + 10 \neq D_p D_c = s(s+2)$  and  $N(s) = s \neq N_c N_p$ . The root-locus for this system is shown in Figure 6.11. ▲

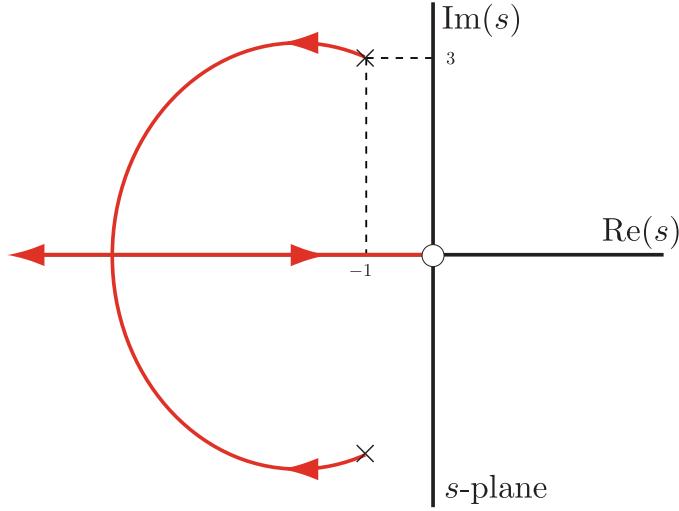


Figure 6.11: Root-locus for Example 6.3.1.

### Improper loop gain

Consider the polynomial  $\pi(s) = D(s) + KN(s)$  and suppose that  $\deg(D) < \deg(N)$ . This situation can arise when part 1 of Assumption 6.1.1 fails to hold as well as other situations. Observe that

$$D(s) + KN(s) = 0 \iff N(s) + \frac{1}{K}D(s) = 0.$$

If we define  $\hat{D}(s) := N(s)$ ,  $\hat{N}(s) := D(s)$  and  $\hat{K} = 1/K$  then the polynomial  $\hat{\pi}(s) = \hat{D}(s) + \hat{K}\hat{N}(s)$  satisfies Assumption 6.1.1 and its root-locus can be drawn for  $\hat{K} \geq 0$ . In order to recover the root-locus of  $\pi$  from the root-locus of  $\hat{\pi}$  we do the following.

1. Switch each  $\circ$  to  $\times$  in the root-locus of  $\hat{\pi}$ .
2. Switch each  $\times$  to  $\circ$  in the root-locus of  $\hat{\pi}$ .
3. Reverse the directional arrows.

**Example 6.3.2.** Consider the usual unity-feedback system in Figure 6.1 with

$$P(s) = \frac{1}{s(s+1)}, \quad C(s) = \frac{s+3}{Ts+1}.$$

Then

$$\pi(s) = (s^2 + 2s + 3) + Ts^2(s+1) =: D(s) + KN(s).$$

In this case  $\deg(D) = 2 < \deg(N) = 3$  so we use the approach described above. Define  $\hat{\pi} = \hat{D}(s) + \hat{K}\hat{N}(s)$  where

$$\hat{D}(s) := s^2(s+1), \quad \hat{N}(s) := s^2 + 2s + 3, \quad \hat{K} = \frac{1}{T}.$$

The root-locus of  $\hat{\pi}$  is shown in Figure 6.12a while the root-locus of  $\pi$  is shown in Figure 6.12b. ▲

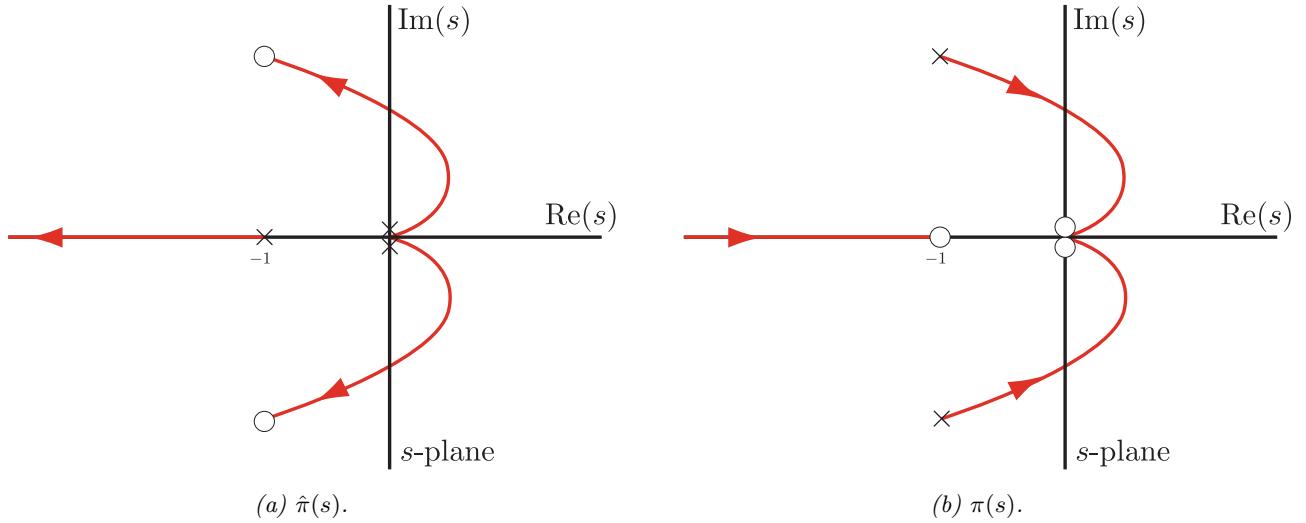


Figure 6.12: Root-loci for Example 6.3.2.

### Negative gain

Assumption 6.1.1 asks that  $K \geq 0$  but it is possible to obtain a root-locus for  $K < 0$ . Such a root-locus shows how the roots of  $\pi(s) = D(s) + KN(s)$  vary as  $K$  goes from 0 to  $-\infty$ . The root-locus for  $K < 0$  is called the **complementary root locus**.

The analysis for  $K < 0$  is very similar to the  $K > 0$  case. The main difference is that, since  $\angle -\frac{1}{K} = 0$ , the various angle conditions are slightly modified. In particular, for  $K < 0$ , the rules presented in Section 6.1 are modified as follows.

**Rule 1** No change.

**Rule 2** No change.

**Rule 3** No change.

**Rule 4** No change.

**Rule 5** As  $K \rightarrow -\infty$  there are  $m$  branches of the root that approach the roots of  $N(s)$ .

**Rule 6** The asymptote angle equation changes to

$$\phi_k := \frac{2k\pi}{n-m}, \quad k \in \{1, \dots, n-m\}. \quad (6.5)$$

**Rule 7** (“no–yes–no rule”) A point  $s_0$  on the real axis, is part of the root-locus if, and only if,  $s_0$  is to the left of an *even* number of poles and zeros of  $P(s)C(s) = N(s)/D(s)$ .

**Rule 8** (angles of arrival and departure) The angles of arrival and departure are calculated and interpreted in the same way except that the angle criterion is

$$\angle(s - z_1) + \cdots + \angle(s - z_m) - \angle(s - p_1) - \cdots - \angle(s - p_n) = 0 \quad (\text{for } K < 0)$$

where  $z_i \in \mathbb{C}$  are the roots of  $N(s)$  and  $p_i \in \mathbb{C}$  are the roots of  $D(s)$ .

**Rule 9** No change.

**Rule 10** No change.

## 6.4 Summary

In this chapter we gave a method for plotting the root-locus of a polynomial by hand. These sketches are useful for quickly understanding a system and how changing a gain affects performance. You should know the following.

1. What exactly are we seeing when we look at a root-locus diagram?
2. How to draw a root-locus by hand.
3. Given a point on the root-locus, you should be able to compute the gain  $K$  for which the polynomial has a root at the given point.
4. You should be able to do all the examples in Section 6.2 in full detail.
5. You should be able to draw the root-locus for any of the non-standard scenarios discussed in Section 6.3.

# Chapter 7

## PID control

In this chapter we briefly introduce **Proportional-Integral-Derivative (P.I.D.)** controllers. This particular control structure has become almost universally used in industrial control<sup>1</sup>. These controllers are popular because they are simple and have proven to be robust in the control of many important applications. The simplicity of these controllers is also their weakness since it limits the range of plants that they can control satisfactorily. The references [**Åström and Hägglund, 1995**], [**Goodwin et al., 2001**], [**Åström and Hägglund, 2006**] contain much more information for the interested student.

### Contents

7.1	Classical PID controller . . . . .	150
7.2	Empirical tuning . . . . .	157
7.3	Pole placement . . . . .	159
7.4	Integrator windup . . . . .	162
7.5	Summary . . . . .	164

### 7.1 Classical PID controller

The block diagrams of two closed-loop systems with ideal PID controllers are shown in Figure 7.1. The transfer

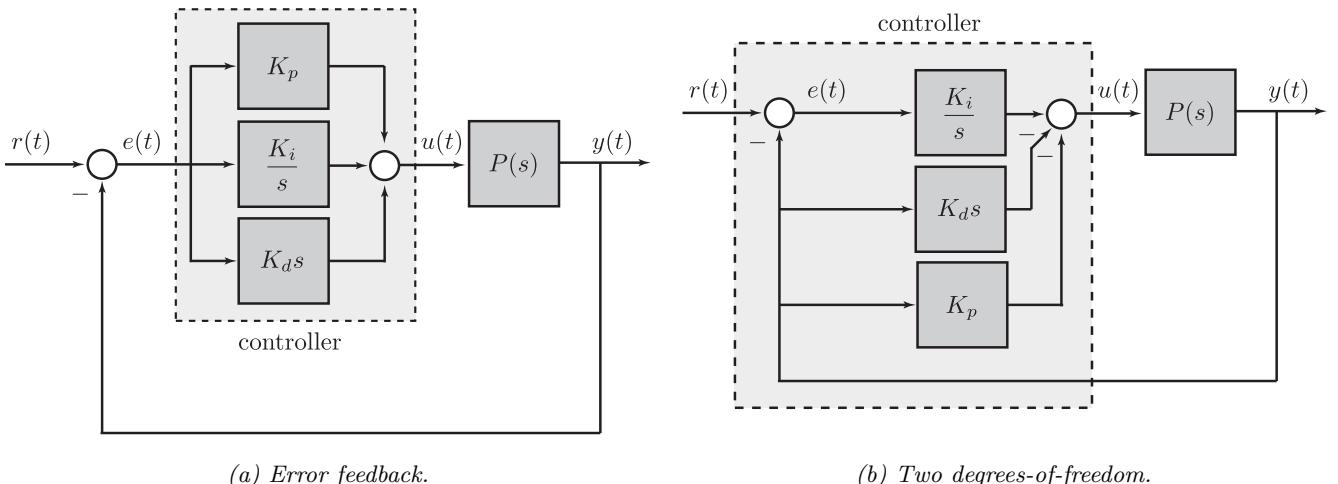


Figure 7.1: Block diagram of closed-loop system with ideal PID controllers.

<sup>1</sup>“Based on a survey of over eleven thousand controllers in the refining, chemicals and pulp and paper industries, 97% of regulatory controllers utilize PID feedback” [Desborough and Miller, 2002], [**Åström and Murray, 2019**], [**Samad, 2017**].

function of the controller in Figure 7.1a is

$$\frac{U(s)}{E(s)} = C(s) = K_p + \frac{K_i}{s} + K_d s. \quad (7.1)$$

The corresponding equation in the time-domain is

$$u(t) = K_p e(t) + K_i \int_0^t e(\tau) d\tau + K_d \frac{de(t)}{dt} \quad (7.2)$$

where  $e(t) = r(t) - y(t)$  as usual. The constants  $K_p$ ,  $K_i$  and  $K_d$  are called, respectively, the **proportional**, **integral** and **derivative gains**.

In Figure 7.1b the integral action acts on the error, but proportional and derivative action act on the process output  $y$ . The input output model is

$$U(s) = \frac{K_i}{s} E(s) - (K_p + K_d s) Y(s) = \frac{K_i}{s} R(s) - \left( K_p + K_i \frac{1}{s} + K_d s \right) Y(s). \quad (7.3)$$

This setup is often preferred when the reference signal  $r$  is discontinuous, e.g., a sequence of step changes. Differentiating a step function creates an impulse which we usually do not want in the control signal  $u$ .

**Exercise 7.1.** Let  $P(s)$  be real rational and write  $P = N_p/D_p$  where  $(N_p, D_p)$  is coprime. Find the TF from  $R$  to  $Y$  for both the systems in Figure 7.1. Show that the system in Figure 7.1a is input-output stable if and only if the system in Figure 7.1b is input-output stable.

The PID controller (7.1) is sometimes expressed in the equivalent form

$$C(s) = K_p \left( 1 + \frac{1}{T_i s} + T_d s \right) \quad (7.4)$$

where  $T_i$  is called the **integral time constant** or **reset time** while  $T_d$  is the **derivative time constant**.

**Exercise 7.2.** Show that the PID controllers (7.1) and (7.4) are identical.

The PID controllers (7.1) and (7.3) are idealized because the resulting TFs are improper which makes them impossible to implement without approximations<sup>2</sup>. Furthermore, an ideal differentiator has the undesirable property that it amplifies high frequency noise (see Figure 3.15). Therefore, in practice the derivative term is “rolled off” and replaced by a low pass filtered version

$$K_d \frac{s}{\tau_d s + 1} \quad \tau_d > 0. \quad (7.5)$$

The TF (7.5) approximates a derivative for low frequencies ( $\omega < 1/\tau_d$ ) and approaches the ideal differentiator as  $\tau_d \rightarrow 0$ . Figure 7.2 shows the Bode plot for an ideal differentiator and the “rolled off” version. With the approximated differentiator the PID controller (7.4) (equivalently (7.1)) becomes

$$C(s) = K_p \left( 1 + \frac{1}{T_i s} + \frac{T_d s}{\tau_d s + 1} \right) \quad (7.6)$$

while the two degree-of-freedom version (7.3) becomes

$$U(s) = \frac{K_i}{s} R(s) - \left( K_p + K_i \frac{1}{s} + K_d \frac{s}{\tau_d s + 1} \right) Y(s). \quad (7.7)$$

**Remark 7.1.1.** In classical PID control design,  $\tau_d \neq 0$  was largely seen as a necessary evil, i.e., as a necessary departure from a pure proportional, integral and derivative action. As a result almost all industrial PID controllers set  $\tau_d$  as a fixed fraction of  $T_d$ , rather than viewing it as an independent design parameter in its own right. A more modern approach to PID design is to treat the time constant of the derivative as an important degree of freedom available to the designer. ♦

<sup>2</sup>An exception is when the derivative term can be measured directly using sensors, e.g., a tachometer on a motor.

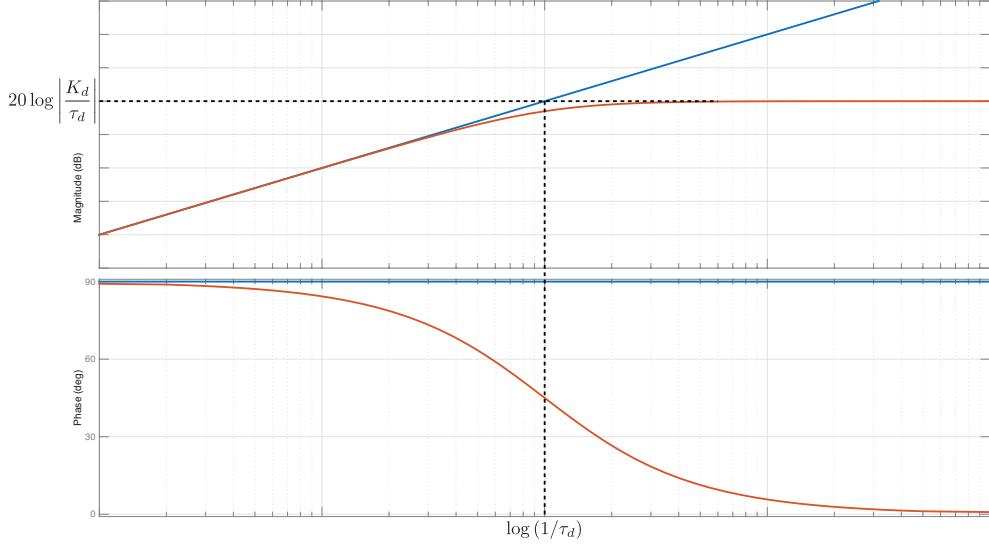


Figure 7.2: Bode plot of an ideal differentiator (blue) and an implementable low pass filtered version (red).

**Remark 7.1.2.** There are various different ways of parametrizing a PID controller. The form (7.1) is useful for analysis because it is linear in the controller gains and turning off the integral action corresponds to setting  $K_i = 0$ . The form in (7.6) is called the **standard form**. In this course we usually stick to the standard form (7.6) or its ideal version (7.4). However, when working with industrial PID controllers you should be aware that different definitions of PID gains abound. ♦

Equation (7.1) is an idealization but it is a useful abstraction for understanding PID controllers. We now use this idealized controller to develop some intuition about the closed-loop system in Figure 7.1a. In our discussion we assume that the plant TF is given by

$$P(s) = \frac{1}{(s+1)^3}. \quad (7.8)$$

Of course, the behaviour of the closed-loop system in Figure 7.1a depends very much on the plant dynamics. By focusing on a specific plant our intention is to simply develop intuition on the role that each component of a PID controller plays.

### 7.1.1 Proportional control

Consider a pure proportional controller

$$C(s) = K_p,$$

i.e., select  $K_i = K_d = 0$  in (7.1). A proportional controller provides a contribution which depends on the instantaneous value of the tracking error. A proportional controller can control any stable plant, but it provides limited performance and generally gives nonzero steady-state tracking error. For the system in Figure 7.1a with plant (7.8) the closed-loop TF from  $r$  to  $y$  is

$$\frac{C(s)P(s)}{1 + C(s)P(s)} = \frac{K_p}{s^3 + 3s^2 + 3s + 1 + K_p}.$$

The Routh array for this system is given by

$$\begin{array}{c|cc} s^3 & 1 & 3 \\ s^2 & 3 & 1 + K_p \\ s^1 & \frac{8 - K_p}{3} \\ s^0 & 1 + K_p. \end{array}$$

Therefore the closed-loop system is internally stable if and only if  $K_p \in (-1, 8)$ . Assuming internal stability and applying the FVT, the steady-state tracking error for a unit step reference signal is

$$e_{ss} = \frac{1}{1 + K_p P(0)} = \frac{1}{1 + K_p}.$$

Thus we see that this system will always have steady-state error though the error gets smaller as  $K_p \rightarrow 8$ . On the other hand, as  $K_p \rightarrow 8$  the system is getting closer to becoming unstable. Figure 7.3 is called a root-locus diagram<sup>3</sup>. It shows how the roots of  $s^3 + 3s^2 + 3s + 1 + K_p$  vary in the  $s$ -plane as we increase  $K_p$ . From

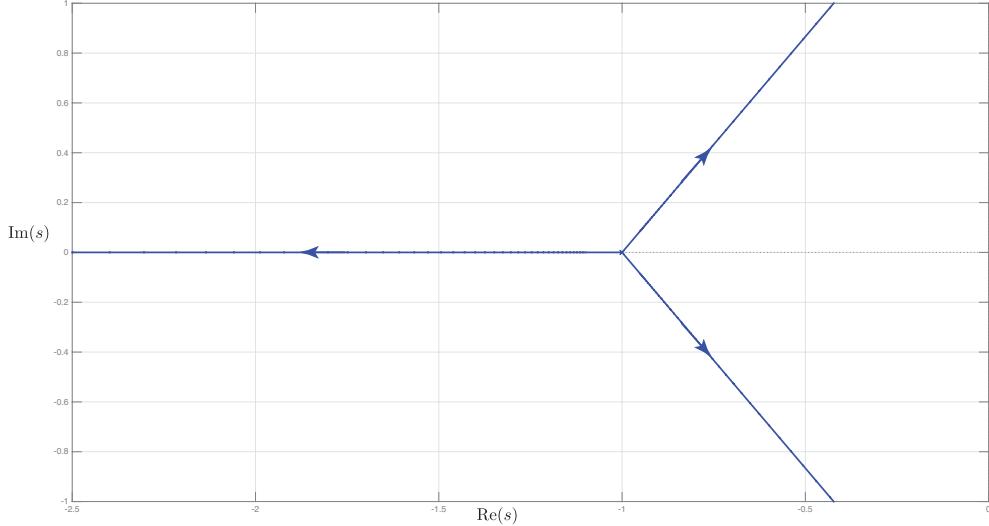
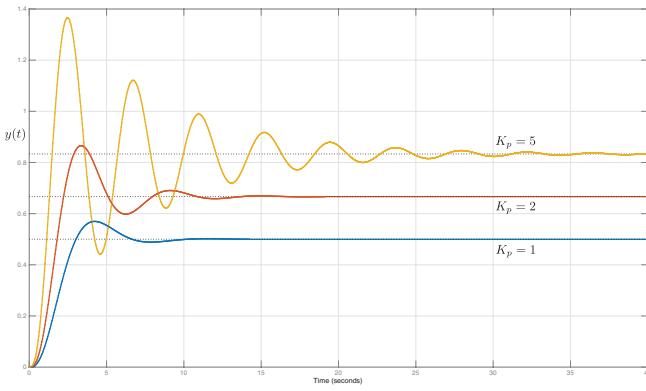
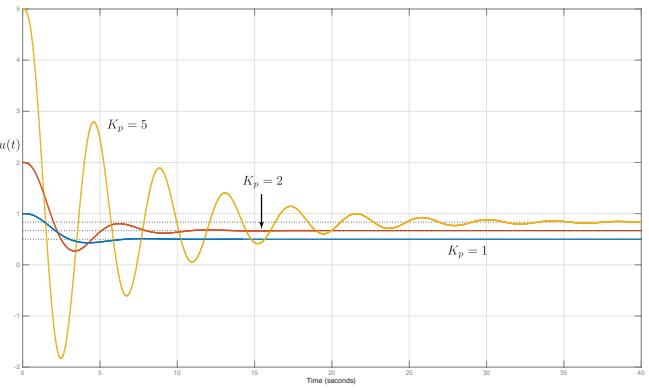


Figure 7.3: Root-locus plot from Section 7.1.1 (proportional control).

Figure 7.3 and based on our discussion in Chapter 4, we observe that as  $K_p$  gets larger the complex conjugate poles approach the imaginary axis and dominate the time response. As  $K_p$  gets larger, the dominant poles have smaller real part (larger settling time), the damping ratio gets smaller (more overshoot) and the imaginary part gets larger (increased frequency of oscillations). We conclude that while we can reduce the steady-state tracking error by increasing the proportional gain, the transient performance gets worse and the system eventually goes unstable. These predictions are verified in Figure 7.4 where we simulate the closed-loop system for different values of  $K_p$ .



(a) Output  $y(t)$ .



(b) Control signal  $u(t)$ .

Figure 7.4: Step response of system in Figure 7.1a with plant (7.8) and proportional control with  $K_p \in \{1, 2, 5\}$ .

<sup>3</sup>See Chapter 6.

To eliminate steady-state tracking error for step inputs the proportional term can be changed to

$$U(s) = K_p E(s) + K_{ff} R(s) \quad (7.9)$$

where the gain  $K_{ff}$  is a **feedforward** or bias term. It is sometimes called **reset** in the PID literature. This control architecture is shown in Figure 7.5 which justifies the term feedforward.

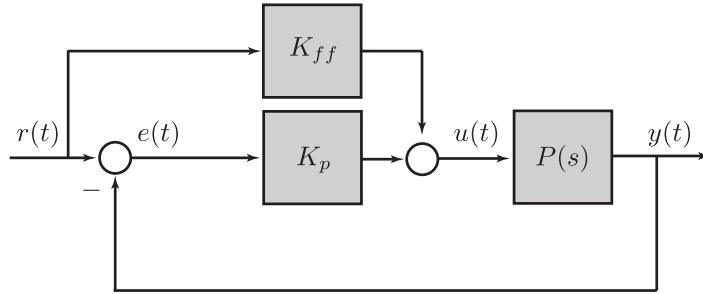


Figure 7.5: Proportional controller with feedforward.

**Exercise 7.3.** Find the TF from  $r$  to  $y$  for the system in Figure 7.5.

If we choose  $K_{ff} = 1/P(0)$  then the steady-state tracking error for step inputs will be zero so long as there are no disturbances.

**Exercise 7.4.** Regarding the system in Figure 7.5, show that (a) input-output stability of the system in does not depend on  $K_{ff}$  and (b) if the system is input-output stable and  $K_{ff} = 1/P(0)$  then  $e_{ss} = 0$  for step inputs.

The problem with the solution for eliminating steady-state tracking error in Figure 7.5 is that (i) it requires perfect knowledge of  $P(0)$  (ii) it only works if there are no disturbances and no noise in the loop and (iii) the value of  $K_{ff}$  often needs to be re-calibrated. As we saw in Section 5.4, a far better solution is to use the internal model principle to achieve perfect step tracking. This helps motivate the integral action in a PID controller.

## 7.1.2 Integral control

Consider a pure integral controller

$$C(s) = \frac{K_i}{s},$$

i.e., select  $K_p = K_d = 0$  in (7.1). The main function of integral action is to make sure that the system output agrees with the setpoint in steady-state. The output of the intergral term is proportional to the accumulated error. If  $K_i > 0$  a small positive tracking error will always lead to an increasing control signal while a small negative tracking error will always lead to a decreasing control signal.

For the system in Figure 7.1a with plant (7.8) the closed-loop TF from  $r$  to  $y$  is

$$\frac{C(s)P(s)}{1 + C(s)P(s)} = \frac{K_i}{s^4 + 3s^3 + 3s^2 + s + K_i}.$$

The Routh array for this system is given by

$$\begin{array}{c|ccc} s^4 & 1 & 3 & K_i \\ s^3 & 3 & 1 & 0 \\ s^2 & \frac{8}{3} & K_i \\ s^1 & 1 - \frac{9}{8}K_i \\ s^0 & K_i. \end{array}$$

Therefore the closed-loop system is internally stable if and only if  $K_i \in (0, 8/9)$ . Thus we see that integral action can make the system go unstable if the integral gain is too high. Figure 7.6 shows how the roots

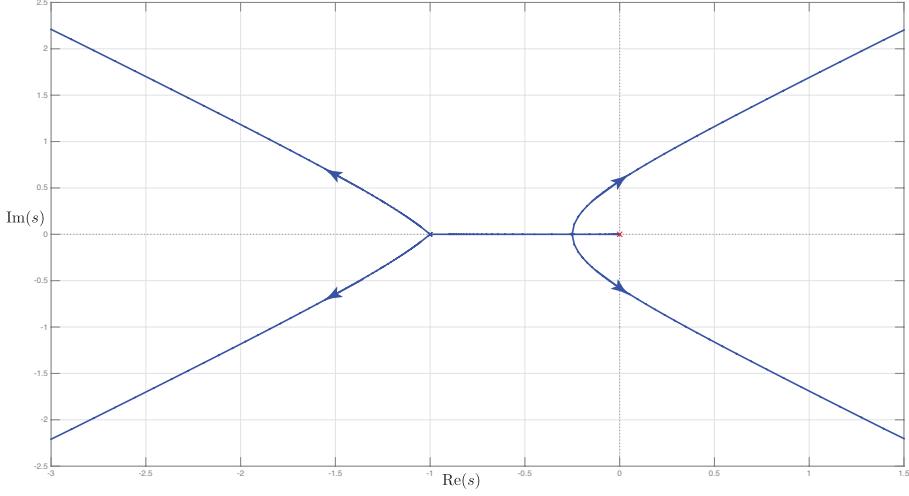
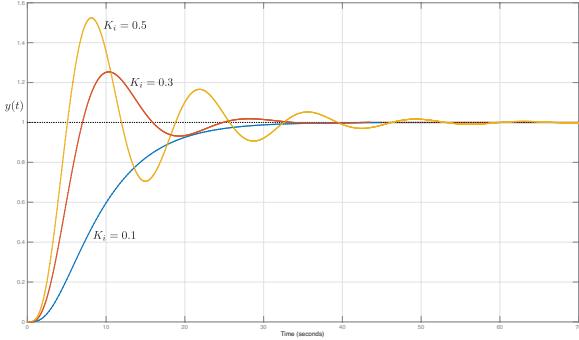
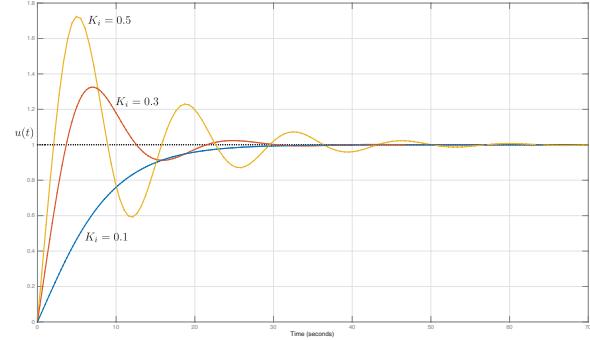


Figure 7.6: Root-locus plot from Section 7.1.2 (integral control).

of  $s^4 + 3s^3 + 3s^2 + s + K_i$  vary in the  $s$ -plane as we increase  $K_i$ . Again we see that as  $K_i$  gets large, two complex conjugate poles get closer to the imaginary axis and dominate the time-domain response. Increasing  $K_i$  results in longer settling time, more overshoot and increased frequency of oscillation. Figure 7.7 shows simulation results for different values of  $K_i$  which match our expectations.

(a) Output  $y(t)$ .(b) Control signal  $u(t)$ .Figure 7.7: Step response of system in Figure 7.1a with plant (7.8) and integral control with  $K_i \in \{0.1, 0.3, 0.5\}$ .

### 7.1.3 Derivative control

Consider an ideal derivative controller

$$C(s) = K_d s,$$

i.e., select  $K_p = K_i = 0$  in (7.1). This controller acts on the rate of change of the tracking error. It is sometimes referred to as a predictive mode because of its dependence on the error trend. To see this consider the time-domain expression of a PD controller

$$u(t) = K_p \left( e(t) + T_d \frac{de}{dt} \right).$$

The term inside the bracket can be interpreted as a prediction of the tracking error at time  $t + T_d$  by linear interpolation as illustrated in Figure 7.8. The prediction horizon  $T_d = K_d/K_p$  is the derivative time constant of the controller.

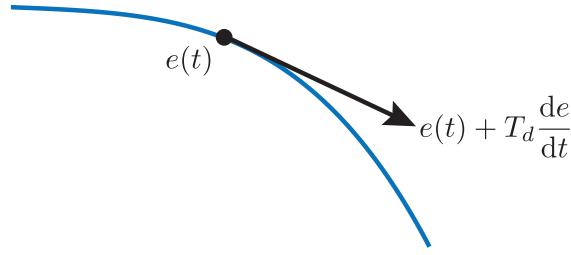


Figure 7.8: PD controller viewed as predictive action. Increasing  $T_d$  increases the prediction horizon.

For the system in Figure 7.1a with plant (7.8) and an ideal derivative controller the closed-loop TF from  $r$  to  $y$  is

$$\frac{C(s)P(s)}{1 + C(s)P(s)} = \frac{K_d s}{s^3 + 3s^2 + (3 + K_d)s + 1}.$$

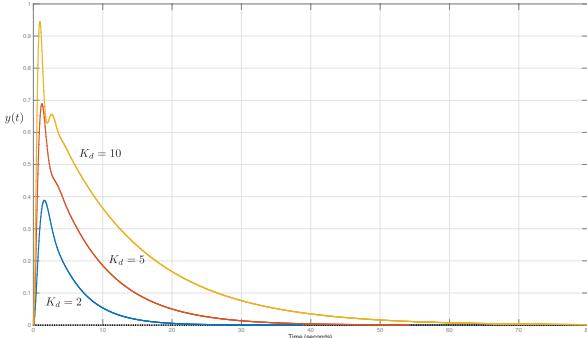
The Routh array for this system is given by

$$\begin{array}{c|cc} s^3 & 1 & 3 + K_d \\ s^2 & 3 & 1 \\ s^1 & \frac{8}{3} + K_d & \\ s^0 & 1. & \end{array}$$

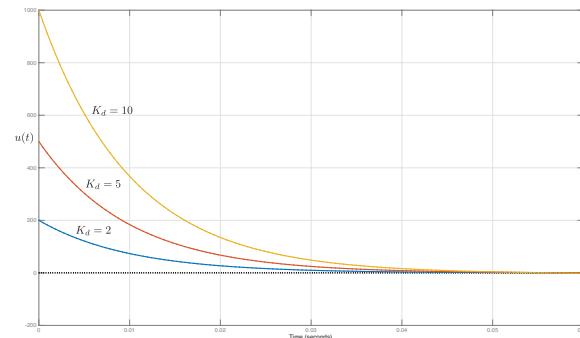
Therefore the closed-loop system is internally stable if and only if  $K_d > -8/3$ . The fact that high gain derivative control does not destabilize the system gives some justification to the intuitive notion that derivative control has a stabilizing effect. Another interpretation of this intuitive notion is that derivative control acts like friction in a mechanical system or resistance in an electrical system and tends to dampen oscillations and dissipate energy resulting in a stable system.

A main draw back of derivative control is that it cannot provide zero steady-state tracking error. Indeed, the simulations in Figure 7.9a show that in the case of plant (7.8), the steady state output is zero<sup>4</sup>. Derivative control can also lead to large control signals in response to high frequency tracking error as seen in Figure 7.9b. This latter concern can be alleviated using the architecture from Figure 7.1b. The simulations in Figure 7.9 were done using a low pass filtered version of the differentiator (7.5) with  $\tau_d = 0.01$ . Note that for the filtered differentiator the closed-loop system can become unstable if  $K_d$  is too large.

**Exercise 7.5.** Let  $\tau_d = 0.01$  and use a Routh table to find the conditions on  $K_d$  so that the system in Figure 7.1a with plant (7.8) and controller (7.5) is internally stable.



(a) Output  $y(t)$ .



(b) Control signal  $u(t)$ .

Figure 7.9: Step response of system in Figure 7.1a with plant (7.8) and derivative control with  $\tau_d = 0.01$  and  $K_d \in \{2, 5, 10\}$ .

<sup>4</sup>This is not surprising since the TF from  $r$  to  $y$  has a zero at  $s = 0$ .

## 7.2 Empirical tuning

Two classical methods for determining the parameters of PID controllers were presented by Ziegler and Nichols in 1942 [Ziegler and Nichols, 1942]. The Ziegler-Nichols methods had a huge impact when they were introduced in the 1940s. The rules are simple to use and give good initial conditions for manual tuning. The Ziegler-Nichols methods are still widely used and often form the basis for tuning procedures by controller manufacturers and process industry [Åström and Hägglund, 1995]. In both of these methods the criterion for optimization was the minimization of

$$\int_0^\infty |e(t)| dt$$

where  $e(t)$  is the tracking error due to a step input. A more modern approach to gain selection is to use a model based technique like the one we discuss in Section 7.3.

### 7.2.1 Step response method

This method requires that the plant be open-loop stable.

**Assumption 7.2.1.** The plant  $P(s)$  in Figure 5.6 is open-loop stable. ◀

The tuning process works as follows.

1. Obtain the step response of the plant which by Assumption 7.2.1 and the FVT converges to a constant value. The step response can be obtained either by simulating the plant model or, if we don't have a model to simulate, by running experiments on the physical system. Plot the step response as in Figure 7.10.
2. From the plot, extract the parameters  $\sigma$  and  $\tau$  shown in Figure 7.10. Physically  $\sigma$  is the maximum slope of the step response. The parameter  $\tau$  is the intersection between the tangent and the coordinate axis.

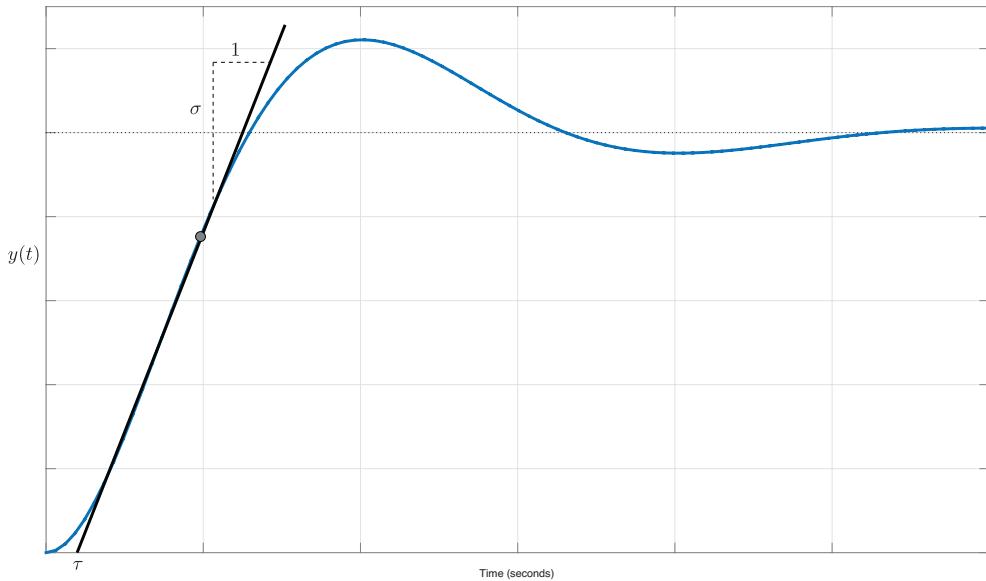


Figure 7.10: Obtaining the parameters  $\sigma$  and  $\tau$  for Ziegler-Nichols tuning of PID controllers.

3. Use the gain formulae in Table 7.1 to obtain a PID controller of the form 7.4.

Table 7.1: Ziegler-Nichols PID gains using step response method.

Controller type	$K_p$	$T_i$	$T_d$
Proportional	$\frac{1}{\sigma\tau}$	—	—
Proportional-Integral	$\frac{9}{10\sigma\tau}$	$3\tau$	—
Proportional-Integral-Derivative	$\frac{6}{5\sigma\tau}$	$2\tau$	$\frac{\tau}{2}$

**Example 7.2.1.** Consider the BIBO stable plant

$$P(s) = \frac{1}{(s+1)^3}$$

and its step response in Figure 7.11a. From that figure we obtain  $\sigma \approx 0.2705$ ,  $\tau \approx 0.806$ . Using the gain formula in Table 7.1 we obtain the PID controller

$$C(s) = 5.504 \left( 1 + 0.6203 \frac{1}{s} + 0.403s \right).$$

The closed-loop step response with this controller is shown in Figure 7.11b. ▲

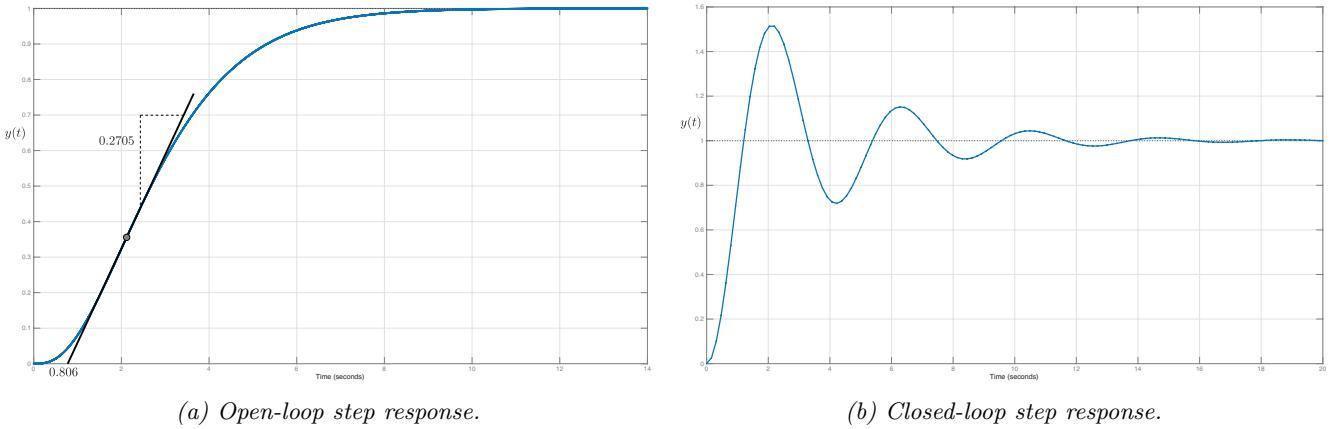


Figure 7.11: Ziegler-Nichols tuning for Example 7.2.1 using step response method.

## 7.2.2 Frequency response method

Consider the TF from  $r$  to  $y$  for the system in Figure 5.6 when the controller is  $C(s) = K_p$ , i.e., a proportional controller

$$\frac{Y(s)}{R(s)} = \frac{K_p P(s)}{1 + K_p P(s)}. \quad (7.10)$$

The method requires the following assumption.

**Assumption 7.2.2.** The closed-loop transfer function with proportional control (7.10) is BIBO stable for very small  $K_p$ . Furthermore, as  $K_p$  increases there is a critical value  $K_u$  at which two of the closed-loop poles lie on the imaginary axis while the rest remain in  $\mathbb{C}^-$ . ◀

Under Assumption 7.2.2, when  $K_p = K_u$  two of the closed-loop poles lie on the imaginary axis at, say,  $s = \pm j\omega_u$ . With  $C(s) = K_u$  the steady-state closed-loop step response will oscillate with period  $T_u = 2\pi/\omega_u$ . The parameters  $K_u$  and  $T_u$  are called, respectively, the **ultimate gain** and **ultimate period**. Once  $K_u$  and  $T_u$  have been obtained the Ziegler-Nichols PID gains can be computed using the entries in Table 7.2.

Table 7.2: Ziegler-Nichols PID gains using frequency response method.

Controller type	$K_p$	$T_i$	$T_d$
Proportional	$0.5K_u$	—	—
Proportional-Integral	$0.4K_u$	$0.8T_u$	—
Proportional-Integral-Derivative	$0.6K_u$	$0.5T_u$	$0.125T_u$

**Example 7.2.2.** We use the same plant as in Example 7.2.1. The closed-loop TF with proportional control was found in Section 7.1.1. Figure 7.3 shows that Assumption 7.2.2 holds for this system. To find  $K_u$  and  $T_u$  we can use the Routh table from Section 7.1.1. From that table we see that when the proportional gain equals 8, we get a row of zeros in the  $s^1$  row. The auxiliary polynomial (see Section 5.3.2) is

$$\pi_a(s) = 3s^2 + 9.$$

The roots of  $\pi_a$  are  $s = \pm j\sqrt{3}$ . From this we conclude that  $K_u = 8$ ,  $\omega_u = \sqrt{3}$  and  $T_u = 3.6276$ . The resulting PID controller using the Ziegler-Nichols gains from Table 7.2 is

$$C(s) = 4.8 \left( 1 + 0.5513 \frac{1}{s} + 0.4543s \right).$$

▲

**Exercise 7.6.** Simulate the closed-loop step response from Example 7.2.2.

### 7.3 Pole placement

The tuning methods of Section 7.2 are a good starting point for obtaining PID gains but they use too little information about the model of the plant. A more modern approach to PID gain selection is to choose the gains so that the closed-loop poles are located at desired places in the  $s$ -plane. This method of controller design is called **pole placement**. Pole placement is an extremely important concept in systems control and is covered in great detail in ECE488 and ECE481. In this section we do not cover pole placement in general; our scope is much more limited. Our objective is to use the ideas of pole placement to choose PID gains.

**Proposition 7.3.1.** Any controller  $C(s) \in \mathbb{R}(s)$  of the form

$$C(s) = \frac{g_2 s^2 + g_1 s + g_0}{s^2 + f_1 s} \tag{7.11}$$

is identical to the PID controller (7.6) with

$$K_p = \frac{g_1 f_1 - g_0}{f_1^2}, \quad T_i = \frac{g_1 f_1 - g_0}{g_0 f_1}, \quad T_d = \frac{g_0 - g_1 f_1 + g_2 f_1^2}{f_1(g_1 f_1 - g_0)}, \quad \tau_d = \frac{1}{f_1}. \tag{7.12}$$

**Exercise 7.7.** Prove Proposition 7.3.1.

**Assumption 7.3.2.** The plant  $P(s) \in \mathbb{R}(s)$  is strictly proper and second order

$$P(s) = \frac{b_1 s + b_0}{s^2 + a_1 s + a_0}, \quad b_0 \neq 0. \tag{7.13}$$

◀

**Remark 7.3.3.** We cannot allow  $b_0 = 0$  in the plant (7.13) because then the plant has a zero at  $s = 0$ . The PID controller (7.11) has a pole at  $s = 0$  due its integral action and if connected to a plant (7.13) with  $b_0 = 0$  there results an unstable pole-zero cancellation. See Definition 5.2.8 and Corollary 5.2.9.  $\blacklozenge$

If we connect the plant (7.13) in unity feedback configuration, as in Figure 5.6, with controller (7.11) then the characteristic polynomial (see Definition 5.2.6) is given by

$$\begin{aligned}\pi(s) &= (b_1 s + b_0)(g_2 s^2 + g_1 s + g_0) + (s^2 + a_1 s + a_0)(s^2 + f_1 s) \\ &= s^4 + (a_1 + f_1 + b_1 g_2)s^3 + (a_0 + a_1 f_1 + b_1 g_1 + b_0 g_2)s^2 + (a_0 f_1 + b_1 g_0 + b_0 g_1)s + b_0 g_0.\end{aligned}$$

Suppose that we would like the closed-loop poles, i.e., the roots of  $\pi(s)$ , to be located at<sup>5</sup>  $\{\lambda_1, \lambda_2, \lambda_3, \lambda_4\} \subset \mathbb{C}^-$ . These desired pole locations can be chosen based on, for instance, desired characteristics of the step response and the ideas of Section 4.3. The desired pole locations generate a *desired characteristic polynomial*

$$\pi_{\text{des}}(s) = (s - \lambda_1)(s - \lambda_2)(s - \lambda_3)(s - \lambda_4) =: s^4 + \alpha_3 s^3 + \alpha_2 s^2 + \alpha_1 s + \alpha_0 \in \mathbb{R}[s].$$

Equating coefficients between  $\pi$  and  $\pi_{\text{des}}$  yields the following linear equation in the unknown controller gains

$$\begin{bmatrix} 1 & b_1 & 0 & 0 \\ a_1 & b_0 & b_1 & 0 \\ a_0 & 0 & b_0 & b_1 \\ 0 & 0 & 0 & b_0 \end{bmatrix} \begin{bmatrix} f_1 \\ g_2 \\ g_1 \\ g_0 \end{bmatrix} = \begin{bmatrix} \alpha_3 - a_1 \\ \alpha_2 - a_0 \\ \alpha_1 \\ \alpha_0 \end{bmatrix}. \quad (7.14)$$

A few observations and facts about equation (7.14) and its derivation.

- If  $N_p$  and  $D_p$  are coprime, then equation 7.14 has a unique solution (see Proposition 7.3.4).
- In deriving (7.14) we have treated  $\tau_d$  as an independent design parameter as opposed to the approach taken in classical control, see Remark 7.1.1.
- The plant (7.13) includes every rational strictly proper system of order two. When designing PID controllers for higher order plants the ideas on model reduction in Section 4.5 can be useful.

**Proposition 7.3.4.** *Equation (7.14) has a unique solution if and only if the numerator and denominator of (7.13) are coprime.*

*Proof.* Let  $M$  denote the  $4 \times 4$  matrix in (7.14). Equation (7.14) has a unique solution if and only if the determinant of  $M$  is non-zero. The determinant of  $M$  is  $\det(M) = b_0(b_0^2 - b_0 b_1 a_1 + b_1^2 a_0)$ . Under Assumption 7.3.2  $b_0 \neq 0$  so  $\det(M) = 0$  if and only if  $b_0^2 - b_0 b_1 a_1 + b_1^2 a_0 = 0$ . We now proceed to prove the proposition.

( $\Rightarrow$ ) Assume that the numerator and denominator of (7.13) are *not* coprime. Then  $b_1 \neq 0$  because otherwise the plant has no zeros and therefore its numerator and denominator are coprime. When  $b_1 \neq 0$  the plant has a zero at  $s = -b_0/b_1$ . By assumption this is also a pole of the plant

$$\left(-\frac{b_0}{b_1}\right)^2 - a_1 \frac{b_0}{b_1} + a_0 = 0.$$

Simplifying this expression we obtain  $b_0^2 - b_0 b_1 a_1 + b_1^2 a_0 = 0$  which shows that  $\det(M) = 0$ . We've shown

$$\text{Not coprime} \Rightarrow \text{Solution to (7.14) not unique}$$

which is logically equivalent to showing<sup>6</sup>

$$\text{Coprime} \Leftarrow \text{Solution to (7.14) unique.}$$

( $\Leftarrow$ ) Conversely, suppose that  $\det(M) = 0$ . Then  $b_0^2 - b_0 b_1 a_1 + b_1^2 a_0 = 0$  which, reversing the argument we just made, implies that the plant's numerator and denominator have a common root.  $\square$

<sup>5</sup>The desired pole locations must be symmetric about the real axis to ensure that the controller gains are real.

<sup>6</sup>Formally, we have proven the **contrapositive** of the statement we sought to prove.

**Example 7.3.1.** Design a PID controller for the plant

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

which yields a closed-loop dominated by the roots of  $s^2 + 4s + 9$ . We solve this problem using our pole placement equations with

$$\pi_{\text{des}}(s) = (s^2 + 4s + 9)(s + 4)^2, \quad b_1 = 0, \quad b_0 = 2, \quad a_1 = 3, \quad a_0 = 2.$$

Note that we've added the extra term  $(s + 4)^2$  to  $\pi_{\text{des}}$  so that its order matches the order of  $\pi$ . Also note that the terms we've added are further to the left in  $\mathbb{C}^-$  so that  $s^2 + 4s + 9$  dominates the response. Equation 7.14 becomes

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 3 & 2 & 0 & 0 \\ 2 & 0 & 2 & 0 \\ 0 & 0 & 0 & 2 \end{bmatrix} \begin{bmatrix} f_1 \\ g_2 \\ g_1 \\ g_0 \end{bmatrix} = \begin{bmatrix} 12 - 3 \\ 57 - 2 \\ 136 \\ 144 \end{bmatrix}.$$

The unique solution is  $(f_1, g_2, g_1, g_0) = (9, 14, 59, 72)$ . Using Proposition 7.3.1 these gains correspond to a PID controller (7.6) with

$$K_p = 5.6667, \quad T_i = 0.7083, \quad T_d = 0.1634, \quad \tau_d = 0.1111.$$



**Exercise 7.8.** Design a PID controller for the plant from Example 7.2.1 using pole placement. Compare the step response of your controller to that in Figure 7.11b.

The ideas presented so far can also be applied to plants of the form

$$P(s) = \frac{K}{\tau s + 1} e^{-sT}.$$

Many plants, especially in process industries, can be satisfactorily described by this model. This plant is not rational because of the time-delay term  $e^{-sT}$ . One way to obtain an approximate rational second order model is to approximate the time delay by a first order all-pass system. In this case we have

$$P(s) = \frac{K}{\tau s + 1} e^{-sT} \approx \frac{K}{\tau s + 1} \frac{-\frac{sT}{2} + 1}{\frac{sT}{2} + 1}.$$

Our previously derived design equations (7.14) can now be used to design PID controllers for the approximated plant.

If the plant has a first order model

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

and we are using the PID controller (7.11) then the closed-loop characteristic polynomial is

$$\pi(s) = \tau s^3 + (\tau f_1 + K g_2 + 1) s^2 + (f_1 + g_1) s + g_0.$$

If our desired characteristic polynomial is

$$\pi_{\text{des}}(s) = s^3 + \alpha_2 s^2 + \alpha_1 s + \alpha_0 \in \mathbb{R}[s]$$

then the design equations become

$$\begin{bmatrix} \tau & K & 0 & 0 \\ 1 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} f_1 \\ g_2 \\ g_1 \\ g_0 \end{bmatrix} = \begin{bmatrix} \tau \alpha_2 - 1 \\ \tau \alpha_1 \\ \tau \alpha_0 \end{bmatrix}. \quad (7.15)$$

Equation (7.15) does not have a unique solution. Letting  $\alpha \in \mathbb{R}$  be a free parameter, the solution is

$$\begin{bmatrix} f_1 \\ g_2 \\ g_1 \\ g_0 \end{bmatrix} = \begin{bmatrix} 0 \\ \frac{1}{K}(\tau\alpha_2 - 1) \\ \tau\alpha_1 \\ \tau\alpha_0 \end{bmatrix} + \alpha \begin{bmatrix} 1 \\ -\frac{\tau}{K} \\ -1 \\ 0 \end{bmatrix}.$$

The resulting PID controller is

$$C(s) = \frac{\frac{1}{K}(\tau\alpha_2 - 1)s^2 + \tau\alpha_1 s + \tau\alpha_0 - \alpha s (\frac{\tau}{K}s + 1)}{s^2 + \alpha s}$$

where again  $\alpha \in \mathbb{R}$  is a free parameter.

### 7.3.1 When can PID control be used?

The discussion on pole placement shows that PID control is sufficient for processes where the dominant dynamics are second order. For such processes there are no benefits gained by using a more complex controller. If the plant's dynamics cannot be satisfactorily modelled by a strictly proper second order model, then there are benefits to using a higher order controller that is not a PID. We refer the interested student to [Åström and Hägglund, 1995, Example 3.3] for an example demonstrating this fact.

## 7.4 Integrator windup

Various modifications and improvements have been made to the classical PID control algorithm presented in this chapter. Most of these modifications are outside the scope of this introductory chapter. In this section we discuss one important modification that is very useful in practice.

All actuators have limitations: a motor has a maximum speed, an operational amplifier has a maximum voltage, a valve cannot be more than fully open or fully shut. For a control system with a wide range of operating conditions it may happen that the control variable reaches the actuator limits. When this happens the controller is no longer working in a linear regime: it can't react appropriately because it cannot change the control signal in one direction. If a controller with integral action is used, the error will continue to be integrated. This means that the integral term will become very large or, colloquially, it "winds up." It is then required that the error has the opposite sign for a long period of time before things return to normal. The result is that a controller with integral action may give large transients when the actuator saturates.

**Example 7.4.1.** Consider the system in Figure 7.12. Suppose that the plant and controller are given by

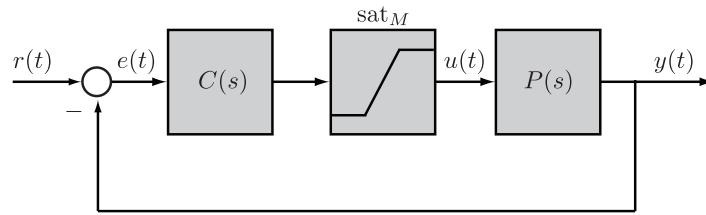


Figure 7.12: Unity feedback system with actuator limits.

$$P(s) = \frac{1}{s}, \quad C(s) = K_p + \frac{K_i}{s}.$$

The saturator block labelled  $\text{sat}_M$  in Figure 7.12 models the actuator limitations and its equation is

$$\text{sat}_M(x) = \begin{cases} x & \text{if } |x| \leq M, \\ \text{sgn}(x)M & \text{if } |x| > M. \end{cases} \quad (7.16)$$

For this example let  $M = 0.1$ , let the gains in the PI controller be  $K_p = K_i = 0.2$  and let  $r(t) = 1(t)$ . The simulation results are shown in Figures 7.13 and 7.14.

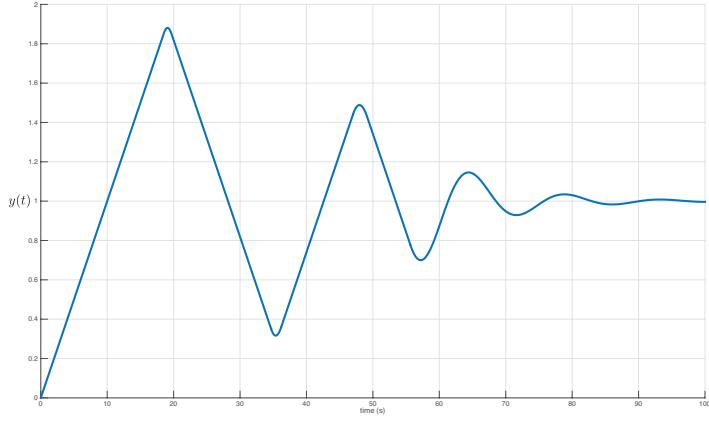
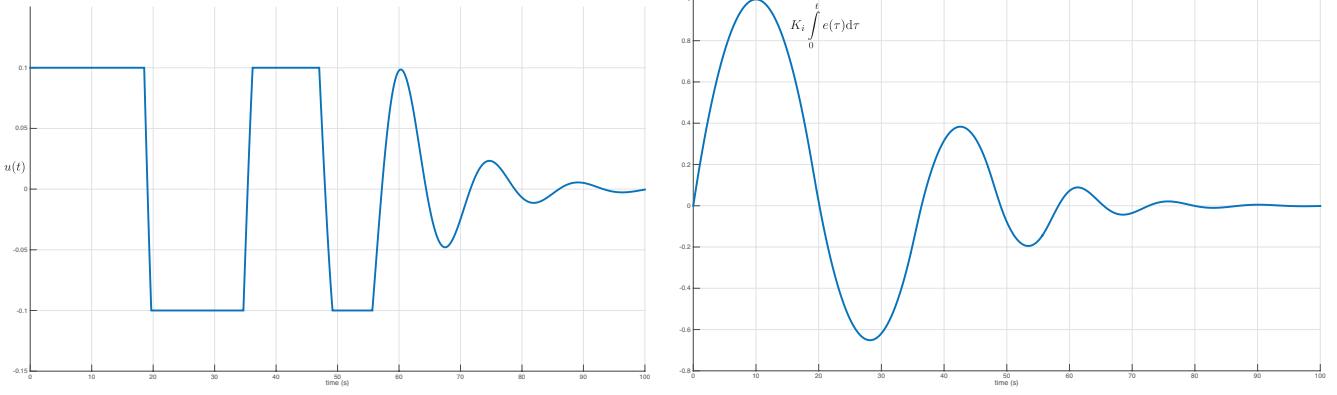


Figure 7.13: Plant output when there is integral windup.



(a) Control signal.

(b) Output of integrator.

Figure 7.14: Control signal and integrator signal from Example 7.4.1.

Let's go through these plots in detail to understand what is happening.

- At  $t = 0$  the reference signal equals 1. The large initial tracking error is enough to cause the proportional part of the control signal to exceed the actuator limitation.
- As a result, the control signal saturates at  $M = 0.1$  just after  $t = 0$ . This is seen in Figure 7.14a. The integral action starts integrating the tracking error and starts to increase. This is shown in Figure 7.14b.
- At  $t \approx 10$  the tracking error finally changes sign as  $y(t)$  finally exceeds 1. See Figure 7.13. As a result of this sign change, we see in Figure 7.14b that at  $t \approx 10$  the integrator starts to decrease.
- Even though the integrator starts to decrease at  $t \approx 10$ , it has accumulated a relatively large value over the first 10 seconds, i.e., it has wound up. The integral part of the control signal dominates the proportional part of the control signal. Hence the overall control signal in Figure 7.14a does not react and remains at its upper bound.
- At  $t \approx 18$  the tracking error becomes sufficiently large (see Figure 7.13) so that the proportional part of the control signal starts to dominate the accumulated integral. The control signal starts to decrease (Figure 7.14a) but the tracking error has become so large that it again saturates, this time at its lower bound, around  $t \approx 20$ .

- The above process begins to repeat except now the integrator value is negative. The magnitude of the tracking error at  $t \approx 20$  is less than it was just after  $t = 0$  so the integrator does not windup as much as it did initially.

The wild transients in Figure 7.13 are a tell-tale sign that windup is happening. ▲

The core idea used to protect systems against the negative effects of windup is to turn the integrator off whenever the input reaches a limit. This can either be done by a switch or by implementing the integrator in such a way that it automatically turns off when the input reaches a limit. An example of the latter type of solution, called **back-calculation**, is shown implemented on a PI controller in Figure 7.15. The system has an

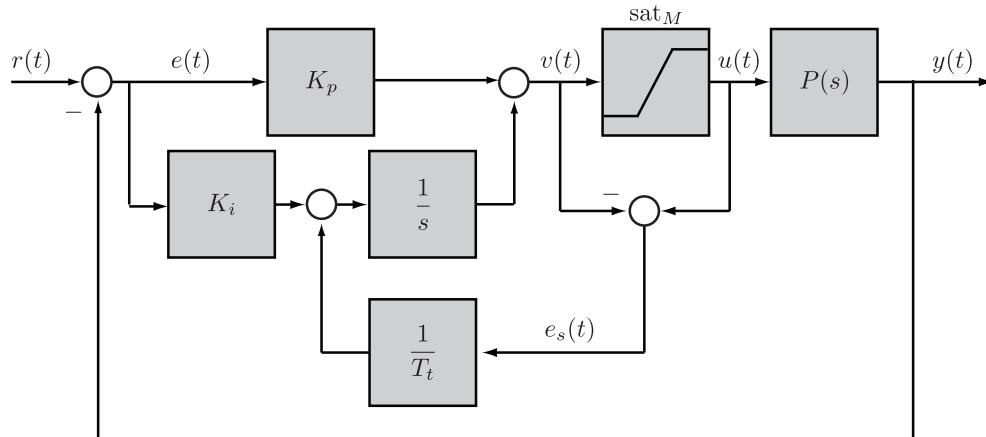


Figure 7.15: Anti-windup based on back-calculation for a PI controller.

extra feedback path that changes the input to the integrator. The error signal  $e_s$  is the difference between the actuator output  $u(t)$  and the controller output  $v(t)$ . This signal is fed through a gain of  $1/T_t$  into the integrator.

When there is no saturation  $e_s = 0$  and the system acts like a standard PI controller. When the actuator saturates, the signal  $e_s$  is fed back to the integrator in such a way that  $e_s$  goes toward zero. This implies that the controller output is kept close to the saturation limit. The controller output will then change as soon as the error changes sign and integral windup is avoided.

The rate at which the controller output is reset is governed by the feedback gain  $1/T_t$ ; a small value of  $T_t$  gives a short reset time. The parameter  $T_t$  is called the **tracking time constant**. It cannot be too small because measurement noise can then cause an undesirable reset. The tracking time constant  $T_t$  should be larger than  $T_d$  and smaller than  $T_i$  in a PID controller. A rule of thumb is  $T_t = \sqrt{T_i T_d}$  [Åström and Hägglund, 1995].

**Example 7.4.2.** Returning to the system from Example 7.4.1 we implement the back-calculation method shown in Figure 7.15 using  $T_t = \sqrt{T_i} = \sqrt{K_p/K_i} = 1$ . The simulation results are shown in Figure 7.16 and 7.17. The behaviour is very different from that in Example 7.4.1. In particular we see that the integrator output is kept relatively small when actuator limits are hit. ▲

**Exercise 7.9.** Re-create the simulation results from Example 7.4.2.

## 7.5 Summary

The purpose of this chapter was to introduce you to the basic structure of one of the most common feedback controllers used in industry. The key concepts to take away from this chapter are as follows.

- Know the ideal PID equation (7.1) (equivalently (7.4)).

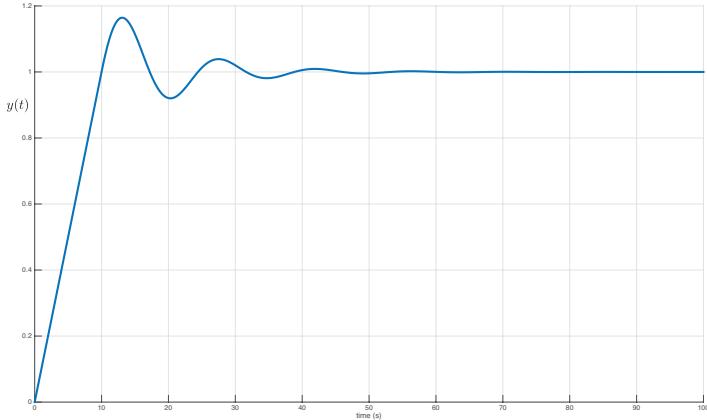
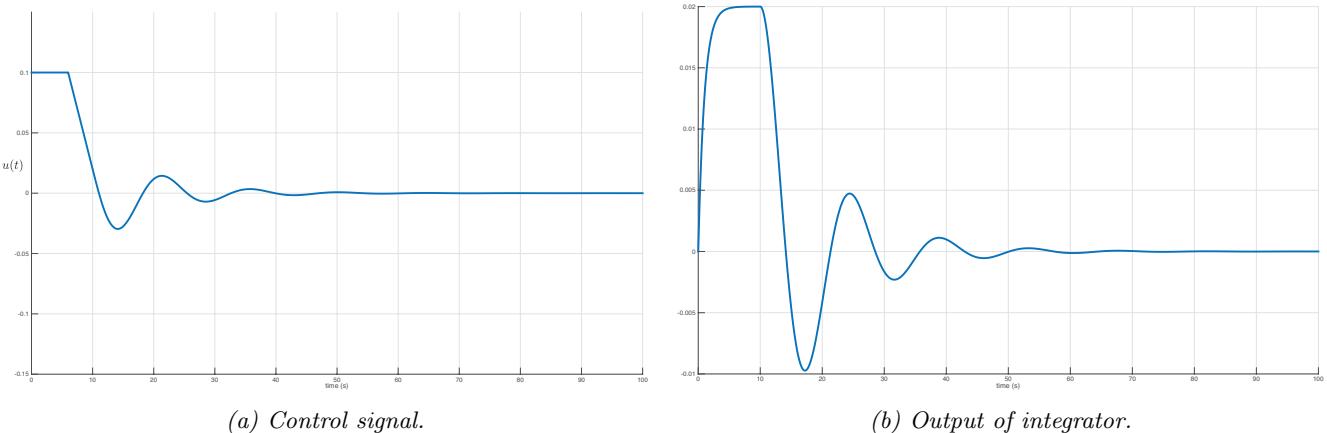


Figure 7.16: Plant output with anti-windup.



(a) Control signal.

(b) Output of integrator.

Figure 7.17: Control signal and integrator signal from Example 7.4.2.

- Understand the issues associated with implementing derivative action and the reasons that the low-pass approximation (7.5) is used in practice. You should also know the frequency ranges in which (7.5) gives a good approximation to a derivative.
- Understand why the two degree of freedom implementation scheme in Figure 7.1b is often better in practice.
- Some intuition about what the “P”, “I” and “D” terms do to a step response. Understand the reason why “D” is sometimes called a predictive mode.
- Be aware of the classical tuning methods of Ziegler-Nichols. You don’t need to know the specific formulae, you just need to understand the basic concept of using experimental results to choose controller gains.
- Section 7.3 is very important. You should know how to design PID controllers using pole placement and use this section to understand the power and the limitations of PID control.
- Know what integrator windup is and the core idea behind anti-windup solutions. Namely, to turn the integrator off whenever the input reaches a limit.

# Chapter 8

---

## Frequency domain methods for stability analysis

In Chapter 5 we developed tests that determine when a feedback system is stable. In this chapter we introduce a new test for stability of feedback systems called the Nyquist criterion [Nyquist, 1932]. The Nyquist criterion tells us if a feedback system is stable but more importantly, it provides guidelines for modifying an unstable system to make it stable. Nyquist's results were an inspiration for many of the design methods that today are called loop shaping. Nyquist's results also allow us to quantify how robust the feedback loop is to modelling uncertainty.

### Contents

8.1	Cauchy's principle of the argument . . . . .	166
8.2	Nyquist stability criterion . . . . .	169
8.3	Examples . . . . .	172
8.4	Stability margins . . . . .	176
8.5	Summary . . . . .	186

### 8.1 Cauchy's principle of the argument

The Nyquist criterion is a test for feedback stability. The criterion is based on the principle of the argument<sup>1</sup> from complex analysis [Wunsch, 2005]. The principle of the argument involves two things: a curve in the complex plane and a complex-valued function of a complex variable.

Consider a closed path (also called a closed curve or a contour) in the  $s$ -plane with no self-intersections and with negative, i.e., **clockwise (C.W.)** orientation as shown in Figure 8.1. We name the path  $\Gamma_s$ .

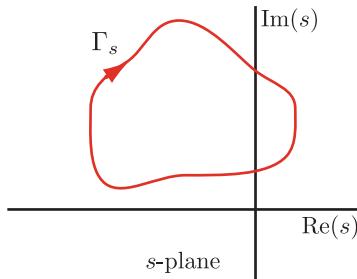


Figure 8.1: Closed path in  $s$ -plane with no self-intersections and negative orientation.

<sup>1</sup>Nyquist's original analysis was fairly complicated. Only later was it pointed out that the result can in fact be easily obtained from the principle of the argument from complex analysis [MacColl, 1945]. A more accessible version of Nyquist's papers are found in [Bellman and Kalaba, 1964], which also includes other interesting early papers on control. Nyquist's paper is also reprinted in an IEEE collection of seminal papers on control [Basar and Basar, 2001].

Now let  $G \in \mathbb{R}(s)$  be a rational function of the complex variable  $s$ . For every point  $s$  in the complex plane,  $G(s)$  is also a point in the complex plane. In other words, we can view the rational function  $G$  as a map  $\mathbb{C} \rightarrow \mathbb{C}$ . To make this clear and avoid clutter we draw two copies of the complex plane,  $s$  in one copy called the  $s$ -plane and  $G$  in the other copy called the  $G$ -plane.

If the curve  $\Gamma_s$  does not pass through a pole of  $G(s)$ , then as  $s$  moves around  $\Gamma_s$  from any starting point, the point  $G(s)$  traces out a different closed curve  $\Gamma_G$ , the image of  $\Gamma_s$  under the map  $G$ .

**Example 8.1.1.** For  $G(s) = s - 1$  we could have :

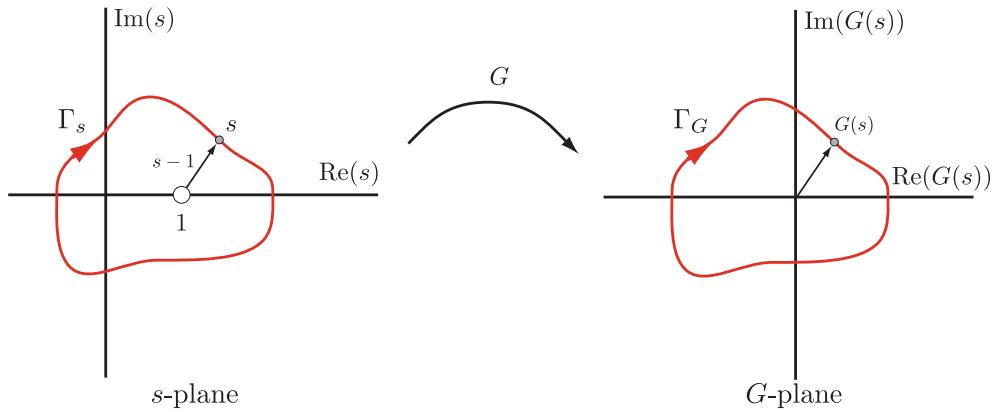


Figure 8.2: Mapping the curve  $\Gamma_s$  to the curve  $\Gamma_G$  when  $G(s) = s - 1$ . In this example each point  $s \in \Gamma_s$  is shifted to the left by 1 unit.

Notice that  $\Gamma_G$  is just  $\Gamma_s$  shifted to the left one unit. The angle of  $G(s)$  is  $\angle(s - 1)$ . Since  $\Gamma_s$  encircles one zero of  $G(s)$ , the angle of  $G$  changes by  $-2\pi$  as  $s$  makes a circuit around  $\Gamma_s$ . Therefore  $\Gamma_G$  encircles the origin once CW.  $\blacktriangle$

**Example 8.1.2.** We keep  $G(s) = s - 1$  but change  $\Gamma_s$ :

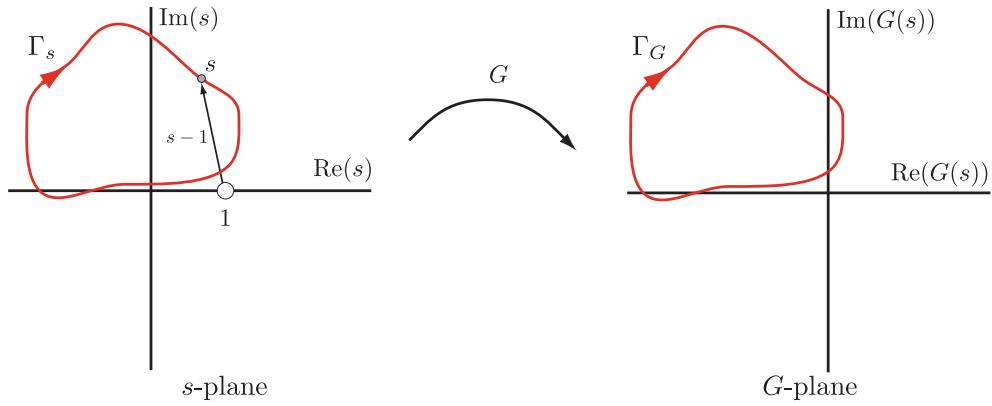


Figure 8.3: Mapping the curve  $\Gamma_s$  to the curve  $\Gamma_G$  when  $G(s) = s - 1$ . In this example,  $\Gamma_s$  does not encircle the zero of  $G$ .

Now  $\Gamma_s$  does not encircle any zeros of  $G$ . The net change in  $\angle(s - 1)$  is therefore zero after one trip around  $\Gamma_s$  which is why  $\Gamma_G$  has no encirclements of the origin.  $\blacktriangle$

**Example 8.1.3.** Now consider

$$G(s) = \frac{1}{s - 1}.$$

The angle of  $G(s)$  equals the negative of the angle of  $s - 1$

$$\angle(G(s)) = \angle(1) - \angle(s - 1) = -\angle(s - 1).$$

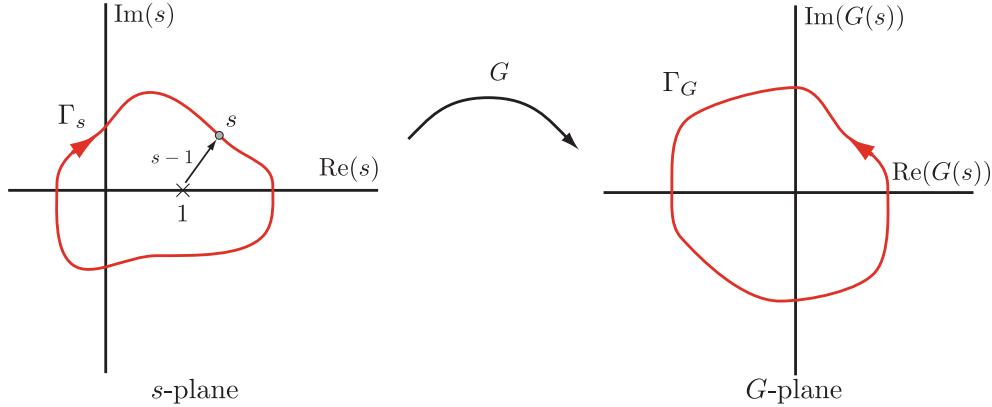


Figure 8.4: Mapping the curve  $\Gamma_s$  to the curve  $\Gamma_G$  when  $G(s) = (s - 1)^{-1}$ .

Since  $\Gamma_s$  encircles the point 1, i.e., a pole of  $G(s)$ , the curve  $\Gamma_G$  encircles the origin once in the **counterclockwise (C.C.W.)** direction. ▲

From these examples we observe that there is a relationship between the number of poles and zeros of  $G(s)$  inside  $\Gamma_s$  and the number of times  $\Gamma_G$  encircles the origin.

**Theorem 8.1.1** (Principle of the argument). *Suppose  $G(s) \in \mathbb{R}(s)$  has no poles or zeros on  $\Gamma_s$ , but  $\Gamma_s$  encloses  $n$  poles and  $m$  zeros of  $G(s)$ . Then  $\Gamma_G$  has exactly  $n - m$  counterclockwise encirclements of the origin.*

*Proof.* Write  $G(s)$  in this way

$$G(s) = K \frac{\prod_i (s - z_i)}{\prod_i (s - p_i)}$$

with  $K$  a real gain,  $\{z_i\}$  the zeros and  $\{p_i\}$  the poles. Then for every  $s$  on  $\Gamma_s$

$$\angle G(s) = \angle K + \sum \angle(s - z_i) - \sum \angle(s - p_i).$$

If  $z_i$  is enclosed by  $\Gamma_s$ , the net change after one trip around the curve  $\Gamma_s$  in the term  $\angle(s - z_i)$  is  $-2\pi$ ; otherwise the change is 0. If  $p_i$  is enclosed by  $\Gamma_s$ , the net change after one trip around the curve  $\Gamma_s$  in the term  $\angle(s - p_i)$  is  $-2\pi$ ; otherwise the change is 0. Hence the net change in  $\angle G$  equals  $m(-2\pi) - n(-2\pi)$  which equals  $(n - m)2\pi$ . □

**Remark 8.1.2.** If we change the orientation of  $\Gamma_s$  so that it is positively oriented (CCW) then the above proof yields  $m - n$  CCW encirclements. ♦

### 8.1.1 Nyquist contour

The special curve  $\Gamma_s$  we use is called the **Nyquist contour** shown in Figure 8.5. If we use a Nyquist contour then the resulting curve  $\Gamma_G$  is called the **Nyquist plot** of  $G(s)$ . If  $G(s)$  has no poles or zeros on  $\Gamma_s$ , then the Nyquist plot encircles the origin exactly  $n - m$  times CCW, where  $n$  equals the number of poles of  $G(s)$  in  $\text{Re}(s) > 0$  and  $m$  equals the number of zeros of  $G(s)$  in  $\text{Re}(s) > 0$ .

**Theorem 8.1.3.** *Suppose  $G(s)$  has no poles or zeros on the Nyquist contour  $\Gamma_s$ . Let  $n$  and  $m$  equal the number of, respectively, poles and zeros in  $\mathbb{C}^+$ . Then  $\Gamma_G$  has  $n - m$  CCW encirclements of the origin.*

A few observations to wrap up this section.

- With respect to Theorem 8.1.3,  $G(s)$  has no poles on the Nyquist contour if, and only if,  $G(s)$  has no poles on the imaginary axis and is proper.

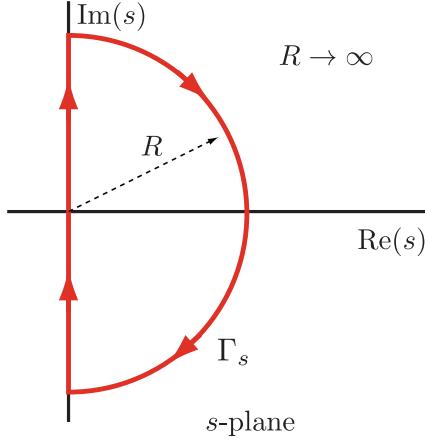
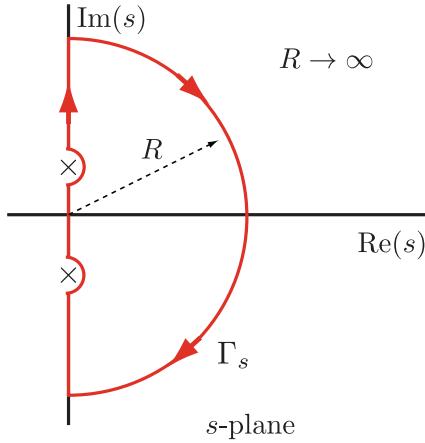


Figure 8.5: Basic Nyquist contour.

- Similarly,  $G(s)$  has no zeros on the Nyquist contour if, and only if,  $G(s)$  has no zeros on the imaginary axis and  $G(s)$  is not strictly proper.
- In our application, if  $G(s)$  does actually have poles on the imaginary axis, then we have to indent the contour around them. You can either indent to the left or the right; we'll always indent to the right as in Figure 8.6.

Figure 8.6: Nyquist contour indenting around poles on  $j\mathbb{R}$ .

- We indent around poles of  $G(s)$  on the imaginary axis because otherwise the function  $G(s)$  would be undefined there; we don't need to indent around the zeros of  $G(s)$ .

## 8.2 Nyquist stability criterion

Now we apply the principle of the argument to the problem of feedback stability. The setup is shown in Figure 8.7. Here  $K$  is a real gain and  $C(s)$ ,  $P(s)$  are rational transfer functions. The key observation is that if the feedback system is to be input-output stable then the poles of the closed-loop transfer function

$$\frac{KC(s)P(s)}{1 + KC(s)P(s)} \quad (8.1)$$

must all lie in  $\mathbb{C}^-$ . The idea is that we work with the TF  $1 + KC(s)P(s)$  to ascertain when the closed-loop system is stable.

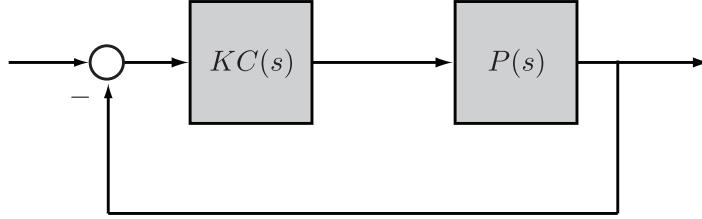


Figure 8.7: Unity feedback system.

**Definition 8.2.1.** The **loop gain** of the system in Figure 8.7 is the transfer function  $L(s) = KC(s)P(s)$ .

We could also have a TF  $H(s)$  in the feedback path in which case  $L(s) = KC(s)P(s)H(s)$  but we'll take  $H(s) = 1$  for simplicity.

**Assumption 8.2.2.** For the system in Figure 8.7

1.  $P(s)$  and  $C(s)$  are proper and  $C(s)P(s)$  is strictly proper.
2. The product  $C(s)P(s)$  has no unstable pole-zero cancellations.
3. The gain  $K$  is non-zero.

The second part of Assumption 8.2.2 is needed because the Nyquist criterion doesn't test for it, and such cancellations would make the feedback system not stable (see Corollary 5.2.9). The third part of Assumption 8.2.2 is needed only because we're going to divide by  $K$  at some point.

**Theorem 8.2.3.** Let  $n$  denote the number of poles of  $P(s)C(s)$  in  $\mathbb{C}^+$ . Construct the Nyquist plot of  $P(s)C(s)$ , indenting to the right around poles on the imaginary axis. Then the feedback system is stable if and only if the Nyquist plot doesn't pass through  $-\frac{1}{K} + j0$  and encircles it exactly  $n$  times CCW.

*Proof.* The closed-loop transfer function from reference input to plant output is (8.1). Define  $G(s) := 1 + L(s)$ . Because we have assumed no unstable pole-zero cancellations, feedback stability is equivalent to the condition that

$$G(s) = 1 + L(s) = 1 + KC(s)P(s) = \frac{D_c(s)D_p(s) + KN_c(s)N_p(s)}{D_c(s)D_p(s)}$$

has no zeros with  $\text{Re}(s) \geq 0$  (Theorem 5.2.11). So we're going to apply the principle of the argument to get a test for  $G(s)$  to have no zeros in the closed right half complex plane.

Please note the logic so far: The closed-loop transfer function from  $R$  to  $Y$  is

$$\frac{KP(s)C(s)}{G(s)}.$$

This should have no poles with  $\text{Re}(s) \geq 0$ . So  $G(s)$  should have no zeros with  $\text{Re}(s) \geq 0$ . So we need a test for  $G(s)$  to have this property.

Note that  $G(s)$  and  $C(s)P(s)$  have the same poles with real part greater than or equal to zero, so  $G(s)$  has precisely  $n$  there. Since  $\Gamma_s$  indents to the right around poles of  $G(s)$  on the imaginary axis and since  $G(s)$  is proper,  $G(s)$  has no poles on  $\Gamma_s$ . Thus by Theorem 8.1.3,  $G(s)$  has no zeros in  $\text{Re}(s) \geq 0$  and hence the feedback system is stable if, and only if, the Nyquist plot of  $G(s)$  doesn't pass through 0 and encircles it exactly  $n$  times CCW. Since  $C(s)P(s) = \frac{1}{K}G(s) - \frac{1}{K}$ , this latter condition is equivalent to: the Nyquist plot of  $P(s)C(s)$  doesn't pass through  $-\frac{1}{K}$  and encircles it exactly  $n$  times CCW.  $\square$

**Remark 8.2.4.** If there had been a non-unity transfer function  $H(s)$  in the feedback path, we would have taken  $G(s) = 1 + KP(s)C(s)H(s)$  in the proof. ♦

**Remark 8.2.5.** By Assumption 8.2.2  $C(s)P(s)$  is strictly proper. As a result, see Definition 2.8.1, the infinite part of the Nyquist counter  $\Gamma_s$  (labelled B in Figure 8.8) gets mapped to the origin. ♦

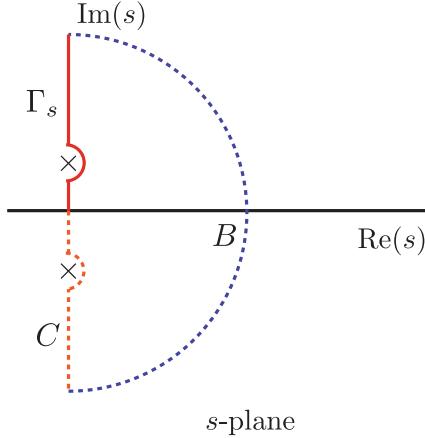


Figure 8.8: Relevant parts of Nyquist contour when plotting a Nyquist plot under Assumption 8.2.2.

**Remark 8.2.6.** Furthermore, see Remark 3.8.1, since  $L(s) \in \mathbb{R}(s)$  we have that  $|L(j\omega)| = |L(-j\omega)|$  and  $\angle L(j\omega) = -\angle L(-j\omega)$ . Therefore the part of the Nyquist plot corresponding to the negative imaginary axis in  $\Gamma_s$  (labelled C in Figure 8.8) is identical to the part corresponding to positive imaginary axis, except mirrored along the real axis. This means we can get a complete Nyquist plot by plotting  $C(s)P(s)$  for  $s$  on that portion of  $\Gamma_s$  in red in Figure 8.8. ♦

**Remark 8.2.7.** Together, Remarks 8.2.5, 8.2.6 show that under Assumption 8.2.2, the Nyquist plot is essentially (aside from indentations caused by poles on the imaginary axis) a polar plot of the loop gain with  $s = j\omega$ . As we saw in Section 3.8, the polar plot and the Bode plot of a TF are closely related. Thus we see the important relationship between the Nyquist plot and the Bode plot of the loop gain. ♦

#### Procedure to apply the Nyquist criterion

1. Pick the Nyquist counter  $\Gamma_s$  as in Figure 8.6 indenting to the right around any poles of  $C(s)P(s)$  on  $j\mathbb{R}$ .
2. Draw the image of  $\Gamma_s$  under the map  $C(s)P(s)$ . If Assumption 8.2.2 holds then we only need to draw the part corresponding the red portion of  $\Gamma_s$  in Figure 8.8, (see Remarks 8.2.5, 8.2.6).
3. Observe  $N$ , the number of counter clockwise encirclement of  $-\frac{1}{K}$  made by the Nyquist plot.
4. By the principle of the argument (Theorem 8.1.1)

$$N = n - m$$

where  $N$  is obtained from the figure in Step 3,  $n$  is the number of poles of  $C(s)P(s)$  in  $\mathbb{C}^+$  (known) and  $m$  is the number of closed-loop poles in  $\mathbb{C}^+$  (unknown).

5. By Theorem 5.2.7 the feedback system is stable if and only if  $m = 0$  which is equivalent, in light of Theorem 8.2.3, to  $N = n$ .

## 8.3 Examples

In this section you'll learn how to draw Nyquist plots and how to apply the Nyquist criterion.

**Example 8.3.1.** Let

$$C(s)P(s) = \frac{1}{(s+1)^2}.$$

In this case  $C(s)P(s)$  has no poles in  $\mathbb{C}^+$  and no poles on the imaginary axis. Hence we use the basic Nyquist counter shown in Figure 8.9. The left side of Figure 8.9 shows the resulting Nyquist plot.

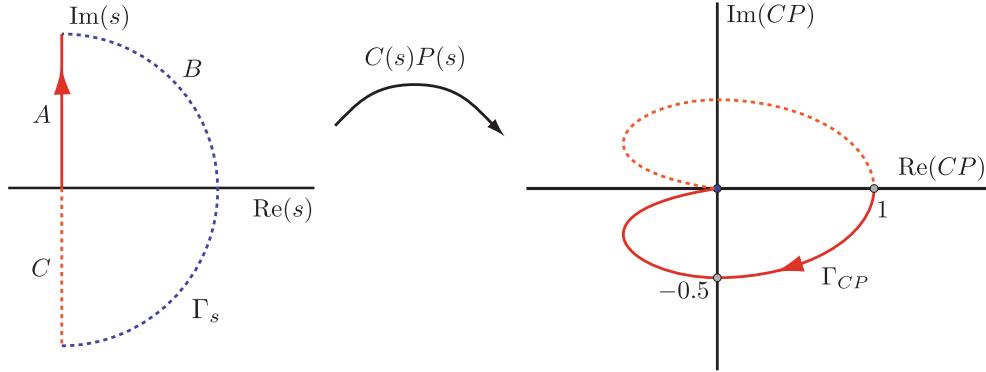


Figure 8.9: Nyquist contour (left) and Nyquist plot (right) for Example 8.3.1.

We map  $\Gamma_s$  one segment at a time. For the segment  $A$  (solid red) we have  $s = j\omega$  and

$$C(j\omega)P(j\omega) = \frac{1}{(1-\omega^2) + j2\omega} = \frac{1-\omega^2}{(1-\omega^2)^2 + 4\omega^2} - j \frac{2\omega}{(1-\omega^2)^2 + 4\omega^2}.$$

To plot the image of segment  $A$  we make the following observations which allow us to plot by hand the red portion of the Nyquist plot.

- On  $A$ ,  $s = j\omega$ . When  $\omega = 0$ ,  $C(j0)P(j0) = 1 + j0$ . As  $\omega$  starts to increase, the imaginary part of  $CP$  becomes negative and never crosses zero again for any finite value of  $\omega > 0$ .
- When  $\omega = 1$  the real part of  $CP$  equals zero, and hence switches from being positive to being negative. In other words, the Nyquist plot of the segment  $A$  crosses the  $j\mathbb{R}$  axis at  $C(j)P(j) = -j1/2$ . The real part of  $CP$  never crosses zero again for any finite value of  $\omega > 1$ .
- As  $\omega \rightarrow +\infty$  the real part of  $CP$  is negative and approaches zero. The imaginary part is also negative while approaching zero.

Segment  $B$  of the Nyquist counter (dashed blue in left side of Figure 8.9) gets mapped to the origin since  $CP$  is strictly proper, see Remark 8.2.5. The image of segment  $C$  (dashed orange in the Nyquist contour of Figure 8.9) is a reflection of the image of section  $A$ . Figure 8.10 shows the Nyquist plot generated by MATLAB for this system.

Now that we have the Nyquist plot, we are ready to apply the Nyquist stability criterion and determine for what range of  $K$  the feedback system is stable. The transfer function  $CP$  has no poles inside  $\Gamma_s$  and therefore  $n = 0$ . So the feedback system is stable if and only if the Nyquist plot encircles  $-1/K$  exactly 0 times CCW. This means, does not encircle it. The number of CCW encirclements for different values of  $-1/K$  are shown in Table 8.1

Thus the conditions for stability are  $-1/K < 0$  or  $-1/K > 1$ ; that is,  $K > 0$  or  $-1 < K < 0$ ; that is,  $K > -1$ ,  $K \neq 0$ . The condition  $K \neq 0$  is ruled out by our initial assumption (which we made only because

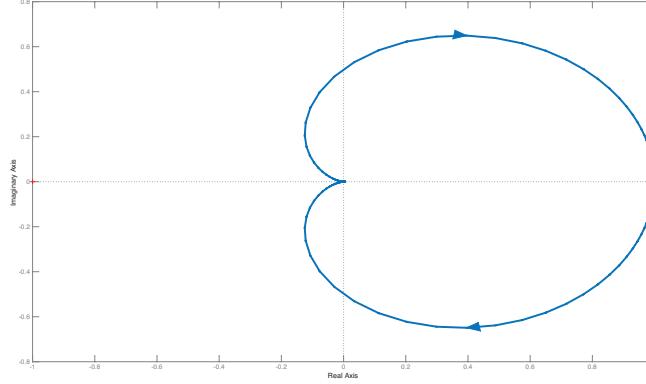


Figure 8.10: Nyquist plot generated by MATLAB for Example 8.3.1.

Table 8.1: Number of CCW for Example 8.3.1.

	$(-\infty, 0)$	$(0, 1)$	$(1, +\infty)$
CCW encirclements	0	-1	0

we were going to divide by  $K$ ). But now, at the end of the analysis, we can check directly that the feedback system actually is stable for  $K = 0$ . So finally the condition for stability is  $K > -1$ . You can readily confirm this by applying Routh-Hurwitz to the closed-loop characteristic polynomial,  $(s + 1)^2 + K$ .  $\blacktriangle$

**Example 8.3.2.** Now consider

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

for which

$$C(j\omega)P(j\omega) = \frac{j-\omega}{\omega(1-j\omega)} = \frac{-2}{1+\omega^2} + j \frac{1-\omega^2}{\omega(1+\omega^2)}.$$

Since there is a pole at the origin we have to indent to the right around it. We use the Nyquist contour  $\Gamma_s$  shown in Figure 8.11. The left side of Figure 8.11 shows the resulting Nyquist plot.

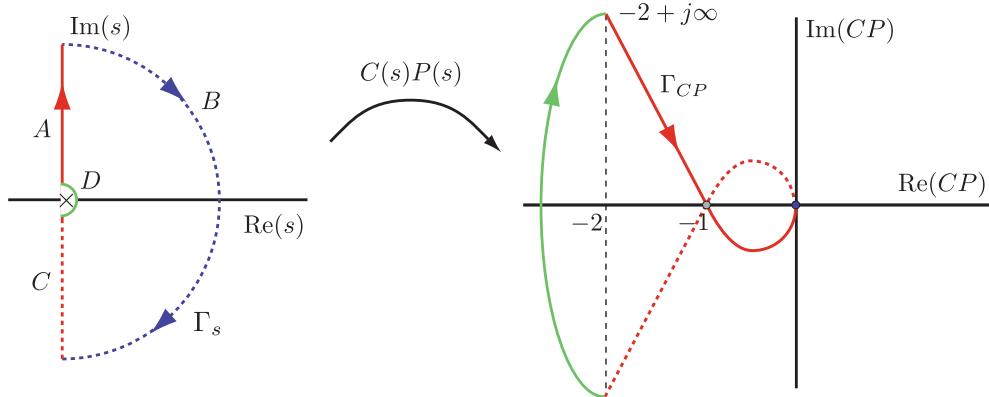


Figure 8.11: Nyquist contour (left) and Nyquist plot (right) for Example 8.3.2.

On segment  $A$ ,  $s$  goes from  $j\varepsilon$ ,  $\varepsilon > 0$  and small, to  $+j\infty$ . When  $s = j\varepsilon$  we have

$$C(j\varepsilon)P(j\varepsilon) = \frac{j-\varepsilon}{\omega(1-j\varepsilon)} = \frac{-2}{1+\varepsilon^2} + j \frac{1-\varepsilon^2}{\varepsilon(1+\varepsilon^2)} \approx -2 + j\infty.$$

This point is shown on the Nyquist plot in Figure 8.11. As  $s$  progresses along the rest of segment  $A$ , we have that the real part of  $C(j\omega)P(j\omega)$  remains negative while the imaginary part equals zero when  $s = j$ . Thus the Nyquist plot crosses the real axis at  $C(j)P(j) = -1 + j0$ . For  $\omega > 1$  both the real and imaginary parts remain negative and approach zero as  $\omega \rightarrow +\infty$ .

Since  $CP$  is strictly proper segment  $B$  gets mapped to the origin. The image of segment  $C$  is the same as the image of segment  $A$  reflected about the real axis.

On segment  $D$ ,  $s = \varepsilon e^{j\theta}$  where  $\varepsilon > 0$  is small and  $\theta$  increases from  $-\pi/2$  to  $\pi/2$ . Therefore

$$C(\varepsilon e^{j\theta})P(\varepsilon e^{j\theta}) = \frac{\varepsilon e^{j\theta} + 1}{\varepsilon e^{j\theta}(\varepsilon e^{j\theta} - 1)} \approx -\frac{1}{\varepsilon e^{j\theta}} = \frac{1}{\varepsilon} e^{j(\pi-\theta)}.$$

Thus the image of segment  $D$  is a semicircle of infinite radius whose argument goes from  $3\pi/2$  (equivalent to  $-\pi/2$ ) to  $\pi/2$ , i.e., CW. Figure 8.12 shows the Nyquist plot generated by MATLAB for this system.

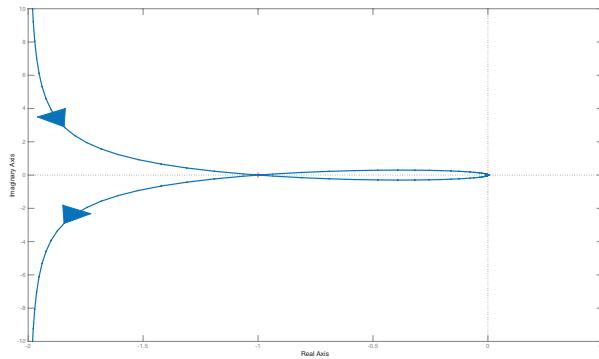


Figure 8.12: Nyquist plot generated by MATLAB for Example 8.3.2.

Now to apply the Nyquist criterion. Since  $C(s)P(s)$  has one pole inside  $\Gamma_s$ , we have  $n = 1$ . Therefore we need exactly 1 CCW encirclement of the point  $-1/K$ .

Table 8.2: Number of CCW for Example 8.3.2.

	$(-\infty, -1)$	$(-1, 0)$	$(0, +\infty)$
CCW encirclements	-1	1	0

Thus feedback stability holds if and only if  $-1 < 1/K < 0$ ; equivalently,  $K > 1$ . ▲

**Example 8.3.3.** In this example take

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

for which

$$C(j\omega)P(j\omega) = \frac{1}{(j\omega+1)(1-\omega^2)} = \frac{1-j\omega}{(\omega^2+1)(1-\omega^2)} = \frac{1}{1-\omega^4} - j \frac{\omega}{1-\omega^4}.$$

You should be able to draw the Nyquist plot shown in Figure 8.13 based on the previous two examples. We observe that on segment  $A$  of  $\Gamma_s$ ,  $s = j\omega$  with  $\omega \in [0, 1 - \varepsilon]$ ,  $\varepsilon > 0$  and small. On segment  $B$ ,  $s = j + \varepsilon e^{j\theta}$  with  $\varepsilon > 0$ , small and  $\theta \in [-\pi/2, \pi/2]$ . On segment  $C$  of  $\Gamma_s$ ,  $s = j\omega$  with  $\omega \in [1 + \varepsilon, +\infty)$ ,  $\varepsilon > 0$  and small. The drawing yields the data in Table 8.3. Since  $n = 0$ , i.e.,  $CP$  has no poles inside  $\Gamma_s$ , we conclude that the system is feedback stable if and only if  $-1/K > 1$ , i.e.,  $-1 < K < 0$ . ▲

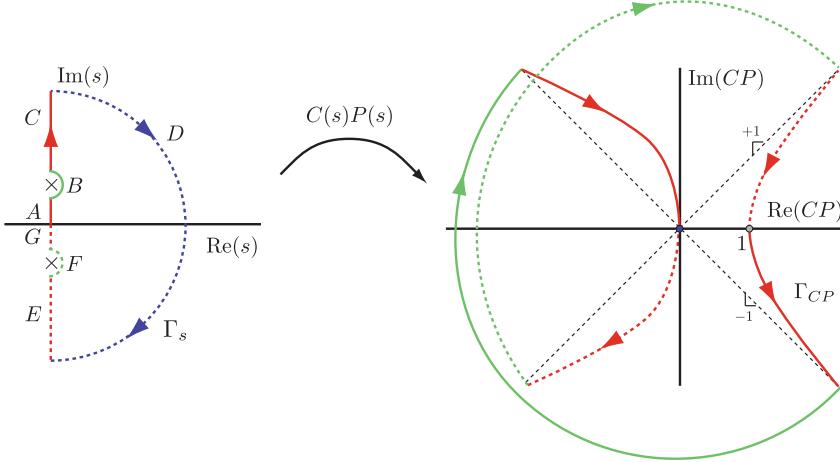


Figure 8.13: Nyquist contour (left) and Nyquist plot (right) for Example 8.3.3.

Table 8.3: Number of CCW for Example 8.3.3.

	$(-\infty, 0)$	$(0, 1)$	$(1, +\infty)$
CCW encirclements	-2	-1	0

**Remark 8.3.1.** This section and Section 3.8.2 have dealt with graphical methods done by hand on paper. Why did we bother? Why didn't we just ask MATLAB to draw the Bode or Nyquist plots? The answer is that by learning to draw the graphs, you acquire understanding. Additionally you should check that MATLAB incorrectly draws the Nyquist plot in Example 8.3.3. MATLAB isn't smart enough to indent the  $\Gamma_s$ -contour around the poles on the imaginary axis. ♦

**Example 8.3.4. (Capturing the essential features of the Nyquist plot from the Bode plot)** Consider the Bode plot in Figure 8.14 of some loop gain  $L(s)$ . In Figure 8.14 the magnitude is in absolute units (not dB), the phase is in degrees. The frequency is in rad/s.

We make the following observations:

- Since the magnitude plot is bounded from above, we conclude that the Nyquist contour doesn't need to indent around poles on the imaginary axis. Therefore we can use the basic Nyquist contour from Figure 8.5 and the Nyquist plot is simply a polar plot of  $L(j\omega)$ .
- Using the Bode plot, we see that at low frequencies ( $\omega < 10^{-2}$  rad/s) the magnitude is  $|L(j\omega)| \approx 5$ . The phase at these low frequencies is  $\angle L(j\omega) \approx -\pi \sim \pi$  and so  $L(j0) \approx -5$ . Thus the Nyquist plot starts at -5 in the complex plane.
- Next we observe from the Bode plot that the phase of  $L(j\omega)$  is always between  $-\pi$  and  $-\pi/2$ . Therefore the part of the Nyquist plot corresponding to moving up the imaginary axis is always in the third quadrant.
- The magnitude Bode plot crosses 1 at about  $\omega = 6$  rad/s and  $\angle L(j6) \approx -157^\circ$ . Therefore the Nyquist plot crosses the unit circle at an angle of  $-157^\circ$ .
- The magnitude Bode plot increases a little at first and then goes to zero as  $\omega \rightarrow \infty$ . Therefore the Nyquist plot moves away from the origin before eventually going to zero.

From the above observations we can sketch the Nyquist plot in Figure 8.15. The Nyquist plot encircles -1 once in the CCW direction. Thus the feedback system is stable if, and only if,  $L$  has one pole in  $\mathbb{C}^+$ . ▲

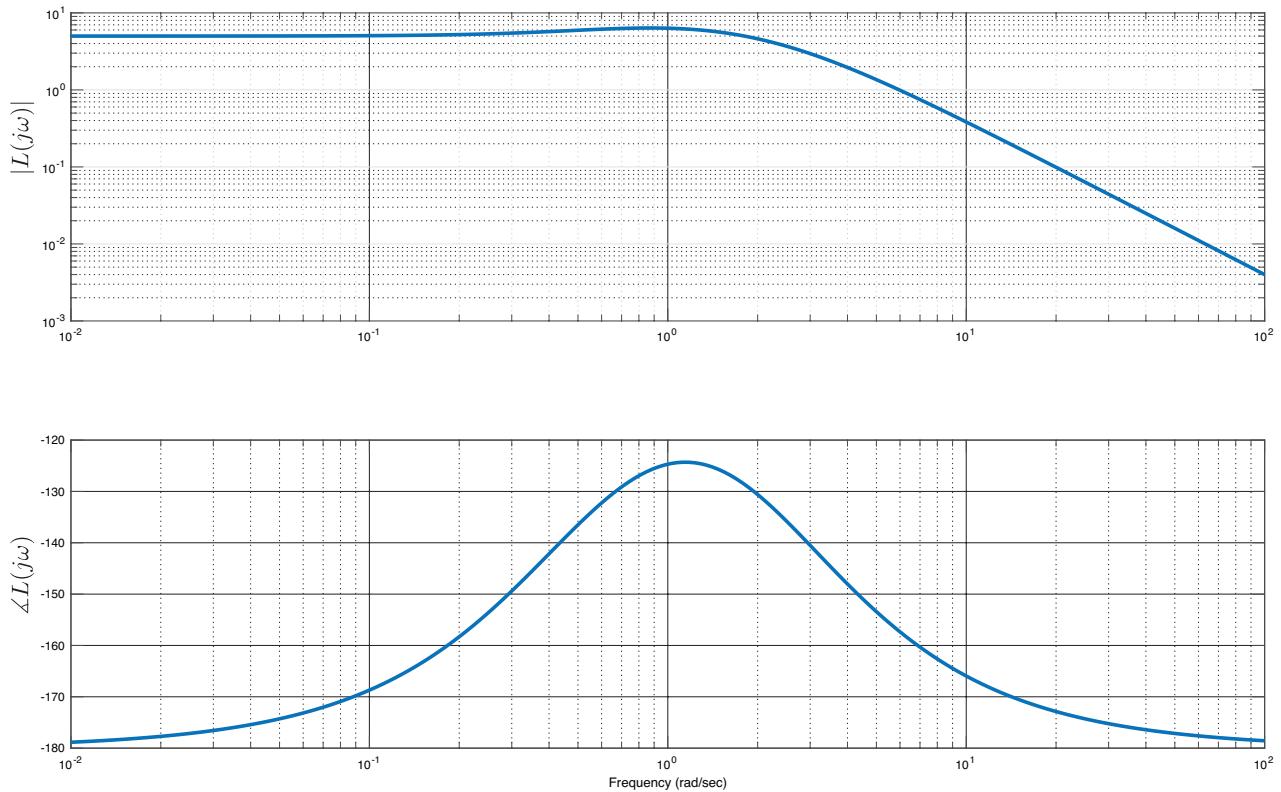


Figure 8.14: Bode plot.

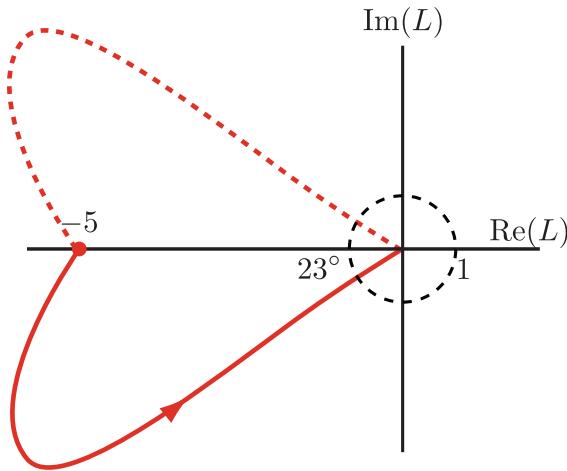


Figure 8.15: Nyquist plot estimated from Bode plot.

## 8.4 Stability margins

If a feedback system is stable, how stable is it? In other words, how far is it from being unstable? This depends entirely on our plant model, how we got it, and what uncertainty there is about the model. In the frequency domain context, uncertainty is naturally measured in terms of magnitude and phase as functions of frequency. In this section we look at this issue. The Nyquist plot is the most revealing, but we will also use the Bode plot.

### 8.4.1 Phase margin

**Example 8.4.1.** Consider the usual feedback loop as in Figure 8.7 and let its loop gain be given by

$$L(s) = \frac{2}{(s+1)^2}.$$

The Nyquist plot of  $L$  is shown in Figure 8.16 where the image of the portion of the Nyquist contour corresponding to negative frequencies is drawn with a red dashed line.

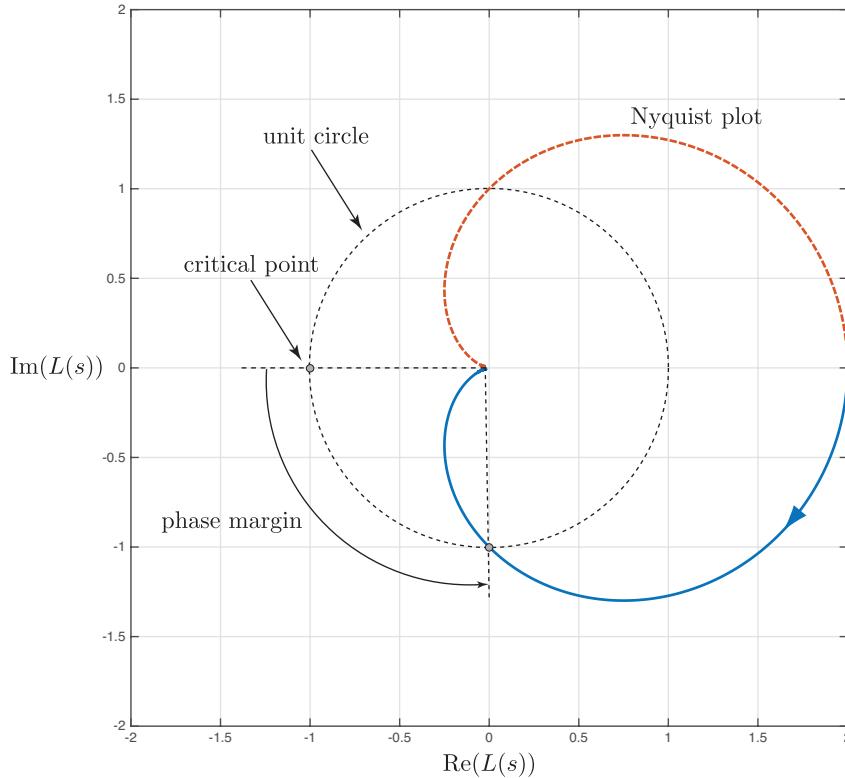


Figure 8.16: Nyquist plot of Example 8.4.1 with phase margin indicated.

There are no encirclements of the critical point  $-1 + j0$  so the feedback system is stable. A decrease in the phase of  $L(s)$  will cause the blue portion of the Nyquist plot to rotate CW about the origin. If we rotate it too far CW the Nyquist plot will eventually encircle  $-1 + j0$  and we'll lose stability. Hence, the maximum amount of phase we can decrease by is exactly the angle indicated in Figure 8.16. This angle is called the **phase margin** of the system and is related to the distance from the critical point  $-1 + j0$  to the point where the Nyquist plot crosses the unit circle.

The phase margin can also be found from the Bode plot of the loop gain because the two plots are closely related (see Remark 8.2.7 and Example 8.3.4). The Bode plot of  $L(j\omega)$  is shown in Figure 8.17. The Nyquist plot intersects the unit circle when  $|L(j\omega)| = 1$ . This corresponds to the magnitude plot crossing through  $20 \log |L(j\omega)| = 20 \log |1| = 0$  dB. Furthermore, the Nyquist plot intersects the negative real axis whenever<sup>2</sup>  $\angle(L(j\omega)) = \pi$ . This corresponds to the phase response crossing  $-180^\circ$ . So to find the phase margin from a Bode plot of  $L$ , we determine the frequency at which the magnitude is 0dB and then measure how far the phase at that frequency is from  $-180^\circ$ , see Figure 8.17. ▲

**Definition 8.4.1.** Let  $L(s) \in \mathbb{R}(s)$  be the loop gain TF for the feedback system in Figure 8.7.

- (i) A **gain crossover frequency**,  $\omega_{gc} \in [0, \infty)$ , for  $L(s)$  is a frequency at which  $|L(j\omega_{gc})| = 1$ .

<sup>2</sup>Remember that we take the angles  $\pi$  and  $-\pi$  to be equivalent. See Appendix 3.A.

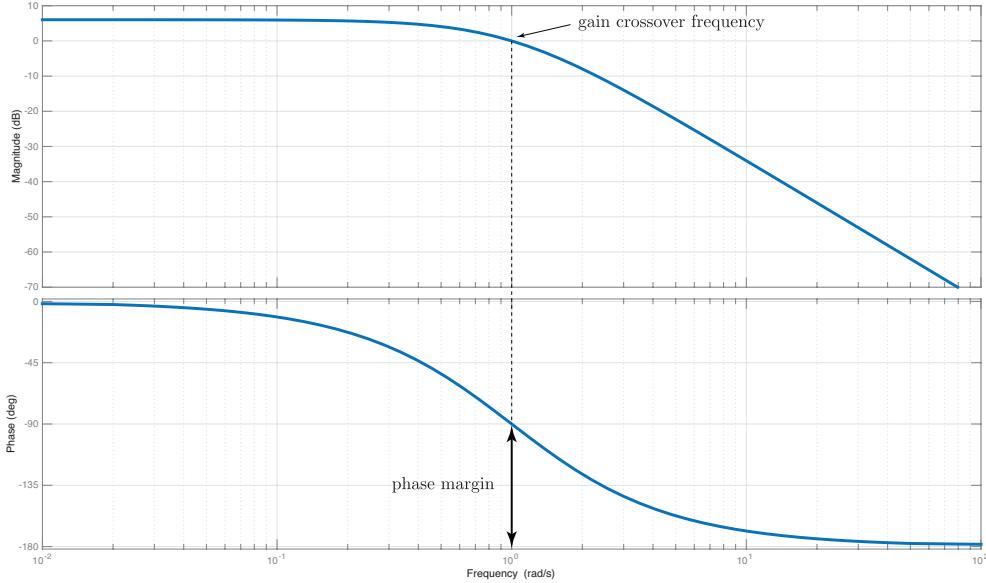


Figure 8.17: Bode plot of  $L(j\omega)$  for Example 8.4.1.

Let  $\omega_{gc,1}, \dots, \omega_{gc,\ell}$  be the gain crossover frequencies for  $L(s)$  and assume these are ordered so that

$$\angle(L(j\omega_{gc,1})) < \dots < \angle(L(j\omega_{gc,\ell})).$$

(ii) The **upper phase margin** of  $L(s)$  is

$$\Phi_{upm} := \begin{cases} 180^\circ + \angle(L(j\omega_{gc,1})), & \text{if } \angle(L(j\omega_{gc,1})) \leq 0 \\ \text{undefined,} & \text{else.} \end{cases} \quad (8.2)$$

(iii) The **lower phase margin** of  $L(s)$  is

$$\Phi_{lpm} := \begin{cases} 180^\circ - \angle(L(j\omega_{gc,\ell})), & \text{if } \angle(L(j\omega_{gc,\ell})) \geq 0 \\ \text{undefined,} & \text{else.} \end{cases} \quad (8.3)$$

In simpler terms: (1) the upper phase margin is the amount of phase *decrease* required to reach the stability limit and (2) the lower phase margin is the amount of phase *increase* required to reach the stability limit. The upper and lower phase margins are easy to read off a Nyquist plot or Bode plot so there is usually no need to compute them using the definition. See Figure 8.18. We adopt the convention that when we simply say *phase margin*, we refer to the smaller of the upper and lower phase margins and denote it by  $\Phi_{pm}$ .

**Remark 8.4.2.** In many texts one simply sees “phase margin” without reference to upper and lower. They are usually referring to the upper phase margin which is more relevant for closed-loop stability when the loop gain  $L(s)$  is minimum phase and has all its poles in  $\mathbb{C}^-$ . ♦

**Example 8.4.2.** Consider the system in Figure 8.7 with

$$C(s) = 14, \quad P(s) = \frac{-1}{s^2 + 4s + 15}$$

The Nyquist plot of  $CP$  is shown in Figure 8.19a. The loop gain has no poles inside  $\mathbb{C}^+$  and there are no

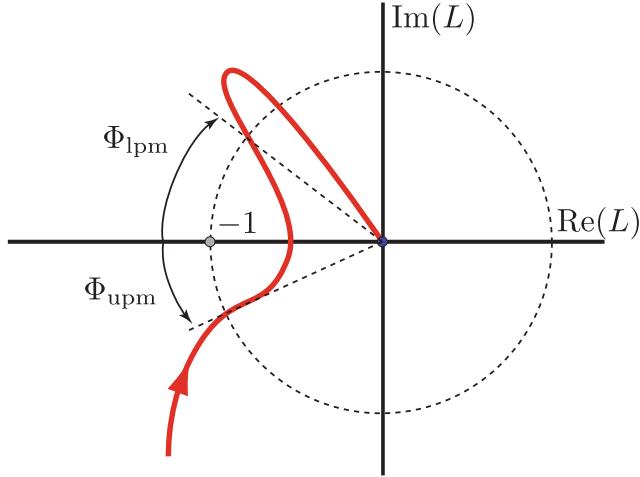


Figure 8.18: A Nyquist plot illustrating lower and upper phase margins.

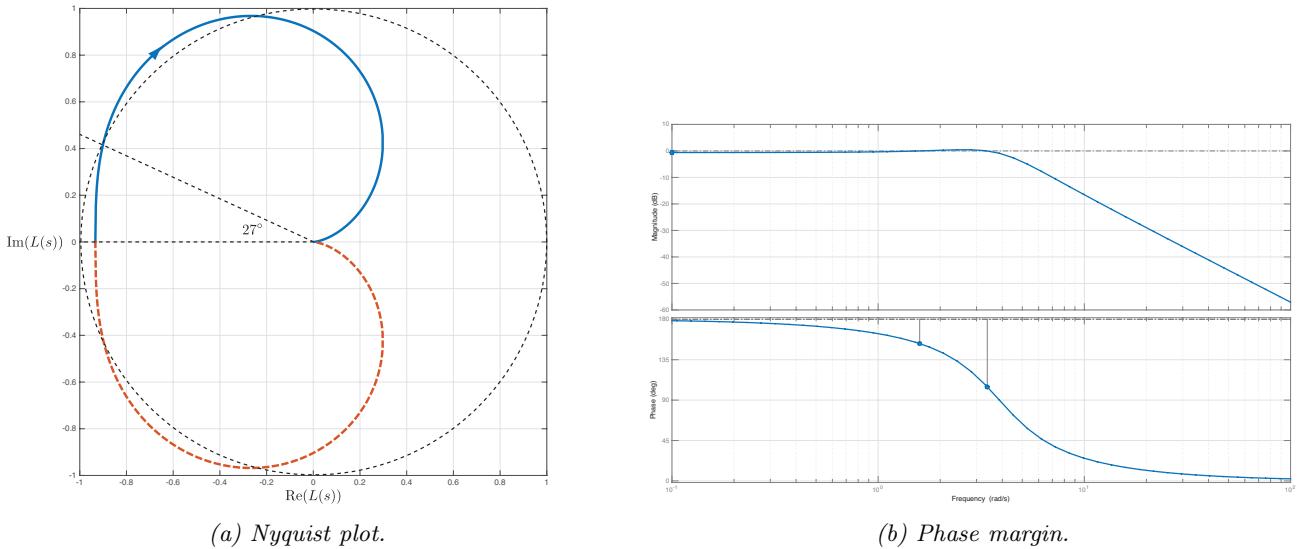


Figure 8.19: Phase margin from Nyquist plot in Example 8.4.2.

encirclement of the critical point  $-1 + j0$  so the closed-loop system is stable. We observe that the blue (non-dashed) portion of the Nyquist plot intersects the unit circle twice and that if we rotate in the CCW direction (adding phase) by  $27^\circ$  we get an encirclement of  $-1 + j0$  thereby losing closed-loop stability.

From the Bode plot in Figure 8.19 we determine that there are two gain crossover frequencies :  $\omega_{gc,1} = 3.37$  rad/s and  $\omega_{gc,2} = 1.59$  rad/s with (see Definition 8.4.1)

$$\angle(L(j\omega_{gc,1})) = 104.7^\circ < \angle(L(j\omega_{gc,2})) = 153^\circ.$$

Thus by Definition 8.4.1 we conclude that lower phase margin is  $\Phi_{lpm} = 27^\circ$  and the upper phase margin is undefined. MATLAB reports this as a negative phase margin of  $-27^\circ$ . Conclusion: We need the Nyquist plot for the correct interpretation of the phase margins.  $\blacktriangle$

## 8.4.2 Gain margin

**Example 8.4.3.** Consider the usual feedback loop as in Figure 8.7 and let

$$C(s) = 2, \quad P(s) = \frac{1}{(s+1)^2(0.1s+1)}.$$

The Nyquist plot of  $L$  is shown in Figure 8.20 where, again, the image of the portion of the Nyquist contour corresponding to negative frequencies is drawn with a dashed line.

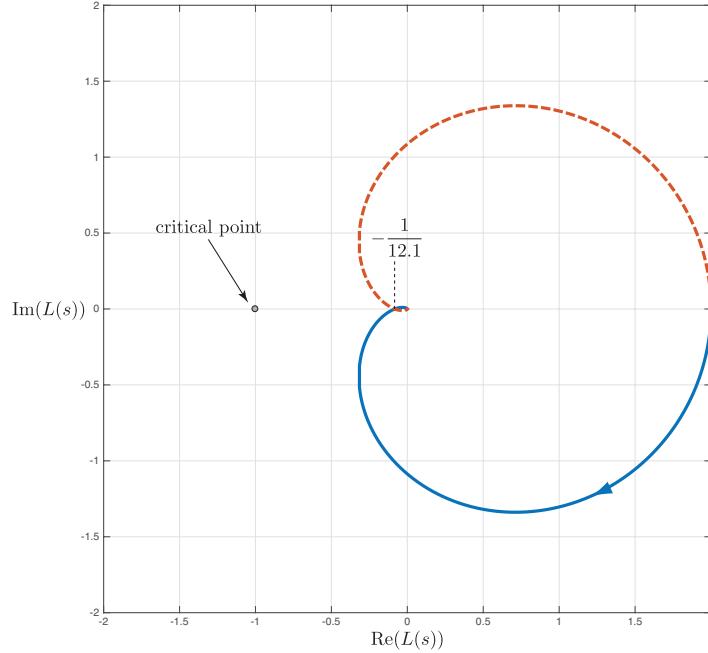


Figure 8.20: Nyquist plot of Example 8.4.3 with gain margin indicated.

There are no encirclements of the critical point  $-1 + j0$  so the feedback system is stable. The feedback loop will remain stable for  $KC(s)$ , where  $K \in \mathbb{R}$  is a gain, so long as the point  $-1/K$  isn't encircled. Since the Nyquist plot crosses the negative real axis at  $-1/(12.1)$ , this means we need

$$-\frac{1}{K} < -\frac{1}{12.1}.$$

Thus  $K$  can be increased from  $K = 1$  up to  $K = 12.1$  (21.67 dB) before the critical point is encircled resulting in instability. This number, 21.67 dB, is called the **gain margin**. It is the distance from the critical point  $-1 + j0$  to the point where the Nyquist plot crosses the negative real axis.

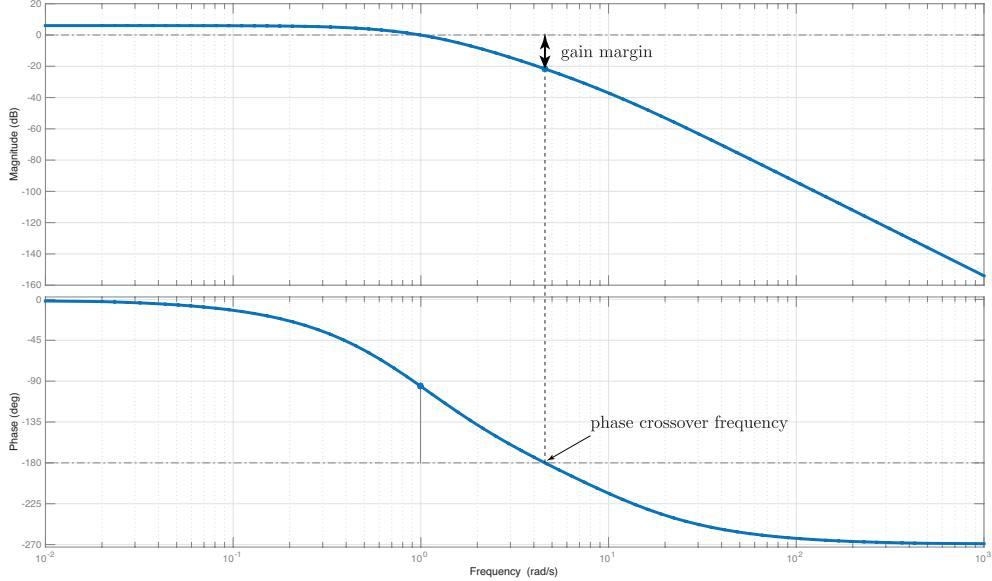
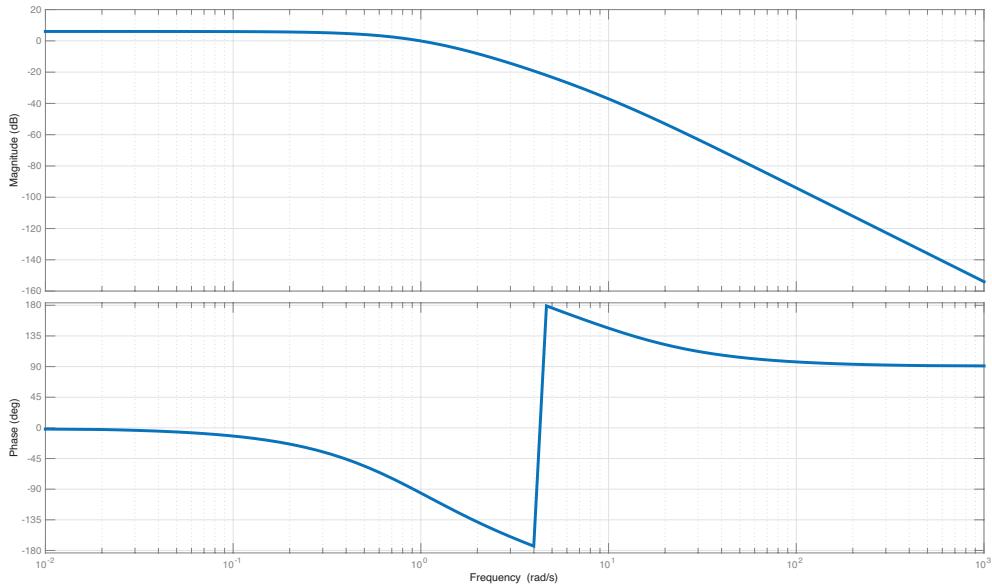
The gain margin can be found from the Bode plot of the loop gain. The Bode plot of  $L(j\omega)$  is shown in Figure 8.21. The Nyquist plot of  $L(j\omega)$  crosses the negative real axis whenever  $\angle(L(j\omega)) = \pi$ . Furthermore, the magnitude of  $L(j\omega)$  equals 0db if  $|L(j\omega)| = 1$ . So to find the gain margin from a Bode plot of  $L$ , we determine the frequency at which the phase is  $\pi$  and then measure how far the gain at that frequency is from 0 dB, see Figure 8.21. ▲

**Remark 8.4.3.** The Bode plot in Figure 8.21 was created using the MATLAB command `bode`. By default, this command “unwraps” the phase which is why the phase in Figure 8.21 goes below  $-180^\circ$ . Figure 8.22 shows the same Bode plot with phase wrapping on. Definitions 8.4.1 and 8.4.4 assume that phase wrapping is in effect. The MATLAB commands used to create Figure 8.22 are shown below.

```

1 s = tf('s');
2 L = 2/(s+1)^2/(0.1*s + 1);
3 opts = bodeoptions('cstprefs');
4 opts.PhaseWrapping = 'on';
5 bodeplot(L, opts);
6 grid on;

```

Figure 8.21: Bode plot of  $L(j\omega)$  for Example 8.4.3.Figure 8.22: Bode plot of  $L(j\omega)$  for Example 8.4.3 with phase wrapping.

**Definition 8.4.4.** Let  $L(s) \in \mathbb{R}(s)$  be the loop gain TF for the feedback system in Figure 8.7.

(i) A **phase crossover frequency**,  $\omega_{pc} \in [0, \infty)$ , for a loop TF  $L(s)$  is a frequency at which  $\angle L(j\omega_{pc}) = 180^\circ$ .

Let  $\omega_{pc,1}, \dots, \omega_{pc,\ell}$  be the phase crossover frequencies for  $L(s)$  and assume these are ordered so that

$$L(j\omega_{pc,1}) < \dots < L(j\omega_{pc,\ell}).$$

(ii) The **upper gain margin** of  $L(s)$  is

$$K_{ugm} := \begin{cases} \frac{1}{|L(j\omega_{pc,k+1})|}, & \text{if there exists a } k \in \{1, \dots, \ell - 1\} \text{ such that } L(j\omega_{pc,k}) < -1 < L(j\omega_{pc,k+1}). \\ \text{undefined,} & \text{if } L(j\omega_{pc,\ell}) < -1. \end{cases} \quad (8.4)$$

(iii) The **lower gain margin** of  $L(s)$  is

$$K_{\text{lgm}} := \begin{cases} \frac{1}{|L(j\omega_{\text{pc},k})|}, & \text{if there exists a } k \in \{1, \dots, \ell - 1\} \text{ such that } L(j\omega_{\text{pc},k}) < -1 < L(j\omega_{\text{pc},k+1}). \\ \text{undefined}, & \text{if } -1 < L(j\omega_{\text{pc},1}). \end{cases} \quad (8.5)$$

In simpler terms: (1) the upper gain margin is the smallest amount that the loop gain can be *increased* before the closed-loop system goes unstable and (2) the lower gain margin is smallest amount that the loop gain can be *decreased* before reaching the stability limit. As with phase margin, the upper and lower gain margins are easy to read off a Nyquist plot or Bode plot so there is usually no need to compute them using the definition, see Figure 8.23. We adopt the convention that when we simply say *gain margin*, we refer to the smaller of the upper and lower phase margins and denote it by  $K_{\text{gm}}$ .

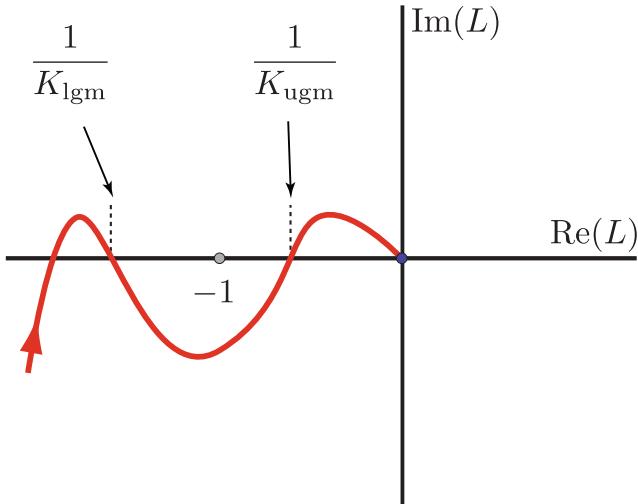


Figure 8.23: A Nyquist plot illustrating lower and upper gain margins.

**Remark 8.4.5.** In many texts one simply sees “gain margin” without reference to upper and lower. They are usually referring to the upper gain margin which is more relevant for closed-loop stability when the loop gain  $L(s)$  is minimum phase and has all its poles in  $\mathbb{C}^-$ . ♦

**Example 8.4.4.** Let

$$C(s) = 2, \quad P(s) = \frac{s + 1}{s(s - 1)}.$$

The Nyquist plot of  $CP$  is shown in Figure 8.24a (see Example 8.3.2). The critical point is  $-1 + j0$  and we need 1 CCW encirclement to ensure the feedback system is stable. The phase margin on the Nyquist plot is shown in Figure 8.24b

If  $KC(s) = 2K$  then the critical point is  $-1/K$  and  $K$  can be reduced from the value of 1 until  $-1/K = -2$ , i.e., the smallest allowable  $K$  is  $K = 0.5$  (6 dB). The Bode plot of this system generated with MATLAB is shown in Figure 8.25.

Using Definition 8.4.4 we conclude that the lower gain margin is 0.5 and the upper gain margin is undefined. MATLAB reports this as a negative gain margin of -6 dB. Conclusion: We need the Nyquist plot for the correct interpretation of the stability margins. ▲

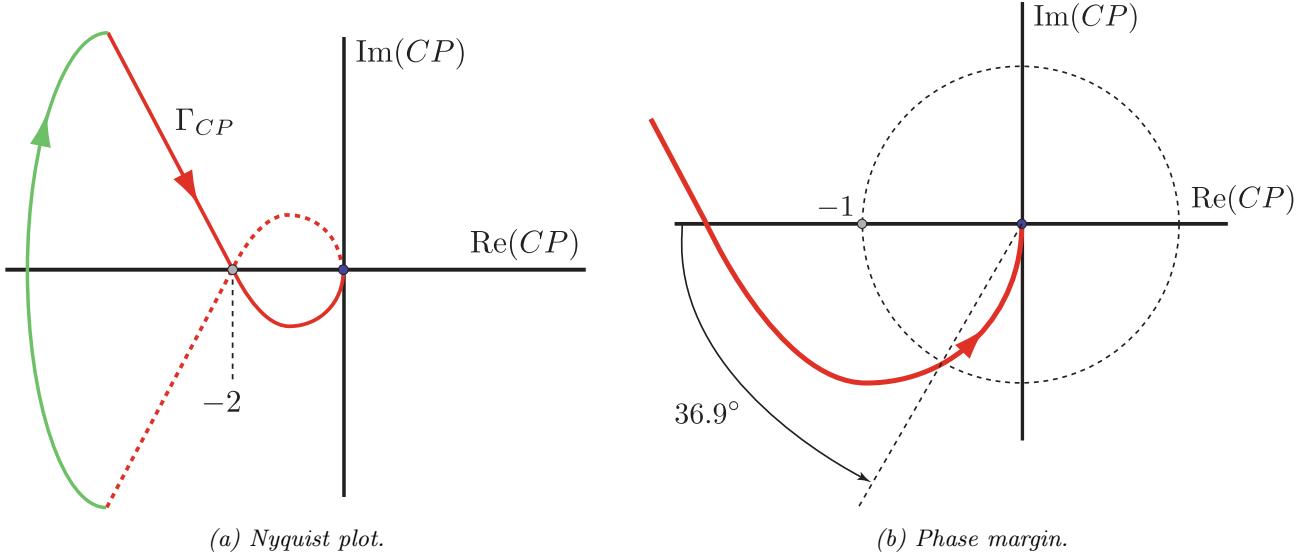
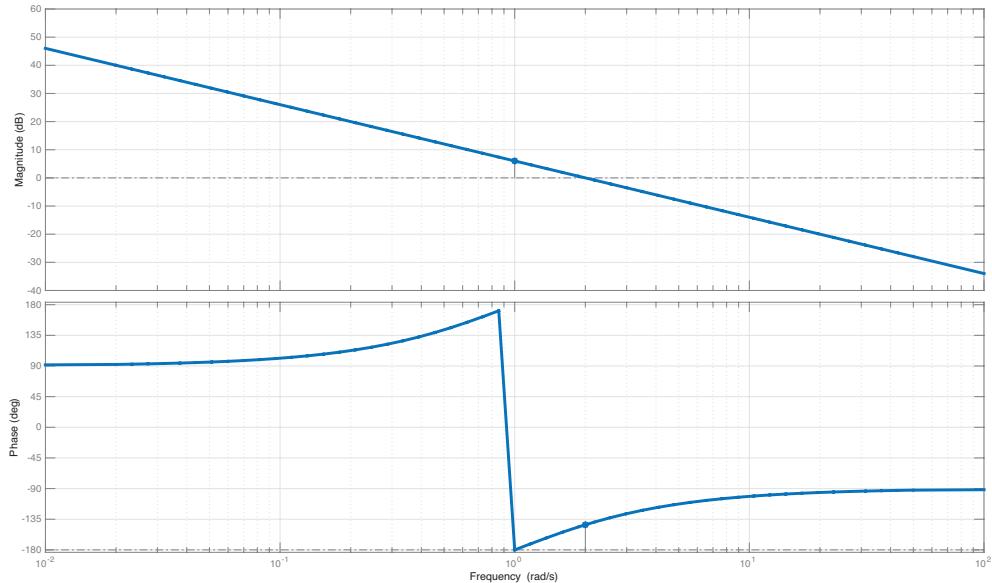


Figure 8.24: Phase margin from Nyquist plot in Example 8.4.4.

Figure 8.25: Bode plot of  $C(j\omega)P((j\omega))$  for Example 8.4.4.

### 8.4.3 Stability margin

Even if both the gain and phase margin are reasonable, the system may not be robust as illustrated by the following example.

**Example 8.4.5. (Good gain and phase margin but poor robustness)** Consider the loop gain

$$L(s) = \frac{0.38(s^2 + 0.1s + 0.55)}{s(s+1)(s^2 + 0.06s + 0.5)}.$$

The Nyquist and Bode plots for this system are shown in Figure 8.26.

From either plot we get the  $\Phi_{pm} \approx 70^\circ$  and  $K_{gm}$  is infinite. These values indicate that the system is robust, but the Nyquist curve is still close to the critical point, as shown in Figure 8.26a. This example illustrates the limitation of gain and phase margins when it comes to quantifying robustness to modelling uncertainty. ▲

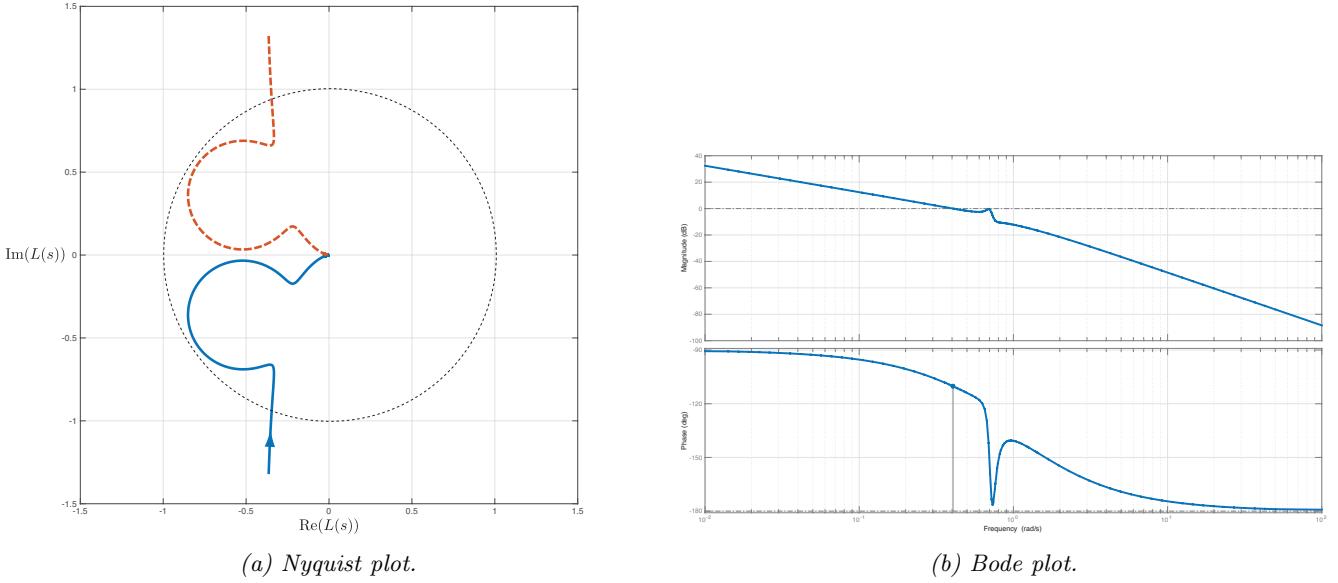


Figure 8.26: Nyquist and Bode plots for Example 8.4.5.

Let's recap: The phase margin is related to the distance from the critical point to the Nyquist plot along the unit circle; the gain margin is related to the distance from the critical point to the Nyquist plot along the real axis. More generally, it makes sense to define the **stability margin** to be the distance from the critical point to the closest point on the Nyquist plot.

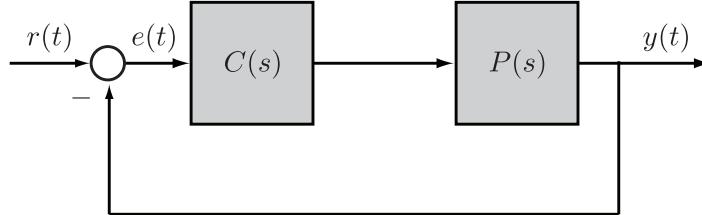


Figure 8.27: Unity feedback system.

Consider the system in Figure 8.27 and define  $S(s)$  to be the TF from the reference input  $r$  to the tracking error  $e$

$$S(s) := \frac{1}{1 + C(s)P(s)}. \quad (8.6)$$

Assume that the feedback system is stable. Then the distance from  $-1 + j0$  to the Nyquist plot of  $L(s) = C(s)P(s)$  can be expressed as follows

$$\begin{aligned} \min_{\omega} |-1 - L(j\omega)| &= \min_{\omega} |1 + C(j\omega)P(j\omega)| \\ &= \left[ \max_{\omega} |S(j\omega)| \right]^{-1} \\ &= \text{reciprocal of peak magnitude on Bode plot of } S(j\omega). \end{aligned}$$

In words: the point on the Nyquist plot which is closest to the point  $-1 + j0$  is a distance equal to the reciprocal of peak magnitude on Bode plot of  $S(j\omega)$ . Thus, to increase the stability margin, one may wish to make the function  $S(s)$  small.

**Definition 8.4.6.** Let  $L(s) \in \mathbb{R}(s)$  be the loop gain TF for the feedback system in Figure 8.7 and let  $S(s) = (1 + L(s))^{-1}$ . The **stability margin**, of the feedback system is

$$s_m := \left[ \max_{\omega} |S(j\omega)| \right]^{-1}.$$

**Example 8.4.6.** Let's return to the system from Example 8.4.5. The Bode plot of the transfer function

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

is shown in Figure 8.28.

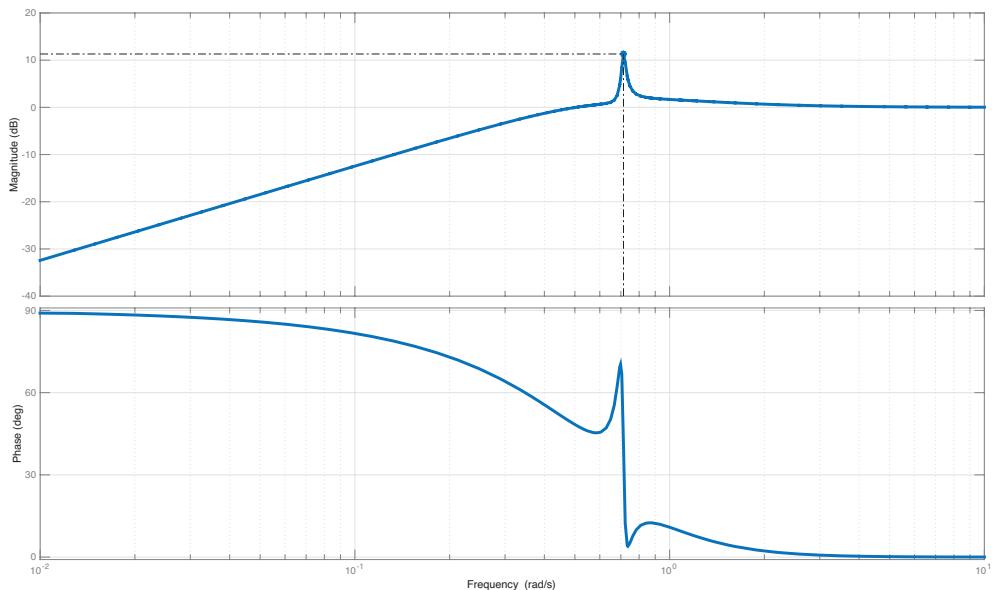


Figure 8.28: Bode plot of  $S(s) = (1 + L(s))^{-1}$  for Example 8.4.6.

From the magnitude plot of  $S(j\omega)$  we get that the peak response is 11.3 dB which means that  $\max_{\omega} |S(j\omega)| = 3.6728$ . The stability margin is therefore

$$\left[ \max_{\omega} |S(j\omega)| \right]^{-1} = \frac{1}{3.6728} = 0.2723$$

which is very low. ▲

There is a very nice geometric interpretation of the stability margin on the Nyquist plot. Suppose that, to ensure good robustness, we have a desired stability margin of  $s_m \geq M$ . This means that we want the Nyquist plot to be a distance greater than or equal to  $M$  from the critical point. In this case we want the Nyquist plot to appear as in Figure 8.29.

When designing feedback systems, it will often be useful to define the robustness of the system using gain, phase and stability margins. These numbers tell us how much the system can vary from our nominal model and still be stable. Reasonable values of the margins are shown in Table 8.4.

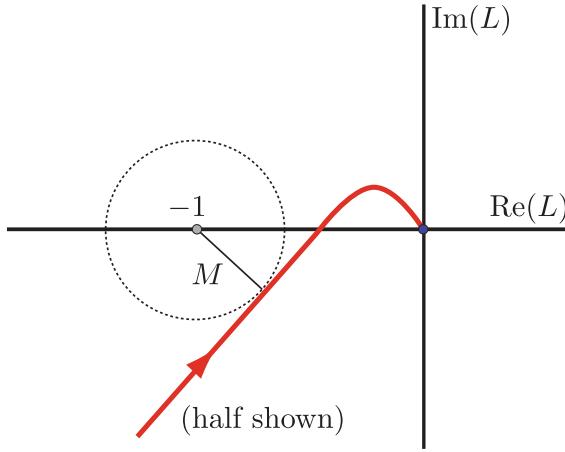


Figure 8.29: Geometric interpretation of a system with stability margin  $s_m \geq M$ .

Table 8.4: Reasonable values for stability margins.

Gain margin	Phase margin	Stability margin
6 – 13 dB	$30^\circ – 60^\circ$	0.5 – 0.8

## 8.5 Summary

This chapter is very important because the Nyquist plot and its interpretation lie at the heart of the design methods discussed in the next chapter. After completing this chapter you should know the following.

1. You should know what it means for a closed curve in the complex plane to have CW (negative) orientation and CCW (positive) orientation.
2. You should understand the basic ideas of Section 8.1 and especially the idea behind the construction of Nyquist plots: A Nyquist plot is the image of a curve in the complex plane under a complex-valued rational function.
3. Given a feedback system you should be able to find its loop gain TF  $L(s)$ .
4. You must be able to draw and, even more importantly, interpret a Nyquist plot.
5. You should completely understand every example in Section 8.3.
6. Stability margins tell us how far a system is from being unstable. This depends entirely on how much modelling uncertainty there is. For TFs, modelling uncertainty is naturally measured in terms of magnitude and phase as functions of frequency.
7. Understanding the meanings of gain and phase margin, in terms of feedback stability, is essential. You should be able to read gain and phase margins off Nyquist and Bode plots.
8. Know the definition of stability margin and be aware of the deficiency of using gain and phase margins alone. You should know how to obtain the stability margin from a Bode plot of (8.6).

# Chapter 9

---

## Introduction to control design in the frequency domain

In this chapter we develop the basic technique of controller design in the frequency domain. Specifications are in the form of bandwidth and stability margin. Of course our real interest is how the system will behave in physical time (e.g. settling time, overshoot, steady-state tracking error) and in using the frequency domain we are employing the duality between the two domains. The methods we use in this chapter fall under the broad heading of loop shaping, as the objective is to shape the Nyquist plot to have desired properties.

### Contents

9.1	Loop shaping . . . . .	187
9.2	Performance specifications . . . . .	189
9.3	Lag compensation . . . . .	191
9.4	Lead compensation . . . . .	198
9.5	Lead-lag compensation . . . . .	205
9.6	Loop shaping theory . . . . .	208
9.7	Summary . . . . .	211

### 9.1 Loop shaping

Consider a unity feedback system as shown in Figure 9.1.

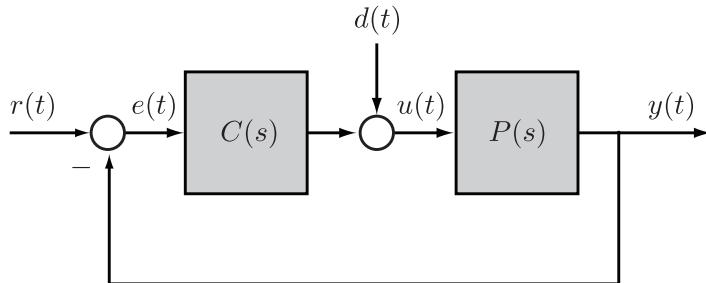


Figure 9.1: Unity feedback system.

The design problem is this: Given  $P$ , the nominal plant transfer function, maybe some uncertainty bounds, and some performance specifications design an implementable  $C$ . The performance specs would include, as a bare minimum, stability of the feedback system. The simplest situation is where the performance can be specified in terms of the transfer function

$$S(s) = \frac{1}{1 + C(s)P(s)}. \quad (9.1)$$

which is called the sensitivity function.

**Definition 9.1.1.** The **sensitivity function** of a feedback system with loop gain  $L(s)$  is  $S(s) = (1 + L(s))^{-1}$ .

**Remark 9.1.2.** The transfer function (9.1) is called the sensitivity function for the following reason. Let  $G(s)$  denote the transfer function from  $r$  to  $y$

$$G(s) := \frac{C(s)P(s)}{1 + C(s)P(s)}.$$

Consider the relative perturbation in  $G$  due to a relative perturbation in  $P$

$$\begin{aligned} \lim_{\Delta P \rightarrow 0} \frac{\Delta G/G}{\Delta P/P} &= \lim_{\Delta P \rightarrow 0} \frac{\Delta G}{\Delta P} \frac{P}{G} \\ &= \frac{dT}{dP} \cdot \frac{P}{T} \\ &= \frac{d}{dP} \left( \frac{C(s)P(s)}{1 + C(s)P(s)} \right) \cdot P \cdot \frac{1 + C(s)P(s)}{C(s)P(s)} \\ &= S(s). \end{aligned}$$

So  $S$  is a measure of the sensitivity of the closed-loop transfer function to variations in the plant transfer function. ♦

For us,  $S$  is important for two reasons: 1)  $S$  is the transfer function from the reference signal  $r$  to the tracking error  $e$ . Thus we want  $|S(j\omega)|$  to be small over the range of frequencies of  $r$  (see Example 5.4.4). 2) The peak magnitude of  $S$  is the reciprocal of the stability margin (see Definition 8.4.6). Thus a typical desired magnitude plot of  $S(j\omega)$  is shown in Figure 9.2.

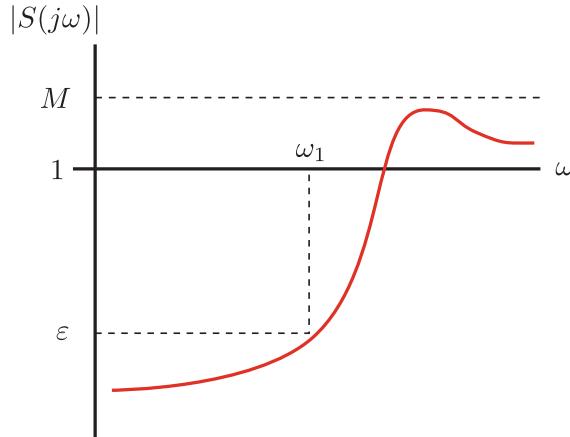


Figure 9.2: Typical desired shape for a sensitivity function.

Here  $\omega_1$  is the maximum frequency of the reference signal  $r$ ,  $\varepsilon < 1$ , and  $M$  is the maximum peak magnitude of  $|S|$ ,  $M > 1$ . If  $|S|$  has this shape and the feedback system is stable, then for the input  $r(t) = \cos(\omega t)$ ,  $\omega \leq \omega_1$ , we have  $|e(t)| \leq \varepsilon$  in steady state, and the stability margin is at least  $1/M$ . A typical value for  $M$  is 2 (see Table 8.4). In these terms, the design problem can be stated as follows: Given  $P$ ,  $M$ ,  $\varepsilon$ ,  $\omega_1$ ; design  $C$  so that the feedback system is stable and  $|S|$  satisfies  $|S(j\omega)| \leq \varepsilon$  for  $\omega \leq \omega_1$  and  $|S(j\omega)| \leq M$  for all  $\omega$ .

**Example 9.1.1.**

$$P(s) = \frac{10}{0.2s + 1}$$

This is a typical transfer function of a DC motor (cf. Example 2.8.5). Let's take a PI controller:

$$C(s) = K_p + \frac{K_i}{s}.$$

Then any  $M$ ,  $\varepsilon$ ,  $\omega_1$  are achievable by suitable  $K_p$ ,  $K_i$ . To see this, start with

$$\begin{aligned} S(s) &= \frac{1}{1 + \frac{10(K_p s + K_i)}{s(0.2s+1)}} \\ &= \frac{s(0.2s+1)}{0.2s^2 + (1 + 10K_p)s + 10K_i}. \end{aligned}$$

We can simplify by reducing to two equal real poles:

$$S(s) = \frac{5s(0.2s+1)}{(s+K)^2}.$$

Clearly, if we find a suitable  $K$ , then we can back-solve for  $K_p$ ,  $K_i$ . Figure 9.3 shows the Bode plot of  $S(j\omega)$  for  $K = 10$ .

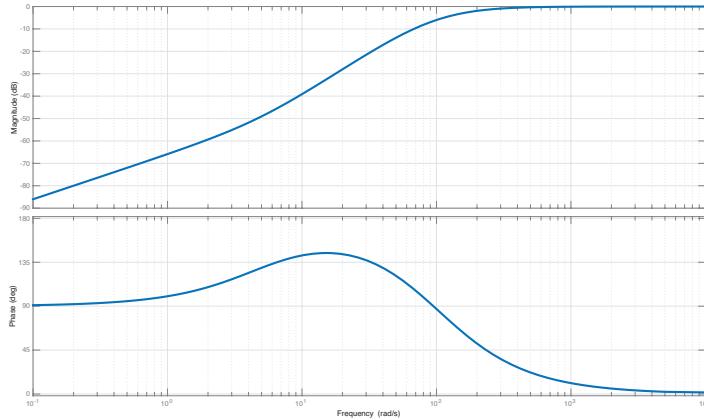


Figure 9.3: Bode plot of sensitivity function from Example 9.1.1.

Sketch the asymptotic Bode plot of  $S$  and confirm that any  $M > 1$ ,  $\varepsilon < 1$ , and  $\omega_1$  can be achieved. ▲

**Exercise 9.1.** Design a PI controller for the system in Example 9.1.1 for  $M = 2$ ,  $\varepsilon = 0.01$ , and  $\omega_1 = 50$  rad/s.

In practice it is common to combine interactively the shaping of  $S$  with a time-domain simulation. Now,  $S(s)$  is a nonlinear function of  $C(s)$ . So in fact it is easier to shape the loop transfer function  $L(s) = C(s)P(s)$  instead of  $S(s) = (1 + L(s))^{-1}$ . Notice that if the loop gain is high, i.e.,  $|L| \gg 1$ , then

$$|S| \approx \frac{1}{|L|}.$$

A typical desired loop gain amplitude plot is shown in Figure 9.4. In shaping  $|L|$  instead of  $|S|$ , we don't have a direct handle on the stability margin, unfortunately. However, we can directly influence the gain and phase margins, as we'll see.

## 9.2 Performance specifications

A key element of the control design process is how we specify the desired performance of the system. Specifications are often given in terms of robustness to modelling uncertainty and responses to reference signals and disturbances. They can be given in terms of both time and frequency responses. In this section we show how to convert time domain specifications into frequency domain specifications. We will do so in the same spirit as Chapter 4 by focusing on second order systems.

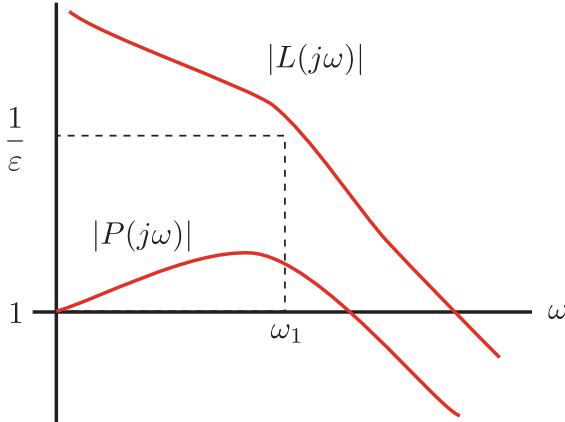


Figure 9.4: Typical desired shape for a loop gain. The loop gain is easier to work with for design than the sensitivity function because it is linear in the controller. High gain at low frequencies provides good tracking and disturbance rejection. Fast roll off to the gain crossover frequency gives good robustness. Low gain at high frequencies gives good rejection of high frequency noise.

### Relationship between damping ratio and phase margin

Consider the system in Figure 9.1 with

$$L(s) = \frac{\omega_n^2}{s(s + 2\zeta\omega_n)} \in \mathbb{R}(s). \quad (9.2)$$

The transfer function from  $r$  to  $y$  is

$$\frac{Y(s)}{R(s)} = \frac{\omega_n^2}{s^2 + 2\zeta\omega_n s + \omega_n^2}$$

which is the prototype second order system from Section 4.2 with damping ratio  $\zeta$ . To find the gain crossover frequencies we set  $|L(j\omega)| = 1$  and solve for  $\omega$  to obtain

$$\omega_{gc} = \omega_n \sqrt{\sqrt{1 + 4\zeta^4} - 2\zeta^2}. \quad (9.3)$$

Using Definition 8.4.1 (after some algebra) we get that the phase margin is

$$\Phi_{pm} = \arctan \left( \frac{2\zeta}{\sqrt{\sqrt{1 + 4\zeta^4} - 2\zeta^2}} \right). \quad (9.4)$$

Observe that the phase margin only depends on the damping ratio  $\zeta$  and not on  $\omega_n$ . Figure 9.5 shows a graph of phase margin  $\Phi_{pm}$  versus damping ratio  $\zeta$  and also shows that it can be reasonably well approximated by a line for small values of  $\zeta$ . We conclude that, for the system (9.2) we have

$$\text{If } 0 \leq \zeta \leq 0.7 \text{ then } \Phi_{pm} \approx 100\zeta \text{ (expressed in degrees).} \quad (9.5)$$

While the expression (9.5) was derived for the loop gain (9.2), we will use it as a general rule-of-thumb for converting overshoot specifications into phase margin specifications.

### Relationship between gain crossover frequencies and bandwidth

We saw in Chapter 4 that the speed of a system's response is closely tied to its bandwidth. Hence performance specifications like settling time and rise time can be cast as a specification on the closed-loop bandwidth  $\omega_{BW}$ . It is therefore useful to approximate the closed-loop bandwidth from the loop gain transfer function  $L(s)$ . If  $L(0)$  is finite, then usually  $\omega_{gc} \leq \omega_{BW} \leq 2\omega_{gc}$ . For design purposes we employ the following useful rule-of-thumb

$$\omega_{BW} \approx \omega_{gc}. \quad (9.6)$$

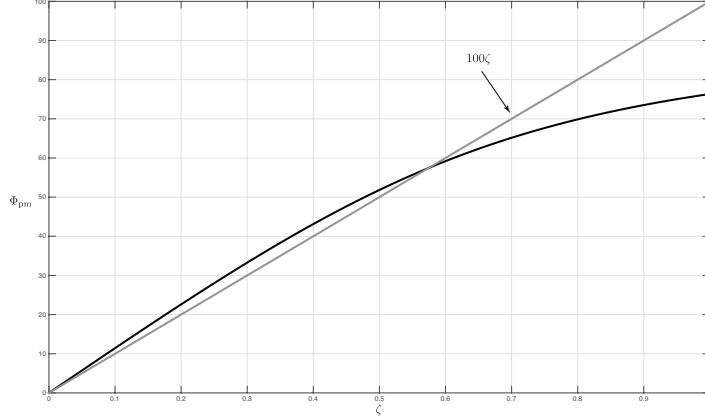


Figure 9.5: Phase margin  $\Phi_{pm}$  as a function  $\zeta$  for (9.2) (black) and its linear approximation (grey).

**Remark 9.2.1.** In the next two sections we present two simple loop shaping controllers. These loop shaping controllers work best for “nice plants” that (i) are stable or, at worst, have a single pole at  $s = 0$  and (ii) have only one gain crossover frequency  $\omega_{gc}$  and only one phase crossover frequency  $\omega_{pc}$ . ♦

### 9.3 Lag compensation

We separate the controller into two parts  $K$  and  $C_1(s)$  as shown in Figure 9.6. The controller is

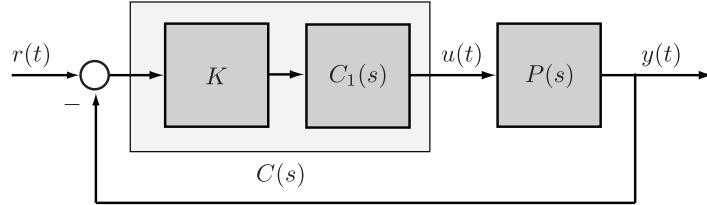


Figure 9.6: Lag controller block diagram.

$$C(s) = KC_1(s) = K \frac{\alpha Ts + 1}{Ts + 1}, \quad 0 < \alpha < 1, \quad T > 0, \quad K > 0, \quad (\text{lag controller}). \quad (9.7)$$

The lag controller has a steady-state gain of  $C(0) = KC_1(0) = K$  and its pole and a zero locations are shown in Figure 9.7. The asymptotic Bode plot of  $C_1(s)$  is shown in Figure 9.8. A lag controller is like a low pass filter.

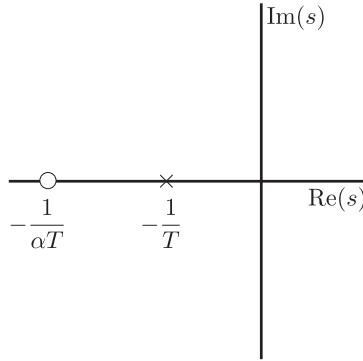


Figure 9.7: Pole-zero configuration of a lag controller: the pole is always closer to the imaginary axis than the zero.

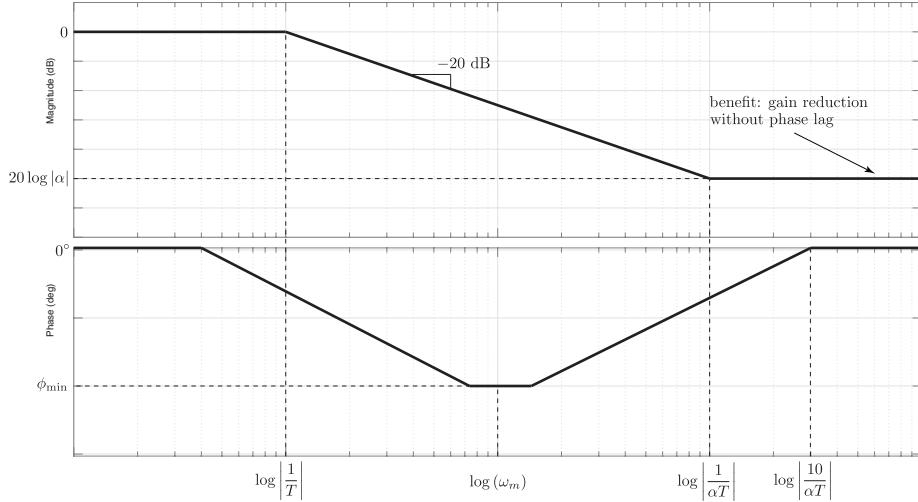


Figure 9.8: Asymptotic Bode plot of a lag controller with  $K = 1$ .

**Exercise 9.2.** Prove that magnitude plot of  $C_1(s)$  levels off at  $20 \log |\alpha|$  as indicated in Figure 9.8.

**Exercise 9.3.** Prove that  $\omega_m$ , the geometric mean of the pole and zero location, is given by  $\omega_m = \frac{1}{T\sqrt{\alpha}}$ .

Lag controllers are used for two distinct purposes:

1. To increase the phase margin. This is done indirectly by lowering the high frequency gain.
2. Boost low frequency gain to improve steady-state tracking and disturbance rejection without having too much effect on gain margin, phase margin nor high frequency behaviour.

The loop shaping idea behind lag design is illustrated in Figure 9.9.

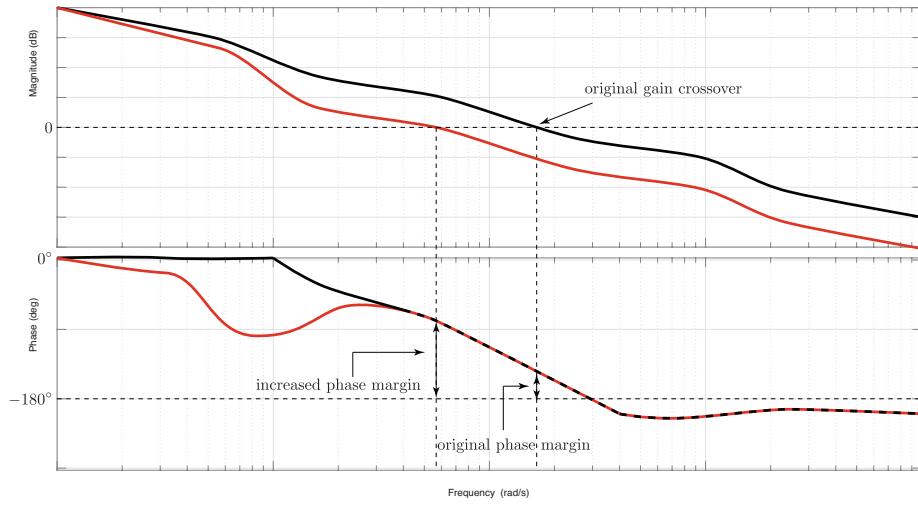


Figure 9.9: The black plot shows the Bode plot of  $KP(j\omega)$  where  $K$  has been chosen to meet steady-state tracking and disturbance rejection specifications. The red plot is the lag compensated system  $KC_1(j\omega)P(j\omega)$ . The lag controller reduces the high frequency gain without phase lag. This lowers the gain crossover frequency which, for “nice plants,” increases the phase margin.

**Example 9.3.1. (Increasing phase margin with a lag controller)** The plant transfer function is

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

The specifications are:

- (a) When  $r(t)$  is a unit ramp, the steady-state tracking error must be less than or equal to 0.05.
- (b) A phase margin of  $45^\circ$  for adequate damping in transient response.

**Step 1:** Choose  $K$  to meet the tracking specification.

$$\begin{aligned} E(s) &= \frac{1}{1 + C(s)P(s)} R(s) \quad (\text{TF from } e \text{ to } r) \\ &= \frac{1}{1 + K \frac{\alpha Ts+1}{Ts+1} \frac{1}{s(s+2)}} \frac{1}{s^2} \\ &= \frac{(s+2)(Ts+1)}{s(s+2)(Ts+1) + K(\alpha Ts+1)} \frac{1}{s}. \end{aligned}$$

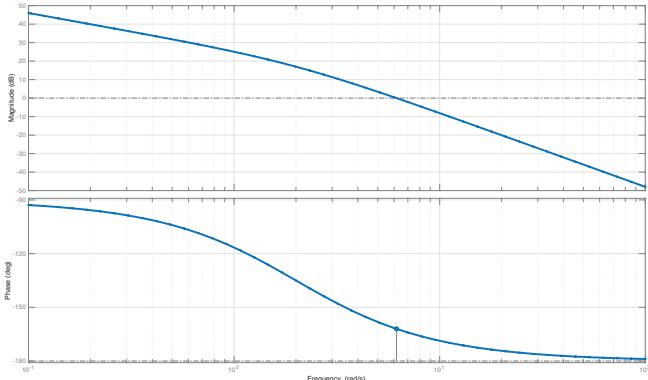
Assuming closed-loop stability, use the FVT

$$e_{ss} = \lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} sE(s) = \frac{2}{K}.$$

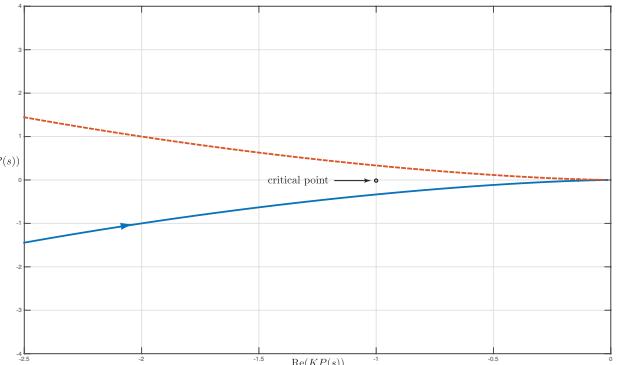
So specification (a) is satisfied if  $K \geq 40$ . We'll pick  $K = 40$  so that

$$KP(s) = \frac{40}{s(s+2)}.$$

**Step 2:** Draw the Bode plot of  $KP(s)$ . It is shown in Figure 9.10a. From Figure 9.10a we find that the gain



(a) Bode plot of  $KP(j\omega)$ .



(b) Nyquist plot of  $KP(s)$ .

Figure 9.10: Bode and Nyquist plot for Example 9.3.1 with only proportional control.

crossover frequency is  $\omega_{gc} = 6.17$  rad/s and the phase margin is  $\Phi_{pm} = 18^\circ$ . Figure 9.10b shows the Nyquist plot of  $KP(s)$ . The system is closed-loop stable with  $K = 40$  since there are no encirclements of  $-1 + j0$  (not shown in Figure 9.10b is the semi-circle of infinite radius on the right). However we see that a rotation of the plot by  $-18^\circ$  will result in an encirclement.

**Step 3:** We now design the lag compensator  $C_1$ . The idea is to reduce the high frequency gain and increase  $\Phi_{pm}$  while not changing the low frequency gain (thus preserving Specification (a) achieved in Step 1).

We want  $\Phi_{pm} = 45^\circ$  but we'll actually aim for  $50^\circ$ , a bit more than  $45^\circ$ , to compensate for the fact that the phase plot of a lag controller only approaches zero asymptotically as frequency increases.

From the Bode plot in Step 2 we find the frequency at which we have the correct phase margin. We obtain

$$\Phi_{pm}^{\text{des}} = 50^\circ = 180 + \angle(KP(j\omega)) \quad \text{when } \omega = 1.7 \text{ rad/s.}$$

So we aim to make the gain crossover frequency 1.7 rad/s.

Again, from the Bode plot of  $KP(j\omega)$ , we see that the gain at  $\omega = 1.7$  rad/s is 19 dB. Therefore we want to reduce the gain by 19 dB at  $\omega = 1.7$  without changing the phase there. We have (see Figure 9.8 and Exercise 9.2)

$$20 \log |\alpha| = -19 \text{dB} \Leftrightarrow \alpha = 0.111.$$

We now pick  $T$  so that we don't change the phase near  $\omega = 1.7$  by ensuring that

$$\frac{10}{\alpha T} \leq 1.7.$$

I'll pick  $T = 52.7$  (equality).

**Step 4:** We now plot the Bode plot of the lag compensated loop gain  $L(s) = KC_1(s)P(s)$ . It is shown in Figure 9.11. From Figure 9.11 we find that the phase margin is now  $\Phi_{pm} = 44.6^\circ$ . Close enough. Figure 9.12a

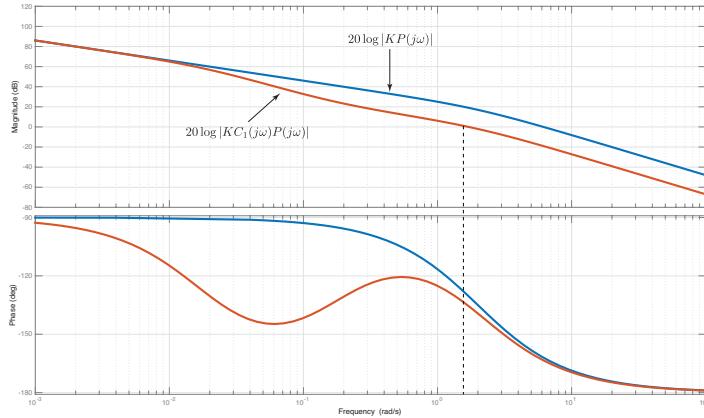


Figure 9.11: Bode plot of  $KC_1(j\omega)P(j\omega)$  (red) and  $KP(j\omega)$  (blue) for Example 9.3.1.

shows the Nyquist plot of the compensated system. We see that the curve is now much farther from the critical point.

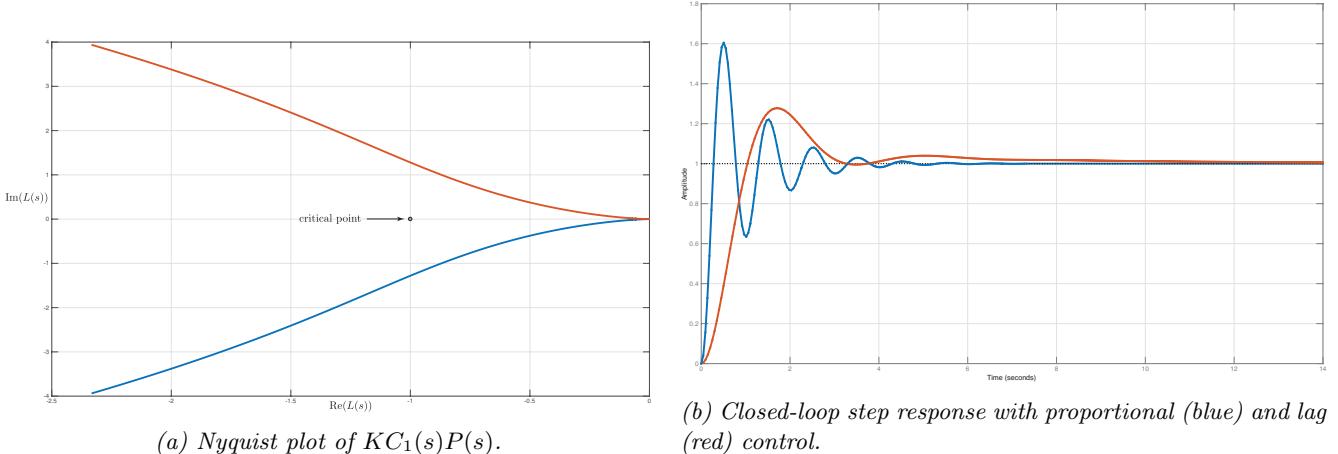


Figure 9.12: Nyquist plot and step response for Example 9.3.1 with final lag controller.

Step responses are shown in Figure 9.12b. The step response of  $KP/(1+KP)$ , that is, the plant compensated only by the gain for Spec (a), is fast but oscillatory. The step response of  $PC/(1+PC)$  is slower, typical of lag compensation<sup>1</sup>, but less oscillatory which was the goal of Specification (b). ▲

<sup>1</sup>A lag controller reduces the gain crossover frequency which, using our rule-of-thumb from Section 9.2, reduces the closed-loop bandwidth. Lowering the bandwidth, as we saw in Chapter 4, results in a slower response.

The MATLAB code below was used to design the controller in the above example.

```

1 s = tf('s'); P = 1/s/(s+2);
2 K = 40;
3 [mag, phase, w] = bode(K*P);
4 Phi_desired = 45; delta = 5;
5 [i] = min(abs(phase + 180 - Phi_desired - delta)); % find freq. with desired phase
6 alpha = 1/mag(i);
7 T = 10/alpha/w(i);
8 C = K*(alpha*T*s + 1)/(T*s + 1);
9 [Gm,Pm,Wpc,Wgc] = margin(C*P); % check if the design meets the spec

```

### Procedure for lag controller design

The given specifications are:

- (a) Steady-state specification (determined by tracking or disturbance rejection requirement),
  - (b) Increase phase margin to be greater than or equal to  $\Phi_{pm}^{des}$  (determined by damping ratio or robustness requirements).
1. Use FVT to fix the steady-state gain  $K$  of the controller.
  2. Draw the Bode plot of  $KP(j\omega)$ . Check  $\Phi_{pm}$ .
  3. If  $\Phi_{pm}$  specification is met, we're done (a proportional controller can do the job!). Otherwise, find  $\omega^*$  such that

$$\text{dist}(\angle(KP(j\omega^*)), 180^\circ) = \Phi_{pm}^{des} + \delta.$$

Here  $\Phi_{pm}^{des}$  is the given specification and  $\delta$  is a buffer to account for approximations, usually  $5^\circ$ .

4. Shift the gain down at the frequency  $\omega^*$  from step 3 to get a new gain crossover frequency

$$\alpha = \frac{1}{K|P(j\omega)|}.$$

5. Ensure that the phase isn't affected near the frequency  $\omega^*$  from steps 3, 4

$$\frac{10}{\alpha T} \leq \omega^*.$$

6. Simulate the closed-loop system and check Bode plot of  $C(j\omega)P(j\omega)$  to make sure all specifications are met.

The next example illustrates the second main use of lag controllers.

**Example 9.3.2. (Boosting low frequency gain without changing high frequency gain)** Consider the plant with time-delay

$$P(s) = 10.4e^{-0.05s} \frac{1 + 0.48s}{s(s + 2)(1 + 0.13s)}.$$

The specifications are:

- (a) When  $r(t)$  is a unit ramp, the steady-state tracking error must be less than or equal to 0.05.
- (b) The phase margin and bandwidth should not be significantly changed.

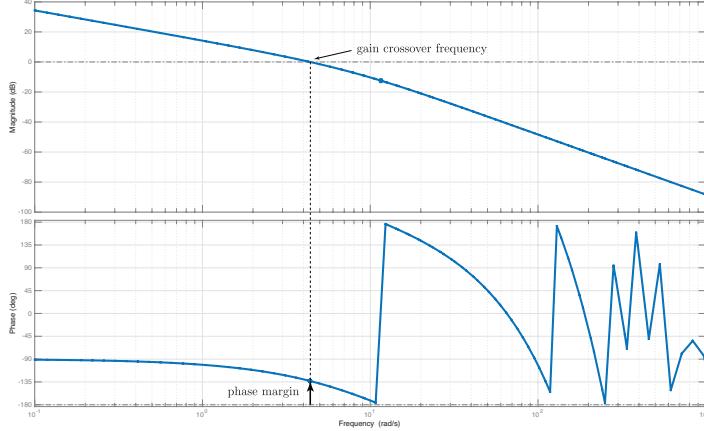


Figure 9.13: Bode plot of  $P(j\omega)$  for Example 9.3.2.

This is a typical lag compensation problem because we only want to increase the low frequency gain. The Bode plot of  $P(j\omega)$  is shown in Figure 9.13. From the Bode plot we obtain that the phase margin is approximately  $47^\circ$  and that  $\omega_{BW} \approx \omega_{gc} = 4.4$  rad/s. We now have numbers to work with for the second specification.

**Step 1:** Choose  $K$  to meet the tracking specification.

$$\begin{aligned} E(s) &= \frac{1}{1 + C(s)P(s)} R(s) \quad (\text{TF from } e \text{ to } r) \\ &= \frac{1}{1 + K \frac{\alpha Ts + 1}{Ts + 1} 10.4e^{-0.05s} \frac{1 + 0.48s}{s(s+2)(1+0.13s)}} \frac{1}{s^2} \\ &= \frac{(s+2)(Ts+1)(1+0.13s)}{s(s+2)(Ts+1)(1+0.13s) + 10.4Ke^{-0.05s}(Ts+1)(1+0.48s)} \frac{1}{s}. \end{aligned}$$

Assuming closed-loop stability, use the FVT

$$e_{ss} = \lim_{t \rightarrow \infty} e(t) = \lim_{s \rightarrow 0} sE(s) = \frac{2}{10.4K}.$$

So specification (a) is satisfied if  $K \geq 3.85$ . We'll pick  $K = 4$  so that

$$KP(s) = 41.6e^{-0.05s} \frac{1 + 0.48s}{s(s+2)(1+0.13s)}.$$

**Step 2:** Draw the Bode plot of  $KP(s)$ . Actually, there is no need in this case since Figure 9.13 has all the needed data but remember to account for the increase in gain from our choice of  $K$ .

**Step 3:** The desired gain crossover frequency is  $\omega_{gc} = 4.4$  rad/s and, since we've only added a gain of  $K = 4$ , the gain of  $KP(j\omega)$  at  $\omega = 4.4$  rad/s is exactly  $20 \log |K|$ .

**Step 4:** We shift the gain down at  $\omega = 4.4$  to get

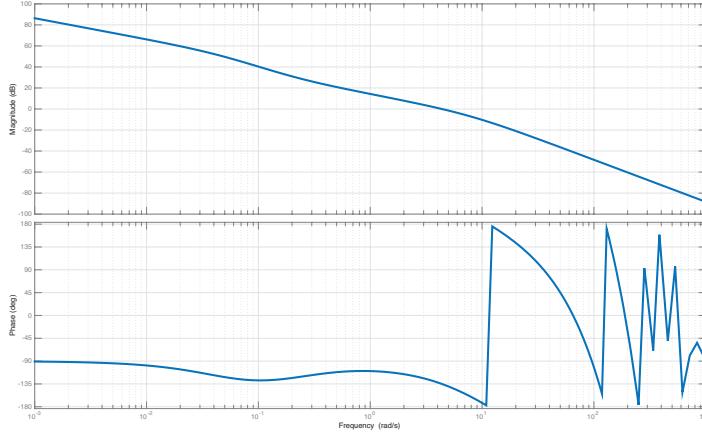
$$\alpha = \frac{1}{K|P(j\omega)|} = \frac{1}{K} = 0.25.$$

**Step 5:** Ensure the phase isn't affected near  $\omega = 4.4$

$$\frac{10}{\alpha T} \leq 4.4.$$

I'll take  $T = 9.1$ . Our overall lag controller is

$$C(s) = KC_1(s) = 4 \frac{0.25(9.1)s + 1}{9.1s + 1} = \frac{9.1s + 4}{9.1s + 1}.$$

Figure 9.14: Bode plot of  $KC_1(j\omega)P(j\omega)$  for Example 9.3.2.

**Step 6:** The Bode plot of the lag compensated loop gain  $L(s) = KC_1(s)P(s)$  is shown in Figure 9.14. From Figure 9.14 we find that the phase margin is  $\Phi_{pm} = 45^\circ$  at  $\omega_{gc} = 4.38$  rad/s. Close enough.

**Exercise 9.4.** Simulate the closed-loop step responses of  $P/(1+P)$  and  $CP/(1+CP)$ . Verify that our lag compensated system has a higher settling time.

Exercise 9.4 shows that the step response of the compensated system is slower. This is typical of a lag compensator.

**Exercise 9.5.** Simulate the closed-loop response of the system to a ramp input and verify that our controller meets the tracking specification.

Exercise 9.5 shows that our lag compensated system does a much better job tracking ramp signals. ▲

### 9.3.1 Proportional-Integral controllers

A lag controller is very closely related to the PI controller

$$C(s) = K \left( 1 + \frac{1}{T_i s} \right).$$

Expressing the PI controller in the form

$$C(s) = \frac{K}{T_i} \left( \frac{T_i s + 1}{s} \right)$$

it can be viewed as a special case of the lag controller in which the lag controller's pole has been moved all the way to the origin. The asymptotic Bode plot of a PI controller is shown in Figure 9.15. The high frequency gain of a PI controller equals  $20 \log |K|$  and its high frequency phase equals zero<sup>2</sup>. The PI controller's gains can be selected using the same procedure as a lag controller.

**Example 9.3.3.** Let's re-do Example 9.3.1 using a PI controller. Once again the specifications are:

- (a) When  $r(t)$  is a unit ramp, the steady-state tracking error must be less than or equal to 0.05.
- (b) A phase margin of  $45^\circ$  for adequate damping in transient response.

Observe that the PI controller has integral action (a pole at  $s = 0$ ) and so  $C(s)P(s)$  will have two poles at the origin. Therefore ramps will be perfectly tracked in steady-state meaning that we only need to worry about the phase margin specification.

<sup>2</sup>In contrast to a lag controller whose high frequency gain is  $20 \log |\alpha|$  and high frequency phase equals zero.

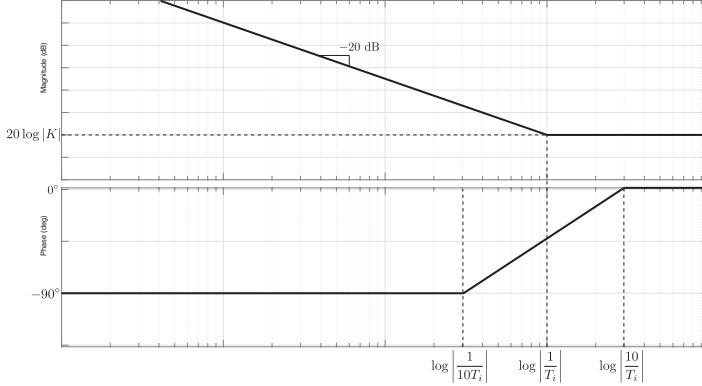


Figure 9.15: Asymptotic Bode plot of a PI controller.

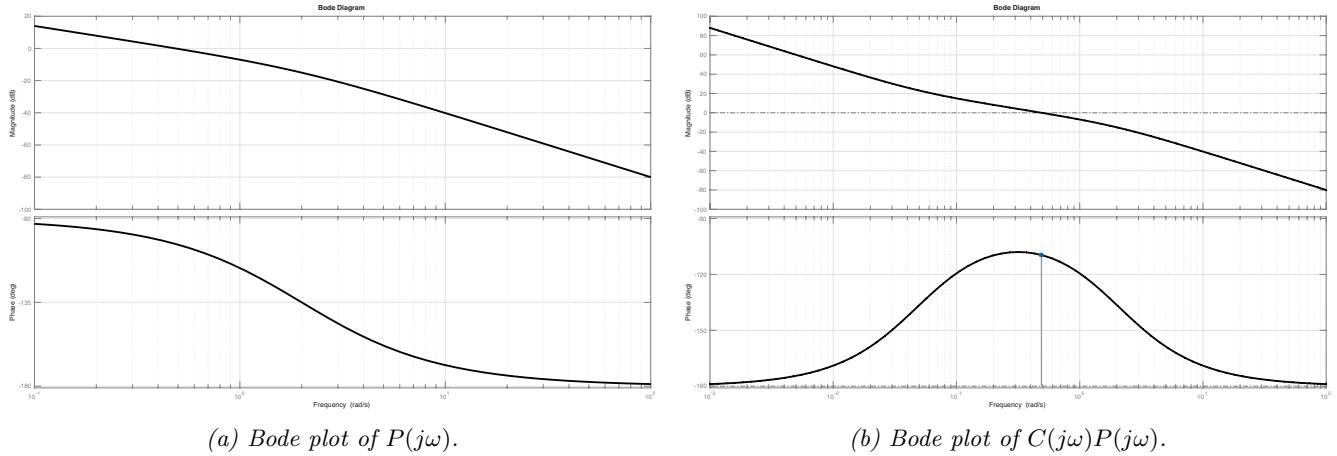
(a) Bode plot of  $P(j\omega)$ .(b) Bode plot of  $C(j\omega)P(j\omega)$ .

Figure 9.16: Uncompensated and PI compensated Bode plots for Example 9.3.3.

**Step 1:** We already meet the tracking specification so there is nothing to do in this step.

**Step 2:** Draw Bode plot of  $P(j\omega)$ . See Figure 9.16a.

**Steps 3 – 4:** From Figure 9.16a we have that the phase margin is  $76^\circ$  at  $\omega_{gc} = 0.5$  rad/s. We're already meeting the specification.

**Step 5:** Since the phase margin specification is already met, we select  $K$  and  $T_i$  to ensure that neither the gain nor the phase is affected very much near the crossover frequency  $\omega_{gc} = 0.5$  rad/s. Looking at the gain Bode plot from Figure 9.15, we should pick  $K = 1$  to ensure that the gain isn't changed near the crossover frequency. Next pick  $T_i$  to ensure the PI controller does not affect the phase near the crossover frequency. Looking at the phase Bode plot from Figure 9.15, we should pick

$$\frac{10}{T_i} \leq 0.5.$$

I'll take  $T_i = 20$ . Our controller is  $C(s) = 1 + 0.05s^{-1}$ .

**Step 6:** The Bode plot of the loop gain  $L(s) = C(s)P(s)$  in Figure 9.16b meets all the specifications. ▲

## 9.4 Lead compensation

We again separate the controller into two parts  $K$  and  $C_1(s)$  as shown in Figure 9.17. The controller is

$$C(s) = KC_1(s) = K \frac{\alpha Ts + 1}{Ts + 1}, \quad \alpha > 1, \quad T > 0, \quad K > 0, \quad (\text{lead controller}). \quad (9.8)$$

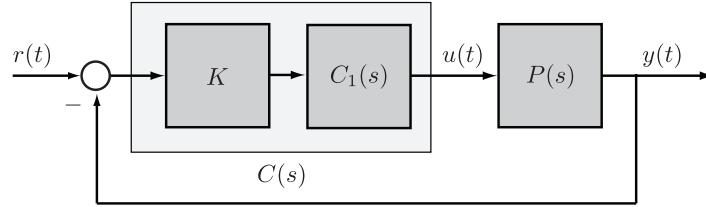


Figure 9.17: Lead controller block diagram.

The lead controller has a steady-state gain of  $C(0) = KC_1(0) = K$  and its pole and a zero locations are shown in Figure 9.18.

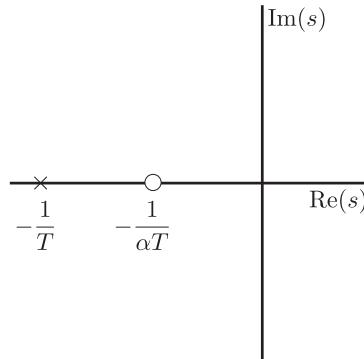


Figure 9.18: Pole-zero configuration of a lead controller: the pole is always farther from the imaginary axis than the zero.

The asymptotic Bode plot of  $C_1(s)$  is shown in Figure 9.19. A lead controller is a high pass filter. Lead

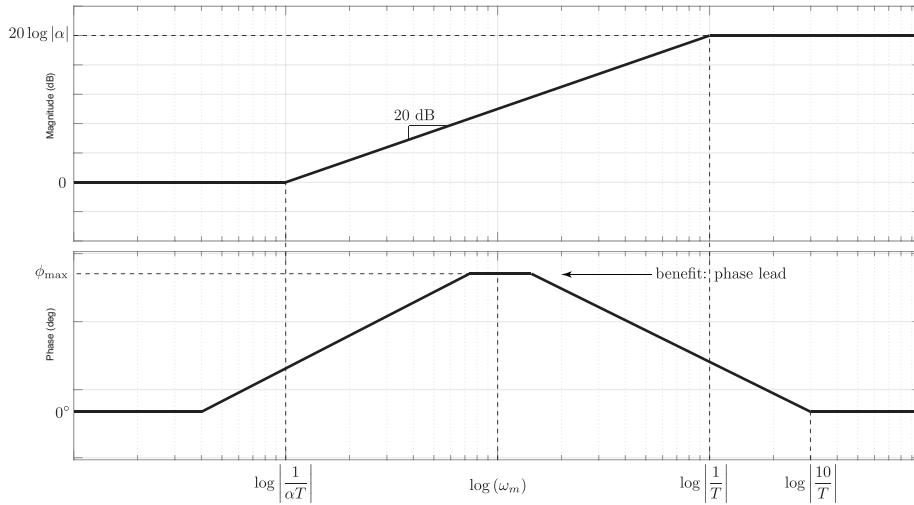


Figure 9.19: Asymptotic Bode plot of a lead controller.

controllers are used for two distinct purposes.

1. To increase the phase margin  $\Phi_{pm}$  by adding phase at the appropriate frequency while also meeting a steady-state tracking requirement.
2. Increase phase margin while simultaneously increasing the closed-loop bandwidth to make the system response faster.

The loop shaping idea behind lead design is illustrated in Figure 9.20. Before doing a design example we need

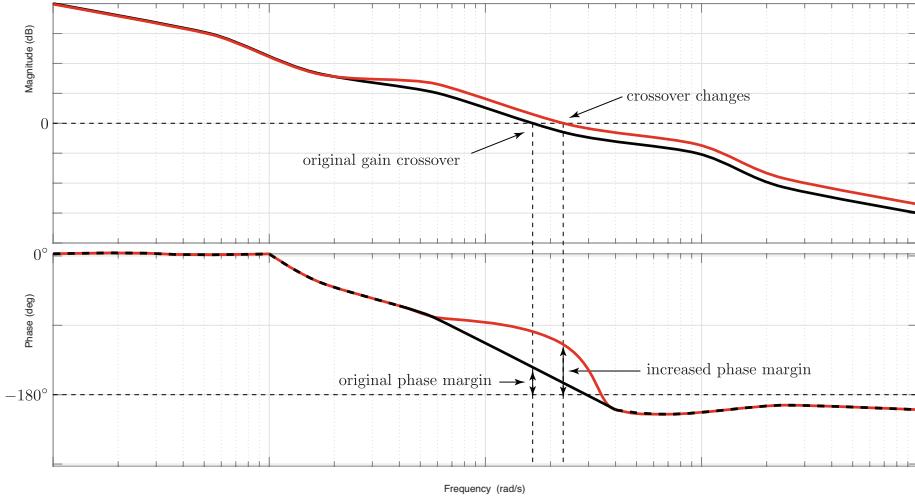


Figure 9.20: The black plot shows the Bode plot of  $KP(j\omega)$  where  $K$  has been chosen to meet steady-state tracking and disturbance rejection specifications. The red plot is the lead compensated system  $KC_1(j\omega)P(j\omega)$ . The lead controller increases the phase of the system at a designer specified frequency. An undesirable side effect is that it also increases the gain at the specified frequency. This changes  $\omega_{gc}$ .

to derive a few design equations. We'll need three formulas.

1. The frequency  $\omega_m$  (see Figure 9.19). This is the midpoint between  $\frac{1}{\alpha T}$  and  $\frac{1}{T}$  on the logarithmically scaled frequency axis. From Exercise 9.3 we have

$$\omega_m = \frac{1}{T\sqrt{\alpha}}. \quad (9.9)$$

2. The magnitude of  $C_1(j\omega)$  at  $\omega_m$ . This is the midpoint between  $20 \log |1|$  and  $20 \log |\alpha|$

$$\begin{aligned} \log |C_1(j\omega_m)| &= \frac{1}{2} (\log |1| + \log |\alpha|) \\ &= \log (\sqrt{\alpha}). \end{aligned}$$

Therefore

$$|C_1(j\omega_m)| = \sqrt{\alpha}. \quad (9.10)$$

3. The angle  $\phi_{\max}$ : This is the angle of  $C_1(j\omega_m)$  (see Figure 9.19). Thus

$$\begin{aligned} \phi_{\max} &= \angle C_1(j\omega_m) \\ &= \angle \frac{1 + \sqrt{\alpha}j}{1 + \frac{1}{\sqrt{\alpha}}j}. \end{aligned}$$

By the sine law (see Figure 9.21)

$$\frac{\sin(\phi_{\max})}{\sqrt{\alpha} - \frac{1}{\sqrt{\alpha}}} = \frac{\sin(\theta)}{\sqrt{1 + \frac{1}{\alpha}}}.$$

But  $\sin(\theta) = \frac{1}{\sqrt{1+\alpha}}$ . Thus

$$\sin(\phi_{\max}) = \left( \sqrt{\alpha} - \frac{1}{\sqrt{\alpha}} \right) \frac{1}{\sqrt{1 + \alpha} \sqrt{1 + \frac{1}{\alpha}}} = \frac{\alpha - 1}{\alpha + 1},$$

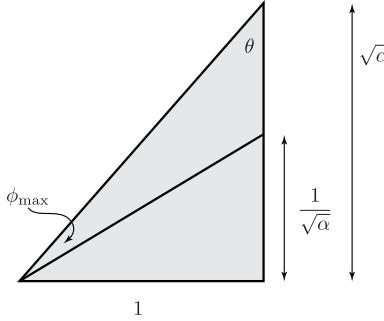


Figure 9.21: Law of sines.

and hence

$$\phi_{\max} = \sin^{-1} \left( \frac{\alpha - 1}{\alpha + 1} \right), \quad \alpha = \frac{1 + \sin(\phi_{\max})}{1 - \sin(\phi_{\max})}. \quad (9.11)$$

**Example 9.4.1. (Increasing phase margin with a lead controller)** Let's re-do Example 9.3.1 using a lead controller.

$$P(s) = \frac{1}{s(s+2)} \quad \text{Specs : (a) } |e_{ss}| \leq 0.05 \text{ for } r(t) = t, \\ \text{(b) } \Phi_{pm} = 45^\circ.$$

We start with a trick; express the lead controller (9.8) in the form

$$C(s) = K \frac{\alpha Ts + 1}{Ts + 1} =: \frac{\hat{K}}{\sqrt{\alpha}} \frac{\alpha Ts + 1}{Ts + 1}.$$

The  $\sqrt{\alpha}$  term is ugly, but as we'll see, it's convenient for design purposes.

**Step 1:** (same as lag) Choose  $\hat{K} \geq 40$  so that  $\hat{K}P(s)$  meets the tracking spec. Boost  $\hat{K}$  by 10dB (a guess) to account for the magnitude reduction that will result from the  $\sqrt{\alpha}$  term in our trick. I'll pick  $\hat{K} = 40 \cdot 10^{\frac{1}{2}}$ . Draw the Bode plot of  $\hat{K}P(s)$  and find that  $\Phi_{pm} = 10.2^\circ$  at  $\omega_{gc} = 11.2$  rad/s.

**Step 2:** We need at least  $45 - 10.2 = 34.8^\circ$  phase addition in order to meet the spec so we set  $\phi_{\max} = 34.8^\circ$ . The lead controller will add the most phase at  $\omega_m$  so we set  $\omega_m = \omega_{gc} = 11.2$  rad/s to get the maximum phase addition at the phase crossover frequency.

**Step 3:** Now that we've selected numerical values for  $\phi_{\max}$  and  $\omega_m$ , we can use our design equations. We get

$$\alpha = \frac{1 + \sin(\phi_{\max})}{1 - \sin(\phi_{\max})} = 3.66, \quad T = \frac{1}{\omega_m \sqrt{\alpha}} = 0.0467$$

and we determine that

$$K = \frac{\hat{K}}{\sqrt{\alpha}} = \frac{40 \cdot 10^{\frac{1}{2}}}{\sqrt{3.66}} = 66.13.$$

The lead controller is

$$C(s) = K \frac{\alpha Ts + 1}{Ts + 1}, \quad \alpha = 3.66, \quad T = 0.0467, \quad K = 66.13.$$

**Step 4:** We now plot the Bode plot of the lead compensated loop gain  $L(s) = KC_1(s)P(s)$ . It is shown in Figure 9.22. From Figure 9.22 we find that the phase margin is now  $\Phi_{pm} = 45^\circ$ .  $\blacktriangle$

The MATLAB code below was used to design the controller in the above example.

```

1 s = tf('s'); P = 1/s/(s+2);
2 hatK = 40*sqrt(10); % boost of 10dB
3 [Gm,Pm,Wpc,Wgc] = margin(hatK*P); % get margins of partially compensated system

```

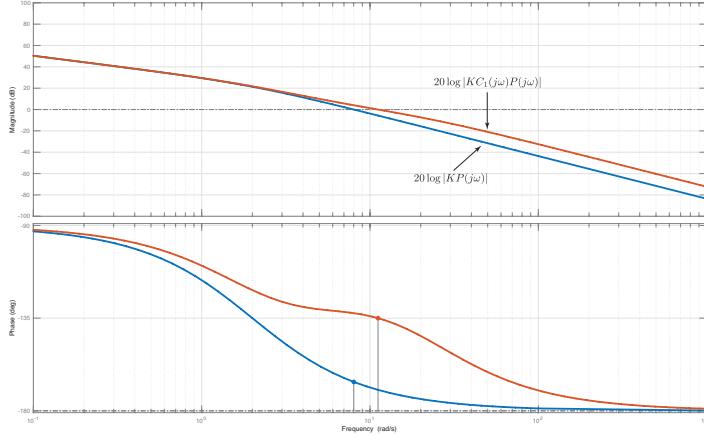


Figure 9.22: Bode plot of  $KC_1(j\omega)P(j\omega)$  (red) and  $KP(j\omega)$  (blue) for Example 9.3.1.

```

4 phi_max = 45 - Pm; Wm = Wcp;
5 alpha = (1+sind(phi_max)) / (1 - sind(phi_max));
6 T = 1/Wm/sqrt(alpha);
7 K = hatK/sqrt(alpha);
8 C = K*(alpha*T*s + 1) / (T*s + 1);
9 [Gm,Pm,Wpc,Wgc] = margin(C*P); % check if the design meets the spec

```

**Exercise 9.6.** Simulate and compare the step responses of the lead and lag controllers from Examples 9.3.1 and 9.4.1. Generate plots of the respective control signals. They both meet the specifications but the lead controller has a faster response (lead increases  $\omega_{gc}$  which increases the bandwidth) at the expense of using more control effort.

The next example illustrates the second main use of lead controllers.

**Example 9.4.2. (Increasing phase margin and bandwidth with lead controller)** Consider the plant

$$P(s) = 4e^{-0.05s} \frac{1}{s(s+2)}$$

and the specifications

- (a) Phase margin of at least  $50^\circ$ .
- (b) Settling time of no more than 2 seconds.

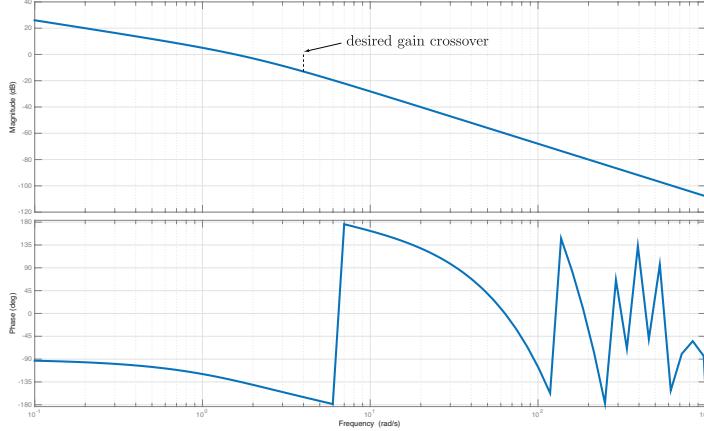
We start by converting the second specification into a frequency domain constraint. Assuming the system behaves like a second order underdamped system we can use the following approximations.

$$\begin{aligned} \Phi_{pm}^{des} = 50^\circ &\Rightarrow \zeta = 0.5, \\ T_s \approx \frac{4}{\zeta \omega_n} \leq 2 &\Rightarrow \zeta \omega_n \geq 2, \\ &\Rightarrow \omega_n \geq 4. \end{aligned}$$

For an underdamped second order system  $\omega_{BW} \approx \omega_n$  so we want  $\omega_{gc} \approx \omega_{BW} = 4$  rad/s. Of course at the end we have to check that our controller satisfies the specifications for the actual plant and not just for a second order approximation.

**Step 1:** Using the trick from the previous example, define  $\hat{K} = K\sqrt{\alpha}$ . Now choose the gain  $\hat{K}$  to meet the bandwidth specification. The Bode plot of  $P(j\omega)$  is shown in Figure 9.23. From the Bode plot we see that we must boost the gain by about 14dB which yields

$$\hat{K} = 10^{\frac{14}{20}} = 5.$$

Figure 9.23: Bode plot of  $P(j\omega)$  for Example 9.4.2.

**Step 2:** Draw Bode plot of  $\hat{K}P(j\omega)$ . Since we already have the Bode plot of  $P(j\omega)$ , we have that the Bode plot of  $\hat{K}P(j\omega)$  is the same as that in Figure 9.23 with the magnitude plot shifted upwards by 14dB.

**Steps 3 – 4:** We want the crossover frequency to be 4 rad/s. From Figure 9.23 we have that the phase at  $\omega = 4$  is about  $-165^\circ$ . Therefore we need to add at least

$$\phi_{\max} = 50^\circ - 15^\circ = 35^\circ.$$

**Step 5:** For  $35^\circ$  the formula (9.11) gives

$$\sin(35^\circ) = \frac{\alpha - 1}{\alpha + 1} \Rightarrow \alpha = 3.69.$$

This yields  $K = \hat{K}/\sqrt{\alpha} = 2.6$ .

**Step 6:** We want to add the phase at exactly 4 rad/s so using (9.9) we get

$$T = \frac{1}{4\sqrt{3.69}} = 0.1301.$$

In summary, the controller is

$$C(s) = K \frac{\alpha Ts + 1}{Ts + 1}, \quad \alpha = 3.69, T = 0.13, K = 2.6.$$

The Bode plot of the lead compensated loop gain  $L(j\omega) = C(j\omega)P(j\omega)$  is shown in Figure 9.24a. From Figure 9.24a we find that the phase margin is  $\Phi_{pm} = 47^\circ$  at  $\omega_{gc} = 4.38$  rad/s. Close enough. The step response of the closed-loop system is shown in Figure 9.24b. The settling time is 1.4 seconds which meets the specification. ▲

### Procedure for lead controller design

The given specifications are:

- (a) Increase phase margin to be at least equal to  $\Phi_{pm}^{des}$  (determined by damping ratio or robustness requirements).
- (b) One of the following:
  - (i) Steady-state specification (determined by tracking or disturbance rejection requirement),
  - (ii) A desired closed-loop bandwidth (determined by time-domain specifications or noise rejection requirements),

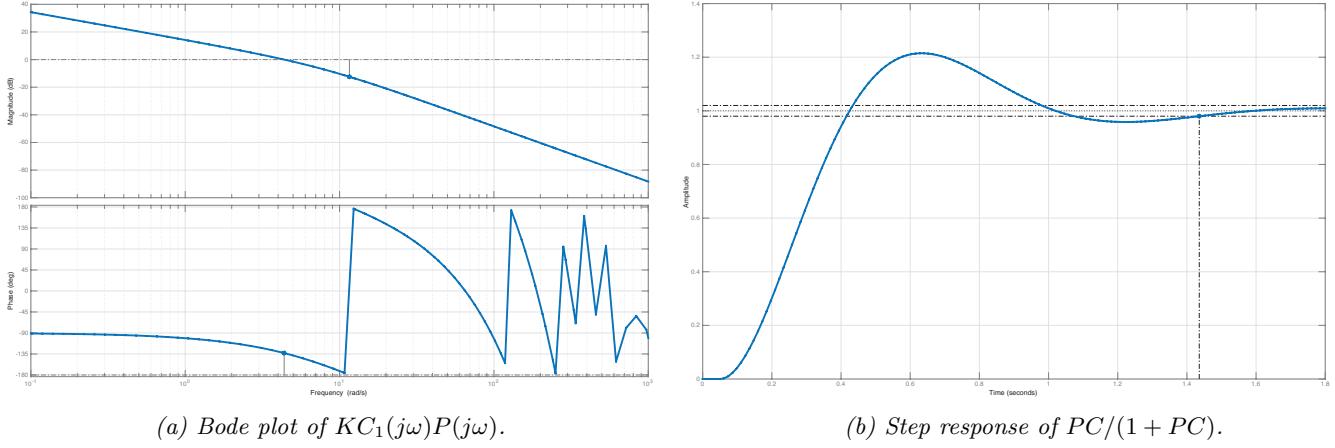


Figure 9.24: Bode plot of loop gain and step response for Example 9.4.2.

1. Define  $\hat{K} := K\sqrt{\alpha}$ .
  - (i) Use FVT to select  $\hat{K}$  such that  $\hat{K}P(s)$  meets the steady-state specification. Purposefully boost the gain by 10dB to compensate for the distortion that comes from  $\alpha$ .
  - (ii) Pick  $\hat{K}$  so that the gain crossover frequency of  $\hat{K}P(j\omega)$  equals the desired bandwidth.
2. Draw the Bode plot of  $\hat{K}P(j\omega)$ .
3. Find  $\omega_{gc}$  and  $\Phi_{pm}$ ; set  $\omega_m = \omega_{gc}$ .
4. Determine the amount of phase to add:  $\phi_{max} = \Phi_{pm}^{des} - \Phi_{pm}$ .
5. Compute  $\alpha$  using (9.11). Set  $K = \hat{K}/\sqrt{\alpha}$ .
6. Compute  $T$  using (9.9).
7. Simulate closed-loop system and check Bode plot of  $KC_1(j\omega)P(j\omega)$  to make sure all specifications are met. If they aren't met, you can return to Step 1 and use a different  $\hat{K}$ .

#### 9.4.1 Proportional-Derivative controllers

Recall that an ideal PD controller has the form

$$C(s) = K_p (1 + T_d s).$$

Its asymptotic Bode plot is shown in Figure 9.25.

As discussed in Chapter 7 this controller cannot be implemented exactly. Instead we low pass filter the derivative term to obtain

$$C(s) = K_p \left( 1 + \frac{T_d s}{\tau_d s + 1} \right).$$

Re-arranging this expression we have

$$C(s) = K_p \left( \frac{(\tau_d + T_d)s + 1}{\tau_d s + 1} \right).$$

This is exactly the form of a lead controller with

$$K = K_p, \quad T = \tau_d, \quad \alpha = 1 + \frac{T_d}{\tau_d}.$$

Therefore we can use the loop shaping ideas of this section without modification to design PD controllers.

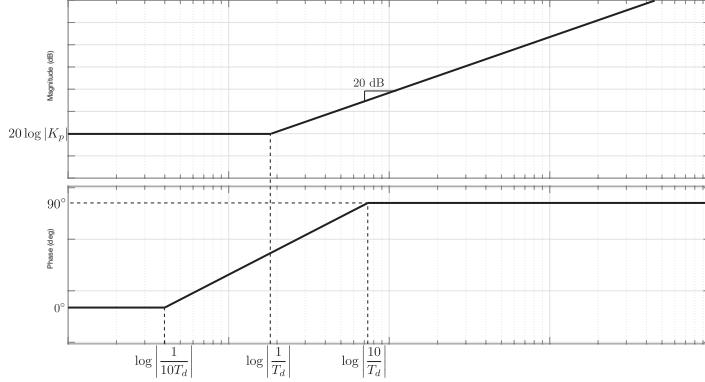


Figure 9.25: Asymptotic Bode plot of an ideal PD controller.

**Example 9.4.3. (Ideal PD controller)** Let's re-do Example 9.3.1 using an ideal PD controller.

$$P(s) = \frac{1}{s(s+2)} \quad \text{Specs : (a) } |e_{ss}| \leq 0.05 \text{ for } r(t) = t, \\ (b) \Phi_{pm} = 45^\circ.$$

**Steps 1 – 3:** Again choose  $K = 40$  to meet the tracking spec. Once again we draw the Bode plot of  $KP(s)$  and find that  $\Phi_{pm} = 18^\circ$  at  $\omega_{gc} = 6$  rad/s (see Figure 9.10a).

**Step 4:** As before, we need at least  $27^\circ$  degrees of phase addition.

**Step 5 – 6:** The phase contribution of an ideal PD controller at the frequency  $\omega$  is

$$\angle C(j\omega) = \angle K_p + \angle(1 + j\omega T_d) = \angle(1 + j\omega T_d) = \arctan(1, \omega T_d).$$

So pick  $T_d$  using

$$T_d = \frac{1}{\omega_{gc}} \tan(27^\circ + \delta) = \frac{1}{\omega_{gc}} \tan(27^\circ + 5^\circ) = 0.1041.$$

Here  $\omega_{gc} = 6$  rad/s is the frequency from the earlier steps and the buffer  $\delta = 5^\circ$  is included because the PD controller increases the gain at  $\omega_{gc}$ .

**Step 7:** The controller we've designed is

$$C(s) = 40(1 + 0.1041s).$$

The Bode plot of  $C(j\omega)P(j\omega)$  is shown in Figure 9.26a. The phase margin is  $51.8^\circ$  at  $\omega_{gc} = 6.9$  rad /s. The tracking error for a ramp reference is shown in Figure 9.26b. ▲

## 9.5 Lead-lag compensation

Consider the controller in Figure 9.27. The controller is

$$C(s) = KC_1(s)C_2(s) = K \frac{\alpha_1 T_1 s + 1}{T_1 s + 1} \frac{\alpha_2 T_2 s + 1}{T_2 s + 1}, \alpha_1 > 1, 0 < \alpha_2 < 1, T_1, T_2 > 0, K > 0, \text{ (lead-lag controller).} \quad (9.12)$$

The lead-lag controller is the product of a lead controller  $C_1(s)$  and a lag controller  $C_2(s)$ . It has a steady-state gain of  $C(0) = K$  and typical pole and a zero locations are shown in Figure 9.18.

Individually, both lead and lag controllers have weaknesses. Lag controllers reduce the closed-loop bandwidth thereby producing a slower response. Lead controllers often produce very large control signals. This motivates us to combine the controllers together into the lead-lag configuration to hopefully obtain a compromise.

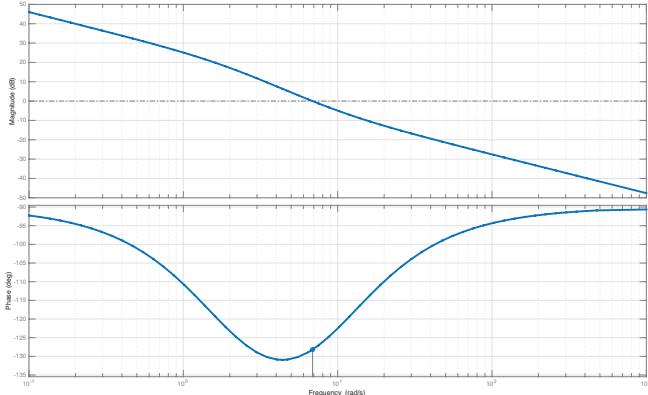
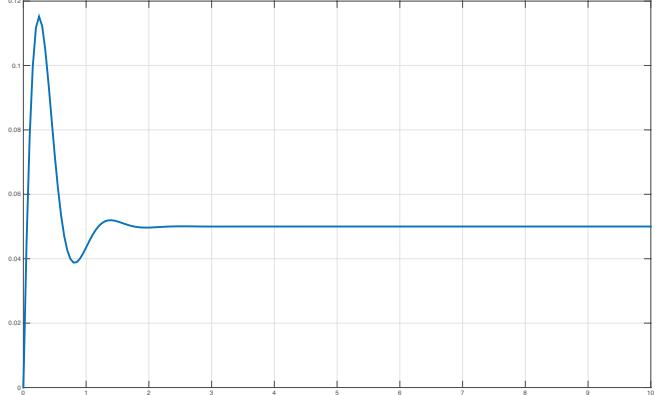
(a) Bode plot of  $C(j\omega)P(j\omega)$ .(b) Ramp tracking error  $e(t) = t - y(t)$ .

Figure 9.26: Bode plot of loop gain and ramp response for Example 9.4.3.

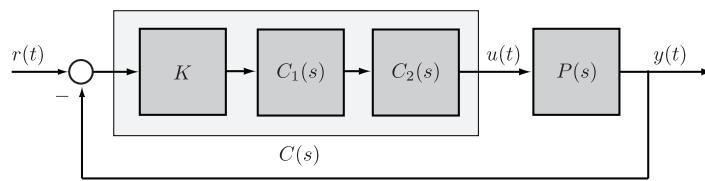


Figure 9.27: Lead-lag controller block diagram.

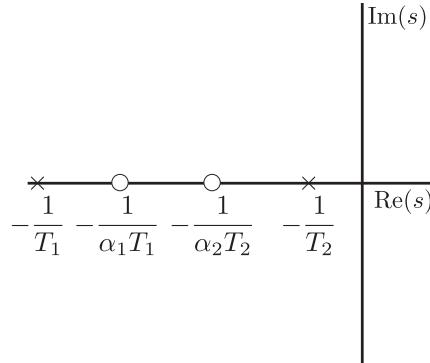


Figure 9.28: Typical pole-zero configuration of a lead-lag controller.

### Procedure for lead-lag controller design

The given specifications are:

- (a) Steady-state specification (determined by tracking or disturbance rejection requirement),
  - (b) Increase phase margin to be greater than or equal to  $\Phi_{\text{pm}}^{\text{des}}$  (determined by damping ratio or robustness requirements).
1. Define  $\hat{K} := K\sqrt{\alpha_1}$ . Use FVT to select  $\hat{K}$  such that  $\hat{K}P(s)$  meets the steady-state specification. Purposefully boost the gain by 10dB to compensate for the distortion that comes from  $\alpha_1$ .
  2. Draw the Bode plot of  $\hat{K}P(j\omega)$ .
  3. If  $\Phi_{\text{pm}}$  specification is met, we're done (a proportional controller can do the job!). Otherwise, divide  $\Phi_{\text{pm}}^{\text{des}}$  into two roughly equal parts  $\Phi_{\text{pm}}^{\text{des}} = \Phi_{\text{pm},1} + \Phi_{\text{pm},2}$ .
  4. Design the lag controller  $C_2(s)$  to get  $\Phi_{\text{pm}} = \Phi_{\text{pm},2}$ .

5. Draw the Bode plot of  $\hat{K}C_2(j\omega)P(j\omega)$ .
6. Design the lead controller  $C_1(s)$  to get  $\Phi_{pm} = \Phi_{pm}^{des}$  for the partially compensated system.
7. Simulate the closed-loop system and check Bode plot of  $C(j\omega)P(j\omega)$  to make sure all specifications are met.

**Remark 9.5.1.** The reason that the lag controller is designed first in the above procedure is that a lead controller will tend to “flatten” the phase curve. This means that if we tried to then design a lag controller, the crossover frequency will be very small and we’ll get a sluggish response. ♦

**Example 9.5.1.** Let’s design a lead-lag controller for the following:

$$P(s) = \frac{1}{s(s+1)(s+20)} \quad \text{Specs : } \begin{cases} \text{(a) } |e_{ss}| \leq 0.01 \text{ for } r(t) = t\mathbf{1}(t), \\ \text{(b) } \Phi_{pm} = 45^\circ. \end{cases}$$

**Step 1:** Define  $\hat{K} = K\sqrt{\alpha_1}$ . Assuming feedback stability, the FVT for  $r(t) = t\mathbf{1}(t)$  gives

$$e_{ss} = \lim_{s \rightarrow 0} sE(s) = \frac{20}{\hat{K}}.$$

To meet the tracking specification we need  $\hat{K} \geq 200$ . Boost  $\hat{K}$  by 10dB (a guess). I’ll pick  $\hat{K} = 200 \cdot 10^{\frac{1}{2}}$ .

**Steps 2–3:** From the Bode plot of  $\hat{K}P(j\omega)$  we find that  $\Phi_{pm} = -5^\circ$  at  $\omega_{gc} = 5.5$  rad/s. Divide  $\Phi_{pm}^{des} = 45^\circ$  into two  $\Phi_{pm,1} = \Phi_{pm,2} = 22.5^\circ$ .

**Step 4:** We follow the standard lag control design procedure to design  $C_2(s)$  to achieve a phase margin of  $\Phi_{pm,2} = 22.5^\circ$ . This yields  $\alpha_2 = 0.088$ ,  $T_2 = 75.6$  and

$$C_2(s) = \frac{6.6s + 1}{75.6s + 1}.$$

**Step 5:** The phase margin of  $\hat{K}C_2(s)P(s)$  is  $\Phi_{pm} = 23.8^\circ$  at  $\omega_{gc} = 1.5$  rad/s.

**Step 6:** We follow the standard lead control design procedure to design  $C_1(s)$  to increase the phase margin by  $\phi_{max} = \Phi_{pm}^{des} - \Phi_{pm} = 21.2^\circ$  at  $\omega_m = 1.5$  rad/s. This yields  $\alpha_1 = 2.13$ ,  $T_1 = 0.45$ ,  $K = 433$  and

$$C_1(s) = \frac{0.961s + 1}{0.45s + 1}.$$

**Step 7:** The overall controller is given by

$$C(s) = KC_1(s)C_2(s) = \frac{1207s^2 + 1599s + 200}{9.796s^2 + 28.94s + 1}.$$

and the phase margin of  $C(s)P(s)$  is  $\Phi_{pm} = 45^\circ$  as desired. ▲

The MATLAB code below was used to design the controller in Example 9.5.1.

```

1 Phi_pm_des = 45;
2 s = tf('s'); P = 1/s/(s+1)/(s+20);
3 hatK = 200*(10^(1/2)); % boost of 10dB
4 [Gm, Pm, Wpc, Wgc] = margin(hatK*P);
5
6 %lag design
7 [mag, phase, w] = bode(hatK*P);
8 Phi_pm_2 = Phi_pm_des/2; delta = 5;
9 [~, i] = min(abs(phase + 180 - Phi_pm_2 - delta)); % find freq. with desired phase
10 alpha2 = 1/mag(i);

```

```

11 T2 = 10/alpha2/w(i);
12 C2 = (alpha2*T2*s + 1)/(T2*s + 1);
13 [Gm,Pm,Wpc,Wgc] = margin(hatK*C2*P); % check if the design meets the spec
14
15 % lead design
16 phi_max = Phi_pm_des - Pm; Wm = Wgc;
17 alphal = (1+sind(phi_max))/(1 - sind(phi_max));
18 T1 = 1/Wm/sqrt(alphal);
19 C1 = (alphal*T1*s + 1)/(T1*s + 1);
20
21 % overall controller
22 K = hatK/sqrt(alphal);
23 C = K*C1*C2;
24 [Gm,Pm,Wpc,Wgc] = margin(C*P); % check if the design meets the spec

```

## 9.6 Loop shaping theory

Although loop shaping gives us a great deal of flexibility in designing the closed-loop response of a system, there are certain fundamental limits on what can be achieved. In this section we look at some theoretical facts that we have to keep in mind while designing controllers via loop shaping.

### 9.6.1 Bode's phase formula

It is a fundamental fact that if  $L = PC$  is stable and minimum phase and normalized so that  $L(0) > 0$  (positive steady-state gain), then the magnitude Bode plot uniquely determines the phase Bode plot.

**Assumption 9.6.1.** The loop gain  $L(s) = C(s)P(s)$  of the system in Figure 9.1 is stable, minimum phase and  $L(0) > 0$ . ◀

The exact formula relating the magnitude to the phase is rather complicated, and is derived using Cauchy's integral theorem [Wunsch, 2005]. Let

$$\omega_0 = \text{any frequency}$$

$$u = \text{normalized frequency} = \ln \left| \frac{\omega}{\omega_0} \right|, \text{ i.e., } e^u = \frac{\omega}{\omega_0}$$

$$M(u) = \text{normalized log magnitude} = \ln |L(j\omega_0 e^u)| = \ln |L(j\omega)|.$$

$$W(u) = \text{weighting function} = \ln (\coth |\frac{u}{2}|).$$

Recall that

$$\coth(x) = \frac{\cosh(x)}{\sinh(x)} = \frac{e^x + e^{-x}}{e^x - e^{-x}}.$$

The phase formula (in radians) is

$$\angle(L(j\omega_0)) = \frac{1}{\pi} \int_{-\infty}^{\infty} \frac{dM(u)}{du} W(u) du \quad (\text{Bode's gain/phase formula}). \quad (9.13)$$

The key point of this is that it shows that  $\angle(L(j\omega_0))$  can be expressed as an integral involving  $|L(j\omega)|$ . What it tells us is that for systems that satisfy Assumption 9.6.1, if one wishes to specify a certain magnitude characteristic for the frequency response (cf. Figure 9.4), the phase characteristic is completely determined.

It turns out we may approximate the weighting function as  $W(u) \approx \frac{\pi^2}{2} \delta(u)$ . Then the phase formula (9.13) gives

$$\angle(L(j\omega_0)) \approx \frac{\pi}{2} \frac{dM(u)}{du} \Big|_{u=0}. \quad (9.14)$$

As an example, consider the situation where

$$L(j\omega) = \frac{c}{\omega^n} \text{ near } \omega = \omega_0.$$

Thus the slope of the magnitude plot is  $-n$  per decade near  $\omega = \omega_0$ . Then

$$\begin{aligned} |L(j\omega_0 e^u)| &= \frac{c}{\omega_0^n e^{nu}} \\ \Rightarrow M(u) &= \ln |L(j\omega_0 e^u)| = \ln \frac{c}{\omega_0^n} - nu \\ \Rightarrow \frac{dM(u)}{du} &= -n \\ \Rightarrow \angle(L(j\omega_0)) &= -n \frac{\pi}{2} \text{ from (9.14).} \end{aligned}$$

Thus we arrive at the observation: If the slope of  $|L(j\omega)|$  near crossover is  $-n$ , then  $\arg(L(j\omega))$  at crossover is approximately  $-n \frac{\pi}{2}$ .

What we learn from this observation is that in transforming  $|P|$  to  $|PC|$  via, say, lag or lead compensation, we should not attempt to roll off  $|PC|$  too sharply near gain crossover. If we do,  $\angle PC$  will be too large near crossover, resulting in a negative phase margin and hence instability.

### 9.6.2 The waterbed effect

This concerns the ability to achieve the specification on the sensitivity function  $S$  shown in Figure 9.29 (cf. Section 9.1).

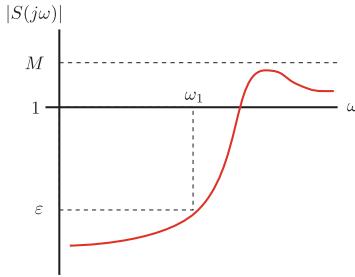


Figure 9.29: Typical desired shape for a sensitivity function.

Let us suppose  $M > 1$  and  $\omega_1 > 0$  are fixed. Can we make  $\varepsilon$  arbitrarily small? That is, can we get arbitrarily good tracking over a finite frequency range, while maintaining a given stability margin ( $1/M$ )? Or is there a positive lower bound for  $\varepsilon$ ? The answer is that arbitrarily good performance in this sense is achievable if and only if  $P(s)$  is minimum phase. Thus, non-minimum phase plants have bounds on achievable performance: As  $|S(j\omega)|$  is pushed down on one frequency range, it pops up somewhere else, like a waterbed.

To understand the result, first note that we can always factor a non-minimum phase TF  $G(s)$  as

$$G(s) = G_{mp}(s)A(s)$$

where  $G_{mp}$  is the minimum phase part of  $G$  and  $A$  is the non-minimum phase part. We normalize  $A$  so that  $|A(j\omega)| = 1$ . The transfer function  $A$  is called **all-pass** because it has gain 1 at all frequencies.

**Example 9.6.1.** Recall the TF from Example 3.8.11

$$G(s) = \frac{(1-s)}{(s+10)(s+1)}.$$

In this case we can write

$$G(s) = G_{\text{mp}}(s)A(s) = \left(\frac{1}{s+10}\right) \left(\frac{1-s}{1+s}\right).$$

▲

**Example 9.6.2.** Consider a time-delay system

$$G(s) = e^{-sT}, \quad T > 0.$$

Use an all-pass approximation of the delay as we did in Section 7.3

$$G(s) = e^{-sT} \approx \frac{-\frac{sT}{2} + 1}{\frac{sT}{2} + 1} = \frac{\frac{2}{T} - s}{\frac{2}{T} + s}.$$

This system is non-minimum phase with  $G_{\text{mp}}(s) = 1$ . So a long time-delay  $T$  acts like a slow non-minimum phase zero at  $s = 2/T$ .

▲

Here's the result:

**Theorem 9.6.2.** Suppose  $P(s)$  has a zero at  $s = z$  with  $\text{Re}(z) > 0$ . Let  $A(s)$  denote the all-pass factor of  $S(s)$ . Then there are positive constants  $c_1, c_2$ , depending only on  $\omega_1$  and  $z$ , such that

$$c_1 \log \varepsilon + c_2 \log M \geq \log |A(z)^{-1}| \geq 0.$$

**Example 9.6.3.**

$$P(s) = \frac{1-s}{(s+1)(s-p)}, \quad p > 0, \quad p \neq 1.$$

The plant has a non-minimum phase zero at  $s = 1$ . If  $C(s) = N_c/D_c$  is a stabilizing controller, then

$$S(s) = \frac{1}{1+PC} = \frac{(s+1)(s-p)D_c}{(s+1)(s-p)D_c + (1-s)N_c}.$$

So  $S$  has a non-minimum phase zero at  $s = p$  and therefore  $(s-p)/(s+p)$  is an all-pass factor of  $S$ . There may be other all-pass factors, so what we can say is that  $A(s)$  has the form

$$A(s) = \frac{s-p}{s+p} A_1(s),$$

where  $A_1(s)$  is some all-pass TF (may be 1). Evaluate at the RHP zero of the plant

$$|A(1)| = \left| \frac{1-p}{1+p} \right| |A_1(1)|.$$

Now  $|A_1(1)| \leq 1$  (why?), so

$$|A(1)| \leq \left| \frac{1-p}{1+p} \right|$$

and hence

$$|A(1)^{-1}| \geq \left| \frac{1+p}{1-p} \right|.$$

The theorem gives

$$c_1 \log \varepsilon + c_2 \log M \geq \log \left| \frac{1+p}{1-p} \right|.$$

Thus, if  $M > 1$  is fixed,  $\log \varepsilon$  cannot be arbitrarily negative, and hence  $\varepsilon$  cannot be arbitrarily small. In fact the situation is much worse if  $p \approx 1$ , that is, if the RHP plant pole and zero are close.

▲

The following quote is from Gunter Stein in the inaugural IEEE Bode Lecture at the 1989 IEEE Conference on Decision and Control describing the waterbed effect:

Sensitivity improvements in one frequency range must be paid for with sensitivity deteriorations in another frequency range, and the price is higher if the plant is open-loop unstable. This applies to every controller, no matter how it was designed.

Stein called this, the principle of Conservation of Sensitivity Dirt. This image refers to the area under the sensitivity function being analogous to a pile of dirt. If one shovels dirt away from some area (i.e. reduces sensitivity in a frequency band), then it piles up somewhere else (i.e. the sensitivity increases at other frequencies).

## 9.7 Summary

In this chapter we introduced two basic loop shaping design methods. Control design is a vast subject area with many other techniques not covered here. Key ideas to extract from this chapter are listed below.

1. Understand how adjusting the shape of the sensitivity function's Bode plot can lead to good performance.
2. Understand Figures 9.2 and 9.4.
3. Understand why we shape the Bode plot of  $L(s)$  instead of  $S(s)$  and the advantages and disadvantages of doing so.
4. Know what a lag controller is and what it is used for. Especially know Figures 9.7 and 9.8.
5. Know how to design a lag controller to meet specs on (i) stability (ii) steady-state tracking (iii) percentage overshoot (converted into a  $\Phi_{pm}$  spec).
6. Know what a lead controller is and what it is used for. Especially know Figures 9.18 and 9.19.
7. Know how to design a lead controller to meet specs on (i) stability (ii) steady-state tracking or closed-loop bandwidth (iii) percentage overshoot (converted into a  $\Phi_{pm}$  spec).
8. Be able to use the loop-shaping ideas of this chapter to design PI and PD controllers.
9. Understand the performance limitations discussed in Section 9.6 and that non-minimum phase systems have inherent performance limitations which no controller can overcome.
10. What is the waterbed effect?

---

# Epilogue

If this is the only course you take on control then it is hoped that you learned the following.

- The value of block diagrams.
- What it means for a system to be linear and the central role that linear systems play in engineering.
- The importance of mathematical models for systematic design.
- Why we use transfer functions and the frequency domain.
- What stability means.
- What feedback is and why it is useful.
- What makes a system easy or hard to control.

If you plan to take another course in control, it is hoped that you additionally learned

- How to model a system using differential equations.
- How to put a differential equation into state-space form.
- How to linearize a nonlinear state-space model.
- How to check if a system is stable using poles or eigenvalues.
- The Nyquist criterion and the meaning of stability margins.
- How to design a simple feedback controller.

The sequels to this course are ECE488 Multivariable Control Systems, which develops the state-space method in continuous time, and ECE481 Digital Control Systems, which treats computer control. There is also ECE486 Robot Dynamics and Control. The subject of control science and engineering is vast and this course can take you in many directions.

---

# Bibliography

- [Anderson and Ye, 2019] Anderson, B. and Ye, M. (2019). Recent advances in the modelling and analysis of opinion dynamics on influence networks. *International Journal of Automation and Computing*, 16(2):129–149. [2](#)
- [Association, 2015] Association, N. H. T. S. (2015). Fatality analysis reporting system. <http://www-fars.nhtsa.dot.gov/Main/index.aspx>. Retrieved March 25 2015. [1](#)
- [Åström and Hägglund, 1995] Åström, K. and Hägglund, T. (1995). *PID Controllers*. Instrument Society of America, 2nd edition. [150](#), [157](#), [162](#), [164](#)
- [Åström and Hägglund, 2006] Åström, K. and Hägglund, T. (2006). *Advanced PID Control*. The Instrumentation, Systems, and Automation Society. [150](#)
- [Åström and Murray, 2019] Åström, K. and Murray, R. (2019). *Feedback systems: An introduction for scientists and engineers*. Princeton University Press, 2 edition. [7](#), [8](#), [150](#)
- [Basar and Basar, 2001] Basar, T. and Basar, T. (2001). *Control Theory: Twenty-Five Seminal Papers (1932-1981)*. Wiley-IEEE Control Systems Society. [166](#)
- [Bellman and Kalaba, 1964] Bellman, R. and Kalaba, R. (1964). *Selected Papers on Mathematical Trends in Control Theory*. Dover. [166](#)
- [Bequette, 2005] Bequette, B. W. (2005). A critical assessment of algorithms and challenges in the development of a closed-loop artificial pancreas. *Diabetes Technology & Therapeutics*, 7(1):28–47. [2](#)
- [Cannon Jr., 2003] Cannon Jr., R. (2003). *Dynamics of Physical Systems*. Dover Publications. [18](#)
- [Chen, 1999] Chen, C. (1999). *Linear System Theory and Design*. Oxford University Press, New York, third edition. [63](#)
- [Cobelli et al., 2009] Cobelli, C., Man, C. D., Sparacino, G., Magni, L., Nicolao, G. D., and Kovatchev, B. (2009). Diabetes: Models, signals, and control. *IEEE Reviews in Biomedical Engineering*, 2:54–96. [2](#)
- [Desborough and Miller, 2002] Desborough, L. and Miller, R. (2002). Increasing customer value of industrial control performance monitoring—honeywell’s experience. In *Sixth International Conference on Chemical Process Control*, volume 98 of *AICHE Symposium Series Number 326*. [150](#)
- [Dorf and Bishop, 2011] Dorf, R. and Bishop, R. (2011). *Modern Control Systems*. Prentice Hall, 12th edition. [59](#)
- [Doyle et al., 1990] Doyle, J., Francis, B., and Tannenbaum, A. (1990). *Feedback Control Theory*. Macmillan Publishing Co. [1](#), [118](#)
- [Dullerud and Paganini, 2000] Dullerud, G. and Paganini, F. (2000). *A Course in Robust Control Theory*. Springer. [92](#), [125](#)
- [Evans, 1948] Evans, W. (1948). Graphical analysis of control systems. *Transactions of the American Institute of Electrical Engineers*, 67(1):547–551. [137](#)

- [Evans, 1950] Evans, W. (1950). Control system synthesis by root locus method. *Transactions of the American Institute of Electrical Engineers*, 69(1):66–69. [137](#)
- [Filieri et al., 2015] Filieri, A., Hoffmann, H., and Maggio, M. (2015). Automated multi-objective control for self-adaptive software design. In *Proceedings of the 2015 10th Joint Meeting on Foundations of Software Engineering*, pages 13–24. [2](#)
- [Goodwin et al., 2001] Goodwin, G., Graebe, S., and Salgado, M. (2001). *Control System Design*. Prentice Hall. [13](#), [150](#)
- [Gopal, 1963] Gopal, M. (1963). *Control Systems: Principles and Design*. McGraw-Hill. [27](#)
- [Hellerstein et al., 2004] Hellerstein, J., Diao, Y., Parekh, S., and Tilbury, D. M. (2004). *Feedback Control of Computing Systems*. Wiley-IEEE Press, New York, U.S.A. [2](#), [8](#)
- [Jia et al., 2015] Jia, P., MirTabatabaei, A., Friedkin, N., and Bullo, F. (2015). Opinion dynamics and the evolution of social power in influence networks. *SIAM Review*, 57(3):367–397. [2](#)
- [Kailath, 1980] Kailath, T. (1980). *Linear Systems*. Prentice-Hall. [13](#)
- [Keshav, 2012] Keshav, S. (2012). *Mathematical Foundations of Computer Networking*. Addison-Wesley. [8](#)
- [Kim and Kumar, 2012] Kim, K. D. and Kumar, P. R. (2012). Cyber physical systems: A perspective at the centennial. *Proceedings of the IEEE*, 100:1287–1308. [2](#)
- [Lessard et al., 2016] Lessard, L., Recht, B., and Packard, A. (2016). Analysis and Design of Optimization Algorithms via Integral Quadratic Constraints. *SIAM Journal of Optimization*, 26(1):57–95. [8](#)
- [Lorenz, 1963] Lorenz, E. N. (1963). Deterministic nonperiodic flow. *Journal of the Atmospheric Sciences*, 20:130–141. [14](#)
- [Lundberg et al., 2007] Lundberg, K., Miller, H., and Trumper, D. (2007). Initial conditions, generalized functions, and the Laplace transform troubles at the origin. *IEEE Control Systems Magazine*, 27(1):22–35. [59](#)
- [MacColl, 1945] MacColl, L. (1945). *Fundamental theory of servomechanisms*. Van Nostrand. [166](#)
- [Ny and Pappas, 2014] Ny, J. L. and Pappas, G. J. (2014). Differentially private filtering. *IEEE Transactions on Automatic Control*, 59(2):341–354. [2](#)
- [Nyquist, 1932] Nyquist, H. (1932). Regeneration theory. *Bell Laboratories Technical Journal*, 11:126–147. [166](#)
- [Patikirikorala et al., 2012] Patikirikorala, T., Colman, A., Han, J., and Wang, L. (2012). A systematic survey on the design of self-adaptive software systems using control engineering approaches. In *7th International Symposium on Software Engineering for Adaptive and Self-Managing Systems*, pages 33–42. [2](#)
- [Pothukuchi et al., 2018] Pothukuchi, R. P., Pothukuchi, S. Y., Voulgaris, P., and Torrellas, J. (2018). Yukta: multilayer resource controllers to maximize efficiency. In *Proceedings of the 45th Annual International Symposium on Computer Architecture*, pages 505–518. [2](#)
- [Qiu and Zhou, 2010] Qiu, L. and Zhou, K. (2010). *Introduction to Feedback Control*. Prentice-Hall, Toronto. [126](#)
- [Routh, 1877] Routh, E. (1877). *A Treatise on the Stability of a Given State of Motion*. Macmillan. [126](#)
- [Samad, 2017] Samad, T. (2017). A survey on industry impact and challenges thereof. *IEEE Control Systems*, 37(1):17–18. [150](#)

## 215 BIBLIOGRAPHY

- [Seierstad and Sydsæter, 1986] Seierstad, A. and Sydsæter, K. (1986). *Optimal control theory with economic applications*. Elsevier. [2](#)
- [Smith, 2015] Smith, B. W. (2015). Human error as a cause of vehicle crashes. <http://cyberlaw.stanford.edu/blog/2013/12/human-error-cause-vehicle-crashes>. Retrieved March 25 2015. [1](#)
- [Society, 2015] Society, I. C. S. (2015). Control systems are ubiquitous. <http://ieeecss.org/general/control-systems-are-ubiquitous>. Retrieved March 25 2015. [1](#)
- [Wunsch, 2005] Wunsch, A. (2005). *Complex variables with applications*. Pearson, Toronto, 3rd edition. [166](#), [208](#)
- [Ziegler and Nichols, 1942] Ziegler, J. and Nichols, N. (1942). Optimum settings for automatic controllers. *ASME Transactions*, 64(8). [157](#)

---

# Index

- All-pass, 208  
Amplitude response, 75  
Argument, 91  
    Principle, 91  
Arrival angle, 139  
Asymptotes, 85, 138  
Asymptotic Bode plot, 85  
Asymptotic stability, 67  
Autonomous system, 14  
Auxiliary polynomial, 128
- Bandwidth, 81  
Block diagram, 16  
Bode plot, 80  
Bounded, 68  
Bounded-input bounded-output stable, 68  
Break frequency, 84
- Cartesian form, 90  
Cascade connection, 51  
Centroid, 138  
Characteristic polynomial, 123  
Complementary root-locus, 147  
Complex conjugate, 90  
Complex number, 90  
Controller, 2  
Convolution, 44  
Coprime, 49
- DC motor, 3, 25  
    back emf, 26  
    rotor, 26  
    stator, 26  
Decade, 83  
Decibels, 80  
Departure angle, 139  
Derivative gain, 150  
Derivative time constant, 150  
Difference Equation, 14  
Disturbance, 2  
Dominant poles, 112  
Dominant zeros, 112
- Equilibrium configuration, 37
- Equilibrium point, 37  
Exogenous inputs, 119
- Feedforward, 153  
Final-value theorem, 73  
Fourier transform, 45  
Frequency response, 75
- Gain crossover frequency, 176  
Gain margin, 181
- Heat equation, 15  
Hurwitz, 125
- Impulse response, 64  
Input-output stable, 122  
Integral gain, 150  
Integral time constant, 150  
Internal stability, 120
- Jordan form, 63
- Lag controller, 190  
Laplace transform, 40  
Lead controller, 198  
Lead-lag controller, 204  
Lower gain margin, 181  
Lower phase margin, 177
- Matrix exponential, 61  
Minimum phase, 80  
Minimum phase system, 79
- Nilpotent, 61  
Non-dominant poles, 112  
Non-dominant zeros, 112  
Non-minimum phase, 80  
Non-minimum phase system, 79  
Nyquist contour, 167  
Nyquist plot, 167
- Open-loop control, 3  
Operational amplifiers, 24
- Peak time, 107  
Percentage overshoot, 105

- Phase crossover frequency, 180
- Phase curve, 66
- Phase margin, 177
- Phase plot, 66
- Phase response, 75
- Phasors, 91
- Plant, 2
- Polar form, 91
- Polar plot, 76
- Pole placement, 158
- Pole-zero cancellation, 124
- Proportional gain, 150
- Prototype first order system, 93
  - Time constant, 96
- Prototype second order system, 97
  - Critically damped, 99
  - Damping ratio, 99
  - Natural undamped frequency, 99
  - Overdamped, 99
  - Undamped, 99
  - Underdamped, 99
- Region of convergence, 40
- Reset, 153
- Reset time, 150
- Rise time, 108
- root-locus, 138
- Schematic diagram, 15
- Settling time, 107
- Signals, 15
- Stability margin, 184
- State of a system, 30
- State vector, 29
- Steady-state gain, 73
- Summing junctions, 16
- System, 2
- Tracking time constant, 163
- Transfer function, 47
  - improper, 48
  - proper, 48
  - rational, 48
  - strictly proper, 48
- Ultimate gain, 157
- Ultimate period, 157
- Unstable, 68
- Unstable pole-zero cancellation, 124
- Upper gain margin, 180
- Upper phase margin, 177
- Well-posed, 120